Universidad Politécnica de Madrid

Escuela Técnica Superior de Ingenieros de Telecomunicación

Sentiment and Emotion Analysis in Social Networks: modeling and linking data, affects and people

Tesis Doctoral

J. Fernando Sánchez Rada
Ingeniero de Telecomunicación

2020

Universidad Politécnica de Madrid

Escuela Técnica Superior de Ingenieros de Telecomunicación

Sentiment and Emotion Analysis in Social Networks: modeling and linking data, affects and people

Tesis Doctoral

J. Fernando Sánchez Rada
Ingeniero de Telecomunicación

2020

Departamento de Ingeniería de Sistemas Telemáticos

Escuela Técnica Superior de Ingenieros de Telecomunicación

Universidad Politécnica de Madrid

Sentiment and Emotion Analysis in Social Networks: modeling and linking data, affects and people

Autor:

J. Fernando Sánchez Rada

Ingeniero de Telecomunicación

Tutor:

Carlos A. Iglesias

Doctor Ingeniero de Telecomunicación

2020

Tribunal nombrado por el Magfco. y Excmo. Sr. Rector de la Universidad Politécnica de Madrid, el día _____ de _____ de _____.

Presidente: _____

Vocal: _____

Vocal: _____

Vocal: _____

Secretario: _____

Suplente: _____

Suplente: _____

Realizado el acto de defensa y lectura de la Tesis el día _____ de _____ de _____. en la E.T.S.I. Telecomunicación habiendo obtenido la calificación de _____.

EL PRESIDENTE LOS VOCALES

EL SECRETARIO

A mi familia y amigos.

Resumen

El objetivo principal de esta tesis doctoral es mejorar el análisis de sentimientos y emociones de texto en redes sociales, aunando técnicas de procesamiento de lenguaje natural, datos enlazados y análisis de redes sociales. La investigación se divide en tres partes muy diferenciadas.

Primero, se desarrolló un vocabulario semántico para describir emociones y procesos de análisis de sentimientos, alineado con la ontología de "procedencia" PROV-O. Este vocabulario permite seguir un enfoque de datos enlazados en el análisis de emociones, tanto en la anotación de recursos (datasets y lexicons), como en la publicación de servicios semánticos de análisis de emociones. Asimismo, se extendió el vocabulario de referencia para opiniones, Marl, para alinearlo con Prov-O.

En segundo lugar, se han modelado los diferentes componentes de los servicios de análisis de sentimientos y emociones, así como los requisitos para crear servicios abiertos, interoperables y que se puedan combinar para lograr análisis avanzados. El resultado es un marco de desarrollo y modelado de servicios, enfocado en la modularidad. Además, se ha desarrollado una implementación de referencia que permite a crear y publicar servicios de análisis de sentimientos y emociones.

En tercer lugar, se ha caracterizado el contexto social, que es el conjunto de información en una red social que complementa al mensaje, y que puede ser utilizado para mejorar el análisis de sentimientos del mensaje. También se ha desarrollado una taxonomía de enfoques de análisis de sentimientos basada en la forma en que el contexto social es construido y utilizado en el análisis. Seguidamente, se han investigado modelos de análisis de sentimientos que utilizan contexto social enriquecido mediante análisis de redes sociales.

Abstract

The main goal of this thesis is to improve sentiment and emotion analysis of text in social media through a combination of natural language processing, linked data and social network analysis. To achieve this goal, we have divided our research into three parts.

First, we developed a semantic vocabulary to describe emotions, emotion models and emotion analysis activities. This vocabulary enables a linked data approach to emotion analysis, including in the annotation and processing of resources (e.g., datasets and lexicons), and the development of public semantic emotion analysis services. We also extended the Marl vocabulary for opinions and sentiment to include concepts of sentiment analysis activities.

Secondly, we modeled the different components in a sentiment or emotion analysis service, as well as the requirements to create public and interoperable services that can be composed to produce advanced analyses. The result is a framework to model and develop modular services. We also developed a reference implementation of this framework, which can be used by researchers and developers to create and publish new sentiment and emotion analysis services.

Thirdly, we studied and formalized the concept of social context, which is the information in a social network that accompanies a text message and can be used to improve the analysis of said text. We also developed a taxonomy of approaches to sentiment analysis based on how they gather social context and how they exploit it in the analysis. In addition to characterizing social context, we investigated several models of sentiment analysis that enrich social context through social network analysis.

Contents

Reader's guide

This thesis is presented in the form of a compendium of publications (*compendio de publicaciones*). This means that this document is not a monograph, its core is a selection of publications by the author, which have been published in peer-reviewed journals and conferences and should support the quality and relevance of the results of this thesis. In particular, at least three of these publications must be published in journals in Q1 or Q2 positions of the JCR or Scopus ranking, and have the PhD student as first author. These requirements can be verified through list of publications in Annex A.

This form of thesis also imposes some additional requirements on the structure and content of the document. This section has been added to aid the reader navigate the document and evaluate its content. The document is structured as follows.

Chapter 1 is an introduction to this thesis, it explains the motivation, background and other relevant aspects. It also details the expected objectives and the initial hypotheses. The last section discusses how these objectives were met, and how the hypotheses were supported.

Chapter 2 explains the methodology used to conduct and evaluate the results of this thesis. It presents the phases in which research has been conducted. It also introduces the best practices followed for each type of result: definition of vocabularies, software, and other academic results. Lastly, it explains the evaluation criteria for each type of result.

Chapter 3 contains the full texts of novel publications that are core to this thesis. Each paper is presented in a separate section, where the full text is preceded by a short table with relevant information (title, authors, etc.), Publications are grouped into three categories, based on their topic: definition of vocabularies vocabularies and schemas, Linked Data for sentiment analysis, and social context for sentiment analysis. There are several tables of publications throughout the document, and all of them refer to the paper's section in this chapter. Some publications by the author have not been included in this chapter, as they do not directly contribute to the objectives. A complete list of relevant publications can be found in Annex A.

Chapter 4 is a general discussion about this thesis. Its different sections present an

overview of the solutions proposed, an analysis of the results, a summary of our conclusions from this work, and a discussion of future lines of research. The overview section (4.1) is aimed to connect all the contributions, and to help contextualize them. The results are presented in Section 4.2, and they are grouped by the objective (Section 1.4) they helped achieve. As a whole, this chapter is required to describe how each publication in Chapter 3 has contributed to this thesis, and their role to fulfil the objectives defined in Section 1.4. It should also justify the unity and coherence of the solutions and results.

The document makes use of several acronyms and abbreviations, all of which are summarized and explained in Section 4.5.

Lastly, a summary of all publications is available in Annex A, including those publications that are indirectly related to this thesis.

CHAPTER 1

Introduction

This chapter presents the context of this PhD thesis. This includes its motivation, relevant technical and social aspects, as well as its research objectives.

1.1 Motivation

With the rise of social media, more and more users are sharing their opinions and emotions online (Pang and Lee, 2008a). The increasing amount of information and number of users is drawing the attention of researchers and companies alike, which seek not only academical results but also profitable applications such as brand monitoring. As a result, many tools and services have been created to enrich or make sense of human generated content. Extracting sentiment from text is a branch of NLP known as sentiment analysis.

There are some terminological issues with terms such as "sentiment analysis" and "opinion mining", which are sometimes used interchangeably. This has been a long standing debate that predates the creation of online social networks. According to Pang and Lee, 2008a, when broad interpretations are applied, sentiment analysis and opinion mining denote the same field of study (which itself can be considered a sub-area of subjectivity analysis). For our purposes, we will use the term sentiment in its broadest sense, and the term affect as an umbrella in the rare occasions where the distinction between sentiment and opinion is important. We will also use the term sentiment analysis services, which are services that provide sentiment analysis of some sort, often over an HTTP protocol.

Terminological issues aside, sentiment analysis is broadly understood as the classification of text as positive or negative. In other words, polarity classification. Some works also introduce a "neutral" label, or finer grained categories such "very positive" or "mildly negative". Some other works use a continuous scale, and content is given a polarity value, often in the -1 to 1 range, although this varies from work to work. All these are variations over the same type of analysis.

This thesis focuses on different ways in which sentiment analysis can be improved, from incorporating new concepts and tools to improve usability, to leveraging sources other than text. As illustrated Figure 1.1, we will cover four types of improvement: the addition of Linked Data to represent both language resources and analysis services, the move from sentiment analysis to emotion analysis, the combination of analysis in text with other types of analysis (multi-modal analysis), and how more information about the social network can be leveraged.

It is important to note that these changes can be added independently, i.e., an analysis can use social context without dealing with emotions or other modalities, but it may as well use all the elements at the same time. We will see what each of these changes mean, why they are important for the future of sentiment analysis, and the challenges they impose.

So far sentiment analysis tools and services have used ad-hoc annotation schemata and

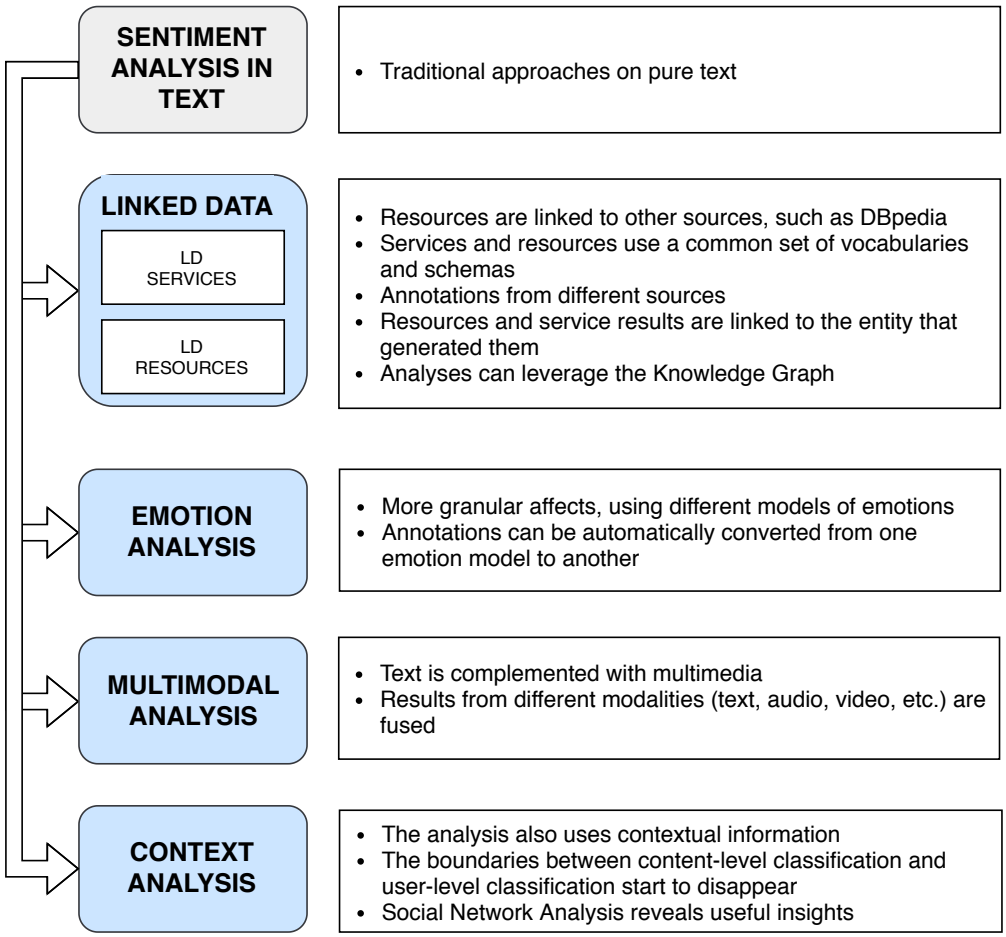


Figure 1.1: Overview of the evolution of sentiment analysis, from pure text to using contextual information.

services remain as isolated data silos. The lack of consensus is detrimental in several ways:

1. it impedes the creation of a homogeneous ecosystem of tools, libraries and applications;
2. it introduces ambiguity in the annotation (e.g., some services use a polarity scale of 0 to 1, whereas others use -1 to 1);
3. it hinders user adoption;
4. it makes evaluation of different services and tools hard.

These are strong barriers for the progress of the field.

Moreover, as more resources and applications appear, new types of sentiment analysis that account for more complex types of affects have started to emerge. For instance, some

works focus on classifying the emotion shown by a user in a specific text, such as happiness or surprise. Other works do stance classification (Pamungkas, Basile, and Patti, 2019), or detecting whether a person supports, denies, queries, or comments on a specific issue. These types of analysis can be seen as more complex forms of sentiment or subjectivity analysis, which pose whole new set of challenges.

In particular, in this thesis we focus on emotion analysis. Emotions have a crucial role in our lives, and even change the way we communicate (Pang and Lee, 2008a). They can be passed on just like any other kind of information, in what some authors call emotional contagion (Barsade, 2002). That is a phenomenon that is clearly visible in social networks (Kramer, Guillory, and Hancock, 2014), and it can be observed in Online Social Network (OSN) through public APIs make it relatively easy to study the social networks and their information flow.

Some social sites are already using emotions natively, giving their users the chance to share emotions or use them in queries. This is exemplified by Facebook, which recently updated the way its users can share personal statuses. And there are some tools and languages to annotate different types of media with emotions, such as EmotionML. However, there is no common semantic vocabulary for emotions.

On the other hand, natural language is often mixed with other types of information that can be analyzed. For instance, a video of a speech has at least three different elements that can be analyzed using different techniques: 1) the script or words can be analyzed using NLP; 2) the voice is analyzed using audio analysis; 3) the posture and gestures can be analyzed using video analysis. The combination of all these analyses is known as multimodal analysis, and it is an active field that gathers experts from different disciplines. Each type media lends itself to different types of analysis. In fact, one of the reasons emotion detection in text far less popular than polarity classification may be that it is easier to detect the valence (positive or negative) of an opinion in text, than it is to detect other dimensions of an emotion, such as the arousal (high intensity level) of the author.

Emotion analysis is more common in audio than in text, as the opposite occurs: it is harder to detect valence than arousal or dominance. Combining different modalities can help counter the weaknesses of each modality, but it can be very challenging, as each field has traditionally worked in isolation, using different tools, naming conventions and models. This leads to technical challenges, such as dealing with different tools and formats, and some conceptual challenges, such as having different emotion models. The technical challenges are mostly caused by the lack of communication between communities, and they can be remedied through tooling and using linked data vocabularies as lingua franca. Conceptual

differences such as model heterogeneity are trickier to solve. Some of these differences stem from the lack of a standard affect model, and the fact that certain models are more suited to specific modalities. As we said, it is easier to detect arousal (high intensity levels) from audio than it is to detect valence (polarity). This may be a reason why the speech analysis community has used dimensional models such as the Valence-Arousal-Dominance model, whereas the text community has relied on positive, negative and neutral tags. Some types of analysis may require converting all annotations to a common model, which is not trivial.

Lastly, despite the advantages of multi-modal analysis, simply analysing the content of a message is still not enough to confidently infer the sentiment of the author. When humans communicate, there is a plethora of information that is assumed or implicitly known, such as known political affiliations of each interlocutor, history of comments, or acquaintances. Social media content relies heavily on this type of information, due to its informal tone and the brevity of the interactions. If this information could be modelled and exploited, it may improve sentiment analysis in social media, as shown by some related works that have already shown promising results.

We have thus identified four shortcomings of the state of the art in sentiment analysis in text: 1) lack of interoperability, i.e., heterogeneity of formats and schemas; 2) underrepresentation of emotion analysis; 3) difficulty to integrate with other types of analysis; and 4) disregard for contextual information. To target interoperability, Linked Data can be used as a lingua franca for data representation as well as a set of tools to process and share such information. Plenty of services have embraced the Linked Data concepts and are providing tools to interconnect the previously closed silos of information (Tummarello, Delbru, and Oren, 2007). It has even proven useful in some areas of Opinion Mining. Some schemata offer semantic representation of opinions (Westerski, Carlos A. Iglesias, and Tapia, 2011), allowing richer processing and interoperability. The same could be applied to sentiment analysis as a whole, and to Emotion Analysis, once there is a proper model for emotions. In addition to vocabularies and schemas, achieving real service interoperability would further require common APIs and methodologies. Once a proper model for emotions is in place, new resources and examples for emotion analysis should become available, thus increasing the popularity of emotion analysis. A Linked Data principled approach would also help with multimodal analysis by providing common models for all modalities, modelling all the transformations necessary to fuse different modalities, and helping track provenance information.

In summary, sentiment and emotion analysis could benefit of the combination of a linked data principled approach, multi-modal analysis, and leveraging contextual information. The

first one has the potential to enable novel applications, interoperability of tools and services, and in general would foster research and applications in the field. The last two may be used to improve the performance of sentiment and emotion classification. This thesis aims to tackle these issues, effectively modelling and linking affects (sentiment and emotion), data (knowledge graph), and people (context in the social network).

1.2 Background

1.2.1 Sentiment and Emotion Analysis

Sentiment analysis has been an active field of research for a long time, but it has grown in popularity with the advent of online opinion-rich resources (Pang and Lee, 2008b). This new type of content comes with new challenges and limitations. The quintessential example is how microblogging sites such as Twitter impose hard limitations on text extension. In the case of Twitter, that limitation has only been recently lifted from the iconic 140 characters. Due to the limitations and the intended audience, those texts also tend to be more relaxed and make heavy use of slang and abbreviations. In contrast, prior to the creation of these OSNs, sentiment analysis dealt mostly with longer and formal texts, such as those in news articles, papers and books. A great deal of the research in recent years has gone into finding ways to combat those challenges. In turn, these resources have also added their own set of limitations and challenges.

Over the last two decades, numerous works have explored different approaches to sentiment analysis. These approaches can be grouped into machine learning, lexicon based, and hybrid (K. Ravi and V. Ravi, 2015). Of the three, machine learning techniques and hybrid approaches seem to be dominant (Araque et al., 2017; Pang, Lee, and Vaithyanathan, 2002; Wang and Manning, 2012), and lexicon techniques are typically incorporated into machine learning approaches to improve their results. Machine learning approaches apply a predictor (a classifier, or an estimator) on a set of features that represent the input. The set of predictors is not very different from those used in other areas. Instead, the complexity in these approaches lies in extracting complex features from the text, filtering only relevant features, and selecting a good predictor (Sharma and Dey, 2012).

One of the most straightforward features is the Bag Of Words (BOW) model. In BOW, each document is represented by the multiset (bag) of its constituent words. Word order is disrupted, and syntactic structures are broken. As a result, a great deal of information from natural language is lost (Xia and Zong, 2010). Therefore, various types of features have been exploited, such as higher order n-grams (Pak and Paroubek, 2010). A more sophisticated

feature is Part of Speech (POS) tagging (Gimpel et al., 2011). In it, a syntactic analysis process is run, and each word is labeled (tagged) with its syntactic function (e.g., noun). Additionally, syntactic trees can be calculated. Using these trees, the words in the input can be rearranged to a more convenient position while still conveying the same meaning. Note how these two types of features only rely on lexical and syntactical information. For this reason, they are sometimes referred to as surface forms.

Surface forms can also be combined with other prior information, such as word sentiment polarity (García-Pablos, Cuadros Oller, and Rigau Claramunt, 2016; Cambria, 2016; Kiritchenko, Zhu, and Mohammad, 2014; Melville, Gryc, and Lawrence, 2009; Nasukawa and Yi, 2003). This prior knowledge usually takes the form of sentiment lexicons, i.e., dictionaries that associate words in a domain or language with a sentiment. Some lexicons also include non-words such as emoticons (Jiang et al., 2015; Hogenboom et al., 2015) and emoji (Novak et al., 2015). These alternative forms of writing have been shown very useful, as they can dominate textual cues and form a good proxy for text polarity (Hogenboom et al., 2015).

The use of lexicon-based techniques has many advantages (Taboada et al., 2011), most of which stem from their combination with other methods. For instance, it is possible to generate lexicons that are domain dependent or that incorporate language-dependent characteristics. Lexicons and syntactic information can also be combined with linguistic context to shift valence (Polanyi and Zaenen, 2006). On the other hand, there are several disadvantages to lexicon approaches. First, creating lexicons is an arduous task, as it needs to be consistent and reliable (Taboada et al., 2011). It also needs to account for valence variability across domains, contexts, and languages. These dependencies make it hard to maintain domain-independent lexicons. An alternative to retain independence while encoding domain, language, and context variability is through semantic representation of the lexical resources in the form of ontologies. An ontology can encode both lexical (McCrae, Spohr, and Cimiano, 2011b) and affective (Sánchez-Rada and Carlos A. Iglesias, 2016) nuances, both in the lexicons and in the automatic annotations (Buitelaar, Arcan, et al., 2013). This is especially useful for aspect-based sentiment analysis, as the differences between aspects can be incorporated into the ontology (Wei and Gulla, 2010).

In recent years, new approaches based on deep learning have shown excellent performance in Sentiment Analysis (Collobert et al., 2011; Bengio, 2009). In contrast with traditional techniques, deep learning techniques learn complex features from data with minimum human interaction. These algorithms do not need to be passed manually crafted features: they automatically learn new complex features. The downside is that the quality of the features

heavily depends on the size of the training data set. Hence, they often require large amounts of data, which is not always available. They also raise other concerns such as interpretability (Marcus, 2018; Lipton, 2016) or its inability to adapt to deal with edge cases (Marcus, 2018). In the realm of NLP, most of the focus is on learning fixed-length word vector representations using neural language models (Kim, 2014). These representations, also known as word embeddings, can then be fed into a deep learning classifier, or used with more traditional methods. One of the most popular approaches in this area is word2vec (Mikolov et al., 2013). The downside of these methods is that they require enormous amounts of training data. Luckily, several researchers have already applied these methods to large corpora such as Wikipedia and released the resulting embeddings.

Lastly, it is also possible to combine independent predictors to achieve a more accurate and reliable model than any of the predictors on their own. This approach is known as ensemble learning. Many ensemble methods have been previously used for sentiment analysis. Ensemble methods can be classified according to two main dimensions Rokach (2010): how predictions are combined (rule-based and meta-learning), and how the learning process is done (concurrent and sequential). A new application of ensemble methods is the combination of traditional classifiers based on feature selection and deep learning approaches (Araque et al., 2017).

1.2.2 Linked Data for NLP

Tim Berners-Lee, 2006 outlined a set of rules for publishing data on the Web in a way that all published data becomes part of a single global data space:

1. Use URIs as names for things
2. Use HTTP URIs so that people can look up those names
3. When someone looks up a URI, provide useful information, using the standards(RDF, SPARQL)
4. Include links to other URIs, so that they can discover more things

In Berner-Lee's own words, rather than rules these are actually expectations of behavior: breaking them does not destroy or break any contract. But by following them we seize the opportunity to data interconnected. Hence, these rules have since been known as the Linked Data principles In general, Linked Data Bizer, Heath, and Berners-Lee, 2009 refers to best practices and technologies for publishing, sharing and connecting structured data on the web.

What is missing from those rules is how the useful information should be organized. It leaves that side to Resource Description Framework (RDF) and other standards. RDF is a standard for establishing semantic interoperability on the Web Decker et al., 2000. It is similar to XML in many aspects. Except that XML only addresses document structure, i.e., the hierarchical relation of different elements, and their attributes. By contrast, RDF provides a data model that can be extended to provide more sophisticated ontological representations. This flexibility means that RDF better facilitates interoperability.

At its core, the RDF model is very simple. It is based on subject-predicate-object expressions. For instance, let us consider the the expression "I studied at UPM". Here, the subject is "I", the predicate is "studied at", and the object is "UPM". For all three, we would need a URI, or unique identifier. As their name implies, these URIs must be unique to each specific concept, and they should be addressable to comply with the linked data principles. We will use an invalid domain as base, and the following full URIs: `http://example.com/jfernando`, `http://example.com/studied-at` and, lastly `http://example.com/UPM`. By the same token, we may add new predicates and URIs. For instance, the example in Listing 1.1 encodes the information: my supervisor and I studied at UPM.

Listing 1.1: Example of RDF in n-triples notation

```
<http://example.com/jfernando> <http://example.com/studied-at> <http://example.com/
    UPM> .
<http://example.com/jfernando> <http://academia.test/supervised-by> <http://example.
    com/cif> .
<http://example.com/cif> <http://example.com/studied-at> <http://example.com/UPM> .
```

Writing all triples explicitly is very cumbersome and it reduces readability. RDF has other representation formats that are better for human consumption, such as Turtle (Listing 1.2). These formats can also use additional features such as prefixes, to make the annotation even easier to read and write.

Listing 1.2: Example of RDF in turtle notation

```
@prefix ex: <http://example.com/> .
@prefix academia: <http://academia.text/> .

ex:jfernando ex:studied-at ex:UPM ;
             academia:supervised-by cif.

cif ex:studied-at ex:UPM .
```

An interesting alternative format is JSON-LD[1], a subset of JSON that incorporates se-

[1] `http://json-ld.org`

mantics. JSON-LD was designed as a lightweight Linked Data format for human consumption and creation. With other RDF representation formats, new tools, standards (SPARQL) and software (e.g., databases) are required to store and query data. Since JSON-LD is fully compatible with all the JSON ecosystem, it can be used anywhere JSON is. This makes it an ideal data format for programming environments, REST Web services, and unstructured databases such as CouchDB and MongoDB. The previous examples could be represented in JSON-LD as Listing 1.3. The semantics of each property and value are provided by the JSON-LD context, which can be re-used and referenced by URL, such as in the example. In our example, the context is shown in Listing 1.4.

In contrast with the graph-based RDF model, JSON uses a hierarchical model-based structure. This means that there are different ways (i.e., schemas) to represent an RDF graph in JSON-LD form. For each situation, some of those schemas are more verbose, and some are easier to use. Part of the challenges in defining a model for sentiment and emotion analysis will lie in selecting a schema that is ergonomic while retaining all the necessary information.

Listing 1.3: Example of RDF in JSON-LD notation

```
{ "@context": "http://example.com/context.jsonld",
  "@id": "jfernando",
  "studied-at": "UPM",
  "supervised-by": {
      "@id": "cif",
    "studied-at": "UPM"
}
```

Listing 1.4: Context for Listing 1.3

```
{
  "@context": {
    "@base": "http://example.com",
    "supervised-by": {
      "@id": "http://academia.test/supervised-by"
    },
    "studied-at": {
      "@type": "@id",
      "@id": "http://example.com/studied-at"
    }
  }
```

```
}
```

So far, we have defined our own URIs and predicates. In other words, we have chosen our representation model for our example domain. This is fine for an ad-hoc case, but one of the advertised advantages of Linked Data is the ability to connect different data sources. If different data sources use very different predicates and URIs for the same concepts, we will not be able to merge their information without intervention. Consequently, we need a way to agree on a set of common predicates and entities that are valid. These models are also referred to as ontologies, vocabularies, or specifications. There are different types of vocabularies. The simplest are just a list of predicates and entities. Some are expressed in rich languages such as Web Ontology Language (OWL) and allow for a fine-grain definition of the domain, which can later be used for reasoning and inference. These ontologies add useful concepts such as classes, subclasses, properties, ranges and domains. Back to our example, we may add classes to our vocabulary such as `PhD student`, and properties such as `date of defense`. We may also include restrictions like *people can only study at a university*, or *people can only be supervised by full professors*. This could be used to check for incoherent data (e.g., if `ex:cif` was listed as a fellow student (not a professor)), or to infer missing data.

To foster reusability and composability, each vocabulary is expected to cover one domain, and to rely on others to model aspects outside of that domain. For instance, in our example we could add my e-mail address re-using properties from other vocabularies such as Friend of a Friend (FOAF). An example of vocabulary that is very widespread on the internet is schema.org[2], which covers different domains that are useful from a web search engine, such as venues, online reviews and people.

Rather than creating an ad-hoc model for each domain, linked data principles encourage reusing already existing models. This thesis covers three different domains that need to be modelled with linked data: 1) language resources; 2) sentiment and emotion analysis services. 3) sentiment and emotions (more generally, affects);.

Linked data technologies have gained wide acceptance in the realm of language resources, as a mechanism to combine heterogeneous resources. Before the use of linked data, several initiatives had addressed interoperability of language resources since the late 1980s such as Text Encoding Initiative (TEI) (Ide and Veronis, 1995), but there was not yet a widely accepted global solution for integrating and combining heterogeneous linguistic resources from different sources (Chiarcos, 2013). In particular, the Linked Open Data (LOD) Project

[2] `http://schema.org`

is a grassroots community effort supported by W3C whose aim is to bootstrap the Web of Data by identifying existing datasets available under open licenses, convert them to RDF following the Linked Data principles, and to publish them on the Web. The data cloud originated from this initiative is known as the LOD Cloud. Several communities such as Open Linguistics Working Group (OWLG) (Chiarcos, 2013) proposed the idea of adopting linked data principles for representing, sharing and publishing open linguistic resources with the aim of developing a sub-cloud of LOD cloud of linguistic resources, known as the Linguistic Linked Open Data (LLOD) cloud (Chiarcos, Hellmann, and Nordhoff, 2011a).

In addition, the use of linked data for modeling linguistic resources provides a clear path to their semantic annotation and linking with semantic resources of the Web of Data. This is especially important for making sense of social media streams whose semantic interpretation is particularly challenging, because they are strongly inter-connected, temporal, noisy, short, and full of slang (Bontcheva and Rout, 2012). Moreover, several authors (Saif, He, and Alani, 2012) have shown that the use of semantics in sentiment analysis outperforms semantics-free methods. Thus, the availability of semantically annotated linguistic resources is a crucial to the development of the field of sentiment analysis.

Similarly, the NLP community has recently started to use alternatives to model NLP processes as linked data, and to achieve interoperability between different NLP tools and services. The most popular initiative in this area is NLP Interchange Format (NIF). NIF 2.0 (Hellmann et al., 2013) defines a semantic format and API for improving interoperability among natural language processing services. NIF follows a linked data principled approach so that different tools or services can annotate a text. To this end, texts are converted to RDF literals and an URI is generated so that annotations can be defined for that text in a linked data way. NIF offers different URI Schemes to identify text fragments inside a string, e.g. a scheme based on RFC5147 (Wilde and Duerst, 2008), and a custom scheme based on context.

Thus, NIF is a natural fit for sentiment and emotion analysis in text. But, for that, NIF needs to be extended to include specific parts of Sentiment and Emotion Analysis, such as a description of Opinions and Emotions. That is the role of vocabularies such as Marl.

Marl (Westerski, Carlos A. Iglesias, and Tapia, 2011) is a vocabulary for Opinion Mining. It was designed to annotate and describe subjective opinions expressed in text. In essence, it provides the conceptual tools to annotate Opinions and results from Sentiment Analysis in an open and sensible format. However, it is focused on polarity extraction and is not capable of representing Emotions.

LLOD (Chiarcos, Hellmann, and Nordhoff, 2011b; Chiarcos, McCrae, et al., 2013) is an

initiative that promotes the use of linked data technologies for modeling, publishing and interlinking linguistic resources. The main benefit of using linked data principles to model linguistic resources is that it provides a graph-based model that allows representing different kinds of linguistic resources (such as lexical-semantic resources, linguistic annotations or corpora) in a uniform way, thus supporting querying across resources. The creation of an LLOD cloud is a cooperative task that is been managed by several communities, such as OWLG of the Open Knowledge Foundation and Ontology-Lexica Community Group (OntoLex) of W3C. As a result of this activity, an initial LLOD is currently available, where several types of resources have been identified: lexical-semantic resources (e.g. machine readable dictionaries, semantic networks, semantic knowledge bases, ontologies and terminologies), annotated corpora and linguistic annotations.

With regards to modeling lexical-semantic resources, we find *Lexicon Model for Ontologies (lemon)* (Buitelaar, Cimiano, et al., 2011). The lemon vocabulary proposes a framework for modeling and publishing lexicon and machine-readable dictionaries as linked data. It also provides a bridge between the most influential lexical-semantic resources, WordNet (Fellbaum, 1998) and DBPedia (Bizer, Lehmann, et al., 2009). Lemon was designed to meet the following challenges:

- RDF-native form to enable leverage of existing Semantic Web technologies (SPARQL, OWL, RIF etc.).
- Linguistically sound structure based on LMF to enable conversion to existing offline formats.
- Separation of the lexicon and ontology layers, to ensure compatibility with existing OWL models.
- Linking to data categories, in order to allow for arbitrarily complex linguistic description. In particular the LexInfo vocabulary is aligned to lemon and ISOcat.
- A small model using the principle of least power - the less expressive the language, the more reusable the data.

With regards to annotated corpora, there are two initiatives (Chiarcos, Hellmann, and Nordhoff, 2012), POWLA (Chiarcos, 2012b) and NIF (Hellmann, 2013) that enable to link lexical-semantic resources to corpora. Finally, Ontologies of Linguistic Annotation (OLiA) (Chiarcos, 2012a) are a repository of annotation terminology for various linguistic phenomena that can be used in combination with POWLA, NIF or *lemon*. OLiA ontologies

allow to represent linguistic annotations in corpora, grammatical specifications in dictionaries, and their respective meaning within the LLOD cloud in an operable way.

The main benefits of modeling linguistic resources as linked data include (Chiarcos, McCrae, et al., 2013) interoperability and integration of linguistic resources, unambiguous identification of elements of linguistic description, unambiguous links between different resources, possibility to annotate and query across distributed resources and availability of mature technological infrastructure.

Another key aspect of the semantic models in this thesis is provenance. Provenance is information about entities, activities, and people involved in producing a piece of data or thing, which can be used to form assessments about its quality, reliability or trustworthiness. The PROV Family of Documents defines a model, corresponding serializations and other supporting definitions to enable the inter-operable interchange of provenance information in heterogeneous environments such as the Web (Moreau et al., 2011). It includes a full-fledged ontology that other ontologies can link to. The complete ontology is covered by the PROV-O Specification (Groth and Moreau, 2013). In essence, Agents take part in Activities to transform Entities (data) into different Entities (modified data). This process can be aggregation of information, translation, adaptation, etc. For the purpose of this thesis, this activity is usually a sentiment or emotion analysis, which turns plain data into semantic sentiment or emotion information. There are many advantages to adding provenance information in emotion analysis, as different algorithms may produce very different results that can then be compared or aggregated.

1.2.3 Emotion Representation

Emotion is a very common concept that gets used often in everyday life, it is ingrained in our vocabulary. The term usually gets used in a very loose and subjective way. For emotion analysis, and any other kind of research on affective computing, we need a more rigorous definition of emotion, including what emotions are possible, and how they can be measured and represented. We refer to that definition as a model of emotions, or emotion model. As it turns out, modelling emotions is a rather complex task. In this section we will cover the most popular attempts to propose a universal model of emotions. We will also cover some alternative ways to represent emotions in a machine-readable way.

Various representation schemes for emotions have been proposed over the years, each based on particular criteria, ranging from the most simplistic and ancient that come from Chinese philosophers to the most modern theories that refine and expand older models (Ekman, 1999; Prinz, 2004). The literature on the topic is vast, and it is out of the scope of

this section to cover all these models. Other works have presented a detailed overview of different emotion models and their use in affective computing (Cambria, Livingstone, and Hussain, 2012). Overall, there are two main types of models: models based on categories, and dimensional models. In the former, emotions are represented with one or more categories or labels (e.g., happy, sad). In practice, in addition to categories there may also be other numeral values, such as value (i.e., how strong the emotion is), and confidence of the annotation. On the other hand, dimensional models represent emotions in the continuum of an n-dimensional space. The number of dimensions varies from model to model. Additionally, some models label specific regions of that space.

One of the most popular categorical models, even in popular culture, is Ekman's model of *six basic emotions* (Anger, Fear, Surprise, Happiness, Disgust, Sadness), which is based on the universality of those emotions (Ekman and Friesen, 1971). The advantage of Ekman's model is its simplicity. At the same time, other researchers have found this simplicity too limiting to represent the wealth of human emotions. For instance, Plutchik's *wheel of emotion* is an alternative to Ekman's model that provides more categories of emotion. It is based on contrast and closeness of emotions (Plutchik, 1980a), and the categories are arranged in concentric circles of growing complexity. At its root, there are eight basic basic emotions: anger, fear, sadness, disgust, surprise, anticipation, trust, and joy. All other emotions are derived from those. Plutchik's model has been extensively used (Borth et al., 2013; Cambria, Havasi, and Hussain, 2012) in the area of Sentiment Analysis and Affective Computing, relating all the different emotions to each other in what is called the rose of emotions. Other categorical models cover affects in general, which include Emotions as part of them. One of them is the work by Strapparava and Valitutti, WordNet-Affect (Strapparava and Valitutti, 2004). It comprises more than 300 affects, many of which are classified as emotions. What makes this categorisation interesting is that it effectively provides a taxonomy of emotions. It both gives information about relationship between emotions and makes it possible to to decide the level of granularity of the emotions expressed. Lastly, the recent work by Cambria et al. (Cambria, Livingstone, and Hussain, 2012) introduces a novel model, The Hourglass of Emotions, inspired by Plutchik's studies (Plutchik, 1980b).

On the dimensional front, we will cover three closely-related models. First, Russel's *Circumplex model* is constructed to capture the core affect in a two dimensional (Arousal and Valence) model (Russell, 2003; Posner, Russell, and Peterson, 2005). Arousal reflects the level of energy in the emotion (e. g., pleased vs. ecstatic), whereas valence reflects the hedonic tone (e. g., pleasant vs. unpleasant). Osgood later identified an additional dimension, and the result is the very widespread PAD (Pleasure, Arousal, Dominance) three-dimensional representation (Osgood, Suci, and Tannenbaum, 1957). Dominance represents the sense of

control or dominant nature of the emotion (e. g., fear vs. anger). More recently, Fontaine et al. identified a fourth dimension (unpredictability) (Fontaine et al., 2007). This new dimension refers to the appraisal of expectedness or familiarity.

Despite the efforts put into these models, there does not seem to be a universally accepted model for emotions (Schröder, Pirker, and Lamolle, 2006). This poses a problem for any field of emotion research, but it is even more troubling for affective computing, where systems need a representation format for emotions. Emotion Markup Language (EmotionML) (Burkhardt et al., 2013) is one of the most notable general-purpose emotion annotation and representation languages. It was born from the efforts made for Emotion Annotation and Representation Language (EARL) (Excellence, 2006; Schröder, Pirker, and Lamolle, 2006) by Human-Machine Interaction Network on Emotion (HUMAINE).

In a discussion regarding EmotionML, Schroder et al. pose that *any attempt to propose a standard way of representing emotions for technological contexts seems doomed to fail* (Schröder, Devillers, et al., 2007). Instead they claim that the markup should offer users choice of representation, including the option to specify the affective state that is being labelled, different emotional dimensions and appraisal scales. The level of intensity completes their definition of an affect in their proposal. As a result, EmotionML offers twelve vocabularies for categories, appraisals, dimensions and action tendencies. A vocabulary is a set of possible values for any given attribute of the emotion. There is a complete description of those vocabularies and its computer-readable form available (Ashimura et al., 2012).

There have also been some efforts to provide a semantic vocabulary for emotions. First, the Chinese Emotion Ontology (Yan et al., 2008) was developed to help understand, classify and recognize emotions in Chinese. The ontology is based on HowNet, the Chinese equivalent of WordNet. The ontology provides 113 categories of emotions, which resemble the WordNet taxonomy and the authors also relate the resulting ontology with other emotion categories. All the categories together contains over 5000 Chinese verbs. Soon after, Grassi presented Human Emotion Ontology (HEO) (Grassi, 2009). This ontology presents an ontology for human emotions for its use for annotating emotions in multimedia data. HEO deals with the heterogeneity of emotional theories by providing a generic characterisation of emotions. There are several limitations with HEO. First of all, despite being a generic ontology, HEO embeds the most common models (Plutchik, Ekman, Hourglass, etc.) in the ontology itself. Moreover, its representation of action tendencies and appraisal as entities, instead of properties, makes its use for dimensional annotation very verbose. It also includes some properties (e.g. *isAnnotatedBy*) that, while useful for certain types of Emotion Analysis, are not generic or there are other established ontologies that provide the

same concept. And it makes assumptions about certain properties, which limits its use. For instance, automated emotion analysis would not have a human annotator, as HEO suggests.

Another work worth mentioning is that of Hastings et al. (Hastings et al., 2011) in Emotion Ontology (EMO), an ontology that tries to reconcile the discrepancies in affective phenomena terminology. It is, however, too general to be used in the context of emotion analysis: it provides a qualitative notion of emotions, when a quantitative one would be needed.

1.2.4 Contextual information for sentiment analysis

Contextual information refers to the collection of users, content, relations, and interactions which describe the environment in which social activity takes place. It encapsulates the frame in which communication in social media takes place. Since this is a new field of research, the definition and use of this contextual information for sentiment analysis was rather vague. This makes it hard to describe or compare works of the state of the art. One of the contributions of this thesis is a formal definition of social context. For the sake of clarity, we will use that terminology throughout the thesis.

Social context is used in sentiment analysis for two reasons that are subtly different. First, it can be used to compensate for implicit elements in the text. An example of this is how slang, abbreviations or semantic variations can be detected and accounted for in the classification. Humans apply a similar process when trying to understand content. Content authors also unconsciously rely on this fact and they assume certain prior knowledge. The second motivation to add social context is that it may help correct ambiguity or situations where textual queues are lacking. For example, a classifier may use the sentiment of earlier posts by the user and similar users on the same topic.

Tan et al. (2011) is one of the first works to incorporate social context information, which the authors called heterogeneous graph on topic, to infer (user) sentiment. The underlying ideas behind that work are user consistency and homophily. A function to measure each of those attributes is provided, and the model tries to maximize the overall value. The authors compare alternative ways to construct the user network, using variations of follower-followee relations and direct replies (interactions). In their results, relations and interactions yield similar results. In the original formulation edges (relations or interactions) are not weighted, so users are influenced equally by all their neighbors. Interactions are bound to be noisy, and aggregating them in this fashion is likely to provide little or no advantage over a simple relation. The SANT model (X. Hu et al., 2013) follows similar ideas but for content classification. It also combines sentiment consistency, emotion contagion and a

unigram model in a classifier.

Pozzi et al. (2013) extended the model by Tan et al. (2011). Their model uses what they call an approval network, which effectively add weights for edges between users. The rationale for that change is that friendship does not imply approval, and that a weighted network of interactions should better capture emotion contagion.

Other models have also exploited other strategies such as community detection in their analysis. An example is Xiaomei et al. (2018), which incorporate weak dependencies between microblogs, using community detection (different algorithms) on a network of microblogs. In their work, microblogs are connected if their authors are (i.e., there is a follower-followee relation). They refer to connections inferred from community detection as *weak connections*.

1.2.5 Social Network Analysis and Community Detection

Social Network Analysis (SNA) is the investigation of social structures Otte and Rousseau, 2002. It provides techniques to characterize and study the connections between people, and their interactions. SNA is not limited to OSN, but to any kind of social structure. Other examples of social network would be a network of citations in publications or a network of relatives. Through SNA techniques, it is possible to extract information from a social network that may be useful for sentiment analysis, such as chains of influence between users, groups of like-minded users, or metrics of user importance.

There are several ways in which SNA techniques can be exploited in sentiment analysis, but most of them fall under one of two categories: those that transform the network into metrics or features that can be used to inform a classifier; and those that limit the analysis to certain groups or partitions of the network.

A simple example of metrics provided by SNA could be user's follower in-degree (number of users that follow the user) and out-degree (number of users followed by the user), which could be used as features for each user (Sixto, Almeida, and López-de-Ipiña, 2018). However, these metrics are not very rich, as they only cover users directly connected to a user, and it does so in a very naive way: all connections are treated equally. Other more sophisticated metrics could be used instead of in/out-degree, such as centrality, a measure of the importance of a node within a network topology, or PageRank, an iterative algorithm that weights connections by the importance of the originating user. Several works have introduced alternative metrics for user and content influence in a network (Hajian and White, 2011; Noro and Tokuda, 2016).

The second category of approaches is what is known either as network partition or

as community detection, depending on whether the groupings may overlap. Intuitively, community detection aims to find subgroups within a larger group. This grouping can be used to inform a classifier, or to limit the analysis to relevant groups only. More precisely, community detection identifies groups of vertices that are more densely connected to each other than to the rest of the network (Papadopoulos et al., 2012). The motivation is to reduce the network into smaller parts that still retain some of the features of the bigger network. These communities may be formed due to different factors, depending on the type of link used to connect users, and the technique used to detect the communities. Each definition has its own set of characteristics and shortcomings. For instance, if users are connected after messaging each other, community detection may reveal groups of users that communicate with each other often (Deitrick and W. Hu, 2013). By using friendship relations, community detection may also provide the groups of contacts of a user (Gao et al., 2012).

The reader is referred to other publications (Papadopoulos et al., 2012; Orman, Labatut, and Cherifi, 2011) for further details of the different definitions of community and algorithms to detect them.

1.3 Hypotheses

After taking into consideration the background and context of this thesis, we formulated the following hypotheses:

- **Hypothesis-1** A Linked Data approach to sentiment analysis would increase interoperability between services and enable advanced capabilities such as automatic evaluation
- **Hypothesis-2** A semantic vocabulary for emotions would ease multi-modal analysis and enable the use of different emotion models
- **Hypothesis-3** Sentiment of social media text can be predicted using additional contextual information (e.g., previous history and relations between users)

1.4 Objectives

In order to test the hypotheses in the previous section, we set the following objectives:

- **Objective-1** Definition of a vocabulary for emotions

  The vocabulary should be applicable to different use cases, including language resources (lexica and corpora) and analysis services.

- **Objective-2** Definition of a model to annotate language resources and to be used in analysis services

  The model should leverage existing vocabularies such as Marl and lemon, and provide a unified representation for both language resources and services.

- **Objective-3** Definition of a reference architecture for sentiment and emotion analysis services

  The architecture will be based on the definition of a REST-ful resource model, in order to facilitate the development of services. The architecture should be adaptable to a Big Data scenario.

- **Objective-4** Development of a reference implementation of the architecture

  The reference implementation will be valuable to evaluate the soundness of the architecture.

- **Objective-5** Modelling the types of contextual information and social theories (such as emotion propagation) that can be leveraged for sentiment and emotion analysis

  This may require the use of additional models, such as social theories of behavior and group formation, or techniques from other fields, such as Social Network Analysis.

1.5 Document outline

The rest of the document is structured as follows: Chapter 2 explains the methodology used to conduct and evaluate the results of this thesis, the phases of this research, best practices followed and evaluation criteria for each type of result; Chapter 3 contains the full texts of novel publications that are core to this thesis; Each paper is presented in a separate section, where the full text is preceded by a short table with relevant information (title, authors, etc.); Publications are grouped into three categories, based on their topic: definition of vocabularies vocabularies and schemas, Linked Data for sentiment analysis, and social context for sentiment analysis; Chapter 4 is a general discussion about this thesis; It presents an overview of the solutions proposed, an analysis of the results, a summary of our conclusions and a discussion of future lines of research. Lastly, a summary of all publications is available in Annex A, including those publications that are indirectly related to this thesis.

CHAPTER 2

Methodology

In layman's terms, research is 'finding out something you don't know.' Luckily, the actual act of research is much less vague. It needs to be methodical, focused and rigorous. This chapter presents the methodological considerations to ensure this thesis follows those principles.

2.1 Introduction

There are two ways to conduct research, using induction or deduction. The first way proposes generalizations based on observations. For this camp, the cornerstone of research is the examination of the adequacy of generalizations, formulated as hypotheses (Phillips, Pugh, et al., 2010)[1]. The deductive approach uses known theory to propose hypotheses, which are then tested. A proponent of this view was Karl Popper who claimed that the nature of scientific method is hypothetico–deductive and not, as is generally believed, inductive. When conducting a PhD thesis, it is important to understand the difference between these two interpretations of the research process to avoid discouragement, and suffering from a feeling of 'cheating' or not going about it the right way.

A popular misconception about scientific method is that it is inductive: that the formulation of scientific theory starts with the basic, raw evidence of the senses – simple, unbiased, unprejudiced observation. Out of these sensory data - commonly referred to as 'facts' – generalizations will form. The myth is that from a disorderly array of factual information an orderly, relevant theory will somehow emerge. However, the starting point of induction is an impossible one.

There is no such thing as unbiased observation. Regardless of how hypotheses arise, either guesswork or by inspiration, they can and must be tested rigorously, using the appropriate methodology. If the predictions you make as a result of deducing certain consequences from your hypothesis are not shown to be correct then you must discard or modify your hypothesis. If the predictions turn out to be correct then your hypothesis has been supported and may be retained until such time as some further test shows it not to be correct. This thesis concerns two types of research: **problem-solving**, and **exploratory**.

Problem-solving research starts from a particular problem, and it applies current knowledge to it, usually from different fields of expertise. The challenge in this type of research comes from rigorously defining the problem, and the method in which the solution will be sought. The use of Linked Data for sentiment and emotion analysis services and resources falls under this category. It is defined in terms of shortcomings of current approaches, and its hypotheses are formulated around real-world outcomes more so than on academic achievements.

On the other hand, we have exploratory research, where little is known about the problem, This type of research will need to examine what theories and concepts are appropriate,

[1]To properly introduce the scientific methodology, we have borrowed heavily from Phillips, Pugh, et al., 2010, which we encourage any aspiring PhD student to read.

developing new ones if necessary, and whether existing methodologies can be used. It involves pushing the frontiers of knowledge in the hope that something useful will be discovered The investigation of contextual information falls under this type of research, as it is a novel field of study, which combines several research areas. In this specific case, there is also a very precise end: improving sentiment and emotion analysis.

The following sections describe the phases in which we have organized our research, as well as the specific methodological considerations for each type of work (e.g., defining semantic vocabularies).

2.2 Phases

This thesis has been structured around three main fields of knowledge: semantic vocabularies, linked data services and social context. Each phase has been further divided into different tasks:

1. Phase 1 (**vocabularies**)
    1. Study of semantic technologies and their application in research
    2. Survey of currently used ontologies in the Sentiment and Emotion domain, as well as in the NLP community
    3. Identification of missing definitions for Sentiment and Emotion Analysis
    4. Definition of missing vocabularies
    5. Publication of the vocabularies
    6. Promoting the vocabularies and improving them based on community feedback
2. Phase 2 (**services**)
    1. Study of the state of the art in semantic analysis services (e.g., NIF, Emotion-ML) and popular tools
    2. Definition of a generic framework for sentiment and emotion analysis services
    3. Development of a reference implementation
    4. Adaptation of popular services, to serve as motivating example
    5. Promotion of the tool and framework, and improvement based on feedback from the community
3. Phase 3 (**social context**)

1. Study of the state of the art in sentiment analysis with contextual information, and related social theories
2. Survey of works
3. Model of contextual information
4. Proposal of novel models for sentiment analysis with social context

2.3 Best practices for the publication of results

The expected results of this thesis include vocabularies, schemas, software architectures, reference implementations, surveys and classification models. All of these results need to be evaluated before being considered a scientific contribution (Section 2.4). Additionally, there are certain best practices that should be followed when publishing some of these types of results. In this section we will cover the publication of vocabularies and software.

These are the requirements for the definition of a vocabulary:

- Domain knowledge
- Thorough research to ensure no other vocabularies exist
- Identification of the core elements of the domain, and elements already modelled by other vocabularies
- Description of the domain using semantic technologies, in the form of a vocabulary
- Publication of the vocabulary and a set of examples to foster its use

There is no strong requirement to release vocabularies publicly, but doing so is highly recommended. It has several advantages, including more exposure, higher likelihook of re-utilization and modification based on feedback from the community. In consequence, the vocabularies in this thesis are public. When publishing a vocabulary we follow a set of rules: 1) the vocabulary should be published in at least one RDF format, preferably a human-readable format such as Notation3 or Turtle; 2) a documentation page with examples of use and the main concepts in the vocabulary is made, 3) the documentation and vocabulary are versioned, with a list of changes in each version, and a link to the previous version. Additionally, open vocabularies can be added to the Linked Open Vocabularies portal (Vandenbussche et al., 2017), a collection of curated reusable vocabularies for different domains. Adding your vocabulary to the LOV portal supports the project and makes it more likely for other researchers to re-use your vocabulary.

Regarding reference implementations and other forms of software results, we have taken measures to ensure that every piece of software is documented, is sufficiently tested, and behaves according to specification. To that effect, we have followed a Test-Driven Development approach, together with the Prototyping model. In practice, this means that features have been documented, tested, and added gradually to the implementation. We have used version control software (git) and an issue tracker. Each release of the software is tagged and automatically tested through a CI/CD (continuous integration, continuous delivery) pipeline. We have published all our software as Open Source on either GitHub or a public instance of GitLab. This enables a community-driven approach, where external researchers are encouraged to find bugs and send their modifications and improvements to the code. For documentation, we have mostly used auto-generated pages from the repository, hosted on ReadTheDocs (or similar). We have also distributed a packaged version of our libraries through the appropriate portals (PyPI for python). To ensure our software is compatible with as many environment as possible, and to simplify installation, we have also provided containerized images of our software. In particular, we have used Docker, the most popular open source containerization platform. Lastly, to promote the software in the research community, we have submitted a paper covering the software to an appropriate journal with an Original Software Track.

2.4 Evaluation

The expected results in this thesis fall under one of the following categories: vocabularies, software frameworks, reference implementations and classifiers (i.e., classification and predictive models).

Evaluating a vocabulary is very challenging. In our work, we will evaluate our vocabularies in two ways. First, we will evaluate the adequacy of each vocabulary by defining a series of scenarios that can not be represented using existing vocabularies, and trying to represent them using the vocabulary. The results of the representation will be tested for coverage, extensibility, and succinctness. This approach has an additional advantage: the scenarios used for evaluation also serve as documentation for the vocabulary. Secondly, our vocabularies are completely public and open to discussion, unlike other vocabularies in the literature. This allows us to receive public criticism, change requests and suggestions of new scenarios that are not properly covered by the vocabulary. We can then update the vocabularies based on that feedback, or open a discussion to cover the scenarios through other vocabularies.

Our evaluation of software frameworks is very similar to that of vocabularies. Namely,

we evaluate the coverage of a framework by its ability to cover different use cases. Its appropriateness and success is measured by its use in the community. To jumpstart the process, we also develop a reference implementation for each framework, which we apply in each use case. This enables us to assess the soundness of the framework, and the feasibility of its implementation. We also monitor for competing frameworks, and for the development of other implementations.

Reference implementations of a software framework are evaluated in different ways. First of all, a test suite is developed, following standard software engineering practices. Second of all, the implementation is tested in each of the use cases defined for the parent framework. In the case of plugin-based implementations, we also evaluate the ease of development and use of different types of plugins, as required by the use cases. The implementation is also released as open source in a public portal such as GitHub or GitLab, which encourages the community to find flaws and supply new use cases. Moreover, we require each reference implementation to be used and/or extended by at least three partners unrelated to the original development.

Lastly, classifiers are the easiest to properly evaluate, as evidenced by the literature. Each classifier is evaluated on multiple datasets, and several metrics, such as Accuracy, Precision and F-Score are calculated.

CHAPTER 3

Publications

This chapter presents the main papers published during this PhD thesis. Each paper is grouped by topic into one of three categories: ontologies and vocabularies for sentiment and emotion, using linked tools for sentiment and emotion analysis and social context for sentiment and emotion analysis. The contributions of each publication to this thesis is discussed in Section 4. A full summary of publications, including those that are not directly related to the objectives of this thesis, is included in Annex A.

3.1 Definition of vocabularies and schemas

3.1.1 Onyx: Describing Emotions on the Web of Data

Title	Onyx: Describing Emotions on the Web of Data
Authors	Sánchez-Rada, J. Fernando and Iglesias, Carlos A.
Proceedings	Proceedings of the First International Workshop on Emotion and Sentiment in Social and Expressive Media: approaches and perspectives from AI (ESSEM 2013)
ISSN	1613-0073
Volume	1096
Year	2013
Keywords	emotion analysis, Emotionml, emotions, Lemon, Linked Data, Ontology, Provenance, Semantic, semantic web, sentiment analysis
Pages	71–82
Abstract	There are several different standardised and widespread formats to represent emotions. However, there is no standard semantic model yet. This paper presents a new ontology, called Onyx, that aims to become such a standard while adding concepts from the latest Semantic Web models. In particular, the ontology focuses on the representation of Emotion Analysis results. But the model is abstract and inherits from previous standards and formats. It can thus be used as a reference representation of emotions in any future application or ontology. To prove this, we have translated resources from EmotionML representation to Onyx. We also present several ways in which developers could benefit from using this ontology instead of an ad-hoc presentation. Our ultimate goal is to foster the use of semantic technologies for emotion Analysis while following the Linked Data ideals.

Onyx: Describing Emotions on the Web of Data

J. Fernando Sànchez-Rada and Carlos A. Iglesias

Intelligent Systems Group
Telematic Systems Engineering Deparment
Technical University of Madrid (UPM)
Email: jfernando@dit.upm.es
cif@dit.upm.es

Abstract. Textual emotion analysis is a new field whose aim is to detect emotions in user generated content. It complements Sentiment Analysis in the characterization of users subjective opinions and feelings. Nevertheless, there is a lack of available lexical and semantic emotion resources that could foster the development of emotion analysis services. Some of the barriers for developing such resources are the diversity of emotion theories and the absence of a vocabulary to express emotion characteristics. This article presents a semantic vocabulary, called Onyx, intended to provide support to represent emotion characteristics in lexical resources and emotion analysis services. Onyx follows the Linked Data principles as it is aligned with the Provenance Ontology. It also takes a linguistic Linked Data approach: it is aligned with the Provenance Ontology, it represents lexical resources as linked data, and has been integrated with Lemon, an increasingly popular RDF model for representing lexical entries. Furthermore, it does not prescribe any emotion model and can be linked to heterogeneous emotion models expressed as Linked Data. Onyx representations can also be published using W3C EmotionML markup, based on the proposed mapping.

Keywords: ontology, emotions, emotion analysis, sentiment analysis, semantic, semantic web, linked data, provenance, emotionml, lemon

1 Introduction

From the tech-savvy to elders, our society is exponentially moving its social and professional activity to the Internet, with its myriad of services and social networks. Facebook[1] or Twitter[2] are only two of the most successful examples, producing flooding streams of user-generated data. Unluckily, quite often that information is just meant for human consumption and is only formatted to be displayed. This prevents us from automatically processing these massive streams of information to aggregate, summarize or transform them and present human users with a bigger picture. In other words, data mining techniques require machine-formatted data input.

[1] https://facebook.com
[2] https://twitter.com

In an attempt to shorten that gap, the multidisciplinary field called Sentiment Analysis or Opinion Mining was born, which aims at determining the subjectivity of human opinions. Many tools have been created to enrich or make sense out of human generated content by applying natural language processing and adding the results as annotations or tags. Whilst this solves the issue at a small scale, for each ad-hoc solution, it raises another problem: data collected by different programs presents different and sometimes incompatible formats. Linked Data introduced a lingua franca for data representation as well as a set of tools to process and share such information. Many services embraced the Linked Data concepts and are providing tools to interconnect the previously closed silos of information [28].

The Sentiment Analysis field is now evolving to determine also human emotions. An important fact about emotions is that they change the way we communicate [20]. They can be passed on just like any other information, in what some authors call emotional contagion [7]. That is a phenomenon that is clearly visible in social networks. Most of them offer a public API that makes studying the networks and information flow relatively easy. For this very reason social network analysis is an active field [18], with Emotion Mining as one of its components.

Social networks aside, another field of application of Emotion Analysis is Affective computing. There are a variety of systems whose only human-machine communication is purely text-based. These systems are often referred to as dialog systems (e.g. Q&A systems). Such systems can use the emotive information to change their behaviour and responses [20].

On the other hand, the rise of services like microblogging will inevitably lead to services that exchange and use affective information. Some social sites are already using emotions natively, giving their users the chance to share emotions or use them in queries. Facebook, for instance, recently updated the way its users can share personal statuses.

These sites have started making heavy use [3] of formats like RDFa [4] or Microformats as a bridge between web pages for human consumption and Linked Data. This made it possible to provide a better user experience and better search results despite the big amount of information these networks contain.

Combining the objective facts already published as Linked Data with subjective opinions extracted using Sentiment and Emotion analysis techniques can enable a wide array of new services. Unfortunately, there is not yet any widely accepted Linked Data representation for emotions. This paper aims at bridging this gap with the definition of a new vocabulary, Onyx.

This paper is structured as follows: Section 2 introduces the technologies that Onyx is based upon, as well as the challenges related to Emotion Analysis and creating a standard model for emotions, including a succinct overview of the formats currently in use; Section 3 covers the Onyx ontology in detail and several use cases for this ontology; Section 4 presents the results of our evaluation of the Ontology, focusing on the coverage of current formats like EmotionML; Section 5 completes this paper with our conclusions and future work.

2 Enabling Technologies

2.1 Models for Emotions and Sentiment Analysis

To work with Emotions and reason about them, we first need to have a solid understanding and model of emotions. This, however, turns out to be a rather complex task.It is comprised of two main components: modelling (including categorisation) and representation.

There are several models for emotions, ranging from the most simplistic and ancient that come from Chinese philosophers to the most modern theories that refine and expand older models [11, 22]. The literature on the topic is vast, and it is out of the scope of this paper to reproduce it. The recent work by Cambria et al. [10] contains a comprehensive state of the art on the topic, as well as an introduction to a novel model, The Hourglass of Emotions, inspired by Plutchik's studies [21]. Plutchik's model has been extensively used [8, 9] in the area of Sentiment Analysis and Affective Computing, relating all the different emotions to each other in what is called the rose of emotions.

Other models cover affects in general, which include Emotions as part of them. One of them is the work done by Strapparava and Valitutti in WordNet-Affect [27]. It comprises more than 300 affects, many of which are considered emotions. What makes this categorization interesting is that it effectively provides a taxonomy of emotions. It both gives information about relationship between emotions and makes it possible to to decide the level of granularity of the emotions expressed.

Despite all, there does not seem to be a universally accepted model for emotions [26]. This complicates the task of representing emotions. In a discussion regarding Emotion Markup Language (EmotionML), Schroder et al. pose that *any attempt to propose a standard way of representing emotions for technological contexts seems doomed to fail* [25]. Instead they claim that the markup should offer users choice of representation, including the option to specify the affective state that is being labelled, different emotional dimensions and appraisal scales. The level of intensity completes their definition of an affect in their proposal.

EmotionML [6] is one of the most notable general-purpose emotion annotation and representation languages. It was born from the efforts made for Emotion Annotation and Representation Language (EARL) [1, 26] by Human-Machine Interaction Network on Emotion (HUMAINE) EARL originally included 48 emotions divided into 10 different categories. EmotionML offers twelve vocabularies for categories, appraisals, dimensions and action tendencies. A vocabulary is a set of possible values for any given attribute of the emotion. There is a complete description of those vocabularies and its computer-readable form available [5].

In the field of Semantic Technologies, Grassi presented Human Emotion Ontology (HEO). This ontology presents an ontology for human emotions for its use for annotating emotions in multimedia data. Another work worth mentioning is that of Hastings et al. [14] in Emotion Ontology (EMO), an ontology that tries to reconcile the discrepancies in affective phenomena terminology.

For Opinion Mining we find the Marl vocabulary [29]. Marl was designed to annotate and describe subjective opinions expressed in text. In essence, it provides the conceptual tools to annotate Opinions and results from Sentiment Analysis in an open and sensible format. However, it is focused on polarity extraction and is not capable of representing Emotions. Onyx aims to remedy this and offer a complete set of tools for any kind of Sentiment Analysis, including advanced Emotion Analysis.

Lastly, it is worth mentioning lemon, the Lexicon Model for Ontologies. As its name indicates, it is a model that supports the sharing of terminological and lexicon resources on the Semantic Web as well as their linking to the existing semantic representation provided by ontologies [16]. Onyx will be used together with lemon to annotate lexicon resources for Emotion Analysis, as will be shown in some of the examples below.

2.2 W3C's Provenance

Provenance is information about entities, activities, and people involved in producing a piece of data or thing, which can be used to form assessments about its quality, reliability or trustworthiness. The PROV Family of Documents defines a model, corresponding serializations and other supporting definitions to enable the inter-operable interchange of provenance information in heterogeneous environments such as the Web [19]. It includes a full-fledged ontology, to which Onyx is linked. The complete ontology is covered by the PROV-O Specification. However, to understand the role of Provenance in Onyx and vice versa, it is enough to understand Figure 1.

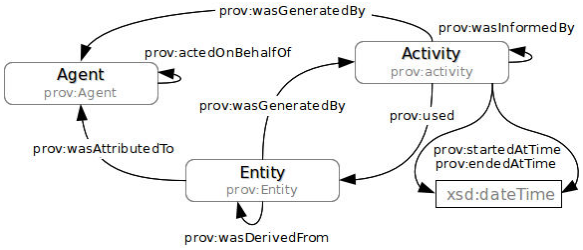


Fig. 1: Simple overview of the basic classes in the Provenance Ontology [12]

As we can see, Agents take part in Activities to transform Entities (data) into different Entities (modified data). This process can be aggregation of information, translation, adaptation, etc. In our case, this activity is an Emotion Analysis, which turns plain data into semantic emotion information.

There are many advantages to adding provenance information in Sentiment Analysis in particular as different algorithms may produce different results. By including the Provenance classes in our Emotion Mining Ontology we can not only link results with the source from which it was extracted, but also with the algorithm that produced them.

3 Onyx

Onyx is a vocabulary to represent the Emotion Analysis process and its results, as well as annotating lexical resources for Emotion Analysis. It includes all the necessary classes and properties to provide structured and meaningful Emotion Analysis results, and to connect results from different providers and applications.

At its core, the Onyx ontology has three main classes: EmotionAnalysis, EmotionSet and Emotion. In a standard Emotion Analysis, these three classes are related as follows: an EmotionAnalysis is run on a source (generally in the form of text, e.g. a status update), the result is represented as one or more EmotionSet instances that contain one or more Emotion instances.

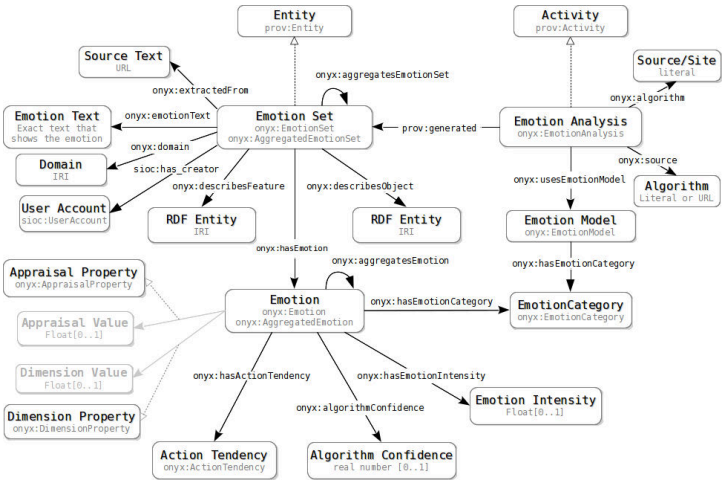


Fig. 2: Class diagram of the Onyx ontology.

The EmotionAnalysis instance contains information about: the source (e.g. dataset, website) from which the information was taken, the algorithm used, and the emotion model that was used to represent emotions. Additionally, it

can make use of Provenance to specify the Agent in charge of the analysis, the resources used (e.g. dictionaries), and other useful information.

An EmotionSet contains a group of emotions found in the text or one of its parts. As such, it contains information about: the original text (extractedFrom); the exact excerpt that contains the emotion or emotions (emotionText); the person that showed the emotions (sioc:has_creator); the entity that the emotion is related to (describesObject); the concrete part of that object it refers to (describesObjectPart); the feature about that part or object that triggers the emotion (describesFeature); and, lastly, the domain detected. All this properties are straightforward, but a note should be given about the domain property. Different emotions could have different interpretations in different contexts (e.g., fear is positive when referred to a thriller, but negative when it comes to cars and safety).

When several EmotionSet instances are related, an AggregatedEmotionSet can be created that links to all of them. For instance, we could aggregate all the emotions about a Movie, or all the emotions shown by a particular user. An AggregatedEmotionSet is a subclass of EmotionSet which contains additional information about the original EmotionSet instances it aggregates.

Considering the lack of consensus on modeling and categorizing emotions, our model of emotions is very generic. In this Emotion model we include: an EmotionCategory or type of emotion (although more could be specified), through the hasEmotionCategory property (e.g. "sadness"); the emotion intensity; action tendencies (ActionTendency) related to this emotion, or actions that are triggered by the emotion; appraisals and dimensions. Appraisals and dimensions are defined as properties, whose value is a float number. On top of that generic model, we have adapted two different systems: the WordNet-Affect taxonomy, and the EmotionML vocabularies for categories, dimensions and appraisals.

WordNet-Affect [27] contains the relationships (concepts and superconcepts) of affects, among which we find emotions. We processed the list of affects and published a SKOS version of the taxonomy [24]. The taxonomy specification includes a navigable tree that contains the concepts (i.e. affect types) in it, aligned with WordNet concepts. This makes it trivial to select an affect that represents the desired emotion. Besides providing a good starting point for other ontologies, this taxonomy also serves as a base to translate between the several different ontologies in the future.

Regarding EmotionML, we have converted its vocabularies [5] the Onyx format. Using this extension we can translate EmotionML resources into Onyx for their use in the Semantic Web.

This is further developed in Section 4.

It is also possible that two separate emotions, when found simultaneously, imply a third emotion. A more complex one. For instance, "thinking of the awful things I've done makes me want to cry" might reveal sadness and disgust, which together might be interpreted as remorse. In such situation, we could add an AggregatedEmotion that represents remorse to the EmotionSet, linking it to the primary emotions with the aggregatesEmotion property.

To group all the attributes that correspond to a specific emotion model, we created the EmotionModel class. Each EmotionModel will be linked to the different categories it contains (hasEmotionCategory), the AppraisalProperty or DimensionProperty instances it introduces (through hasAppraisalProperty and hasDimensionProperty), etc.

Figure 2 shows a complete overview of all these classes, as well as all their properties.

After this introduction of the ontology, we will present several use cases for it. This should give a better understanding of the whole ontology by example. Rather than exhaustive and complex real life applications, these examples are meant as simple self-contained showcases of the capabilities of semantic Emotion Analysis using Onyx. For the sake of brevity, we will omit the prefix declaration in the examples.

Case	**N3 Representation**

An example Emotion-Analysis.	`:customAnalysis`
	`  a onyx:EmotionAnalysis;`
	`  onyx:algorithm "SimpleAlgorithm";`
	`  onyx:usesEmotionModel wna:WNAModel.`

Processing "I lost one hour today because of the strikes!!", by the user JohnDoe	`:result1`
	`  a onyx:EmotionSet;`
	`  prov:wasGeneratedBy :customAnalysis;`
	`  sioc:has_creator [`
	`  sioc:UserAccount <http://blog.example.`
	`  com/JohnDoe>. ];`
	`  onyx:hasEmotion [`
	`  onyx:hasEmotionCategory wna:anger;`
	`  onyx:hasEmotionIntensity :0.9 ];`
	`  onyx:emotionText "I lost one hour today`
	`  because of the strikes!!" ;`
	`  dcterms:created "2013-05-16T19:20:30+01:00"`
	`  ^^dcterms:W3CDTF.`

Example of annotation of a lexical entry using Onyx and lemon [17].	`:fifa`
	`  a lemon:Lexicalentry;`
	`  lemon:sense [`
	`  lemon:reference wn:synset-fear-noun-1;`
	`  onyx:hasEmotion`
	`  [ oyx:hasEmotionCategory wna:fear. ].`
	`  ];`
	`  lexinfo:partOfSpeech lexinfo:noun.`

Table 1: Representation with Onyx

4 Evaluation

Evaluating ontologies is always a difficult task. Evaluation methodologies are highly debatable and there are no standards [13]. For the evaluation of Onyx we focused on its practical use as well as in its correctness. This means testing the adequacy of the model for existing applications as well as scenarios with several emotion models. In particular we have chosen two different test scenarios: the

Case	Query

Finding all the users that did not feel good during last New Year's Eve, and the exact emotions they felt.	`SELECT DISTINCT ?creator ?cat`
	`WHERE {`
	`  ?set onyx:hasEmotion`
	`  [ onyx:hasEmotionCategory ?cat];`
	`  dcterms:created ?date;`
	`  sioc:has_creator ?creator.`
	`  ?cat skos:broaderTransitive* wna:negative`
	`  -emotion.`
	`  FILTER( ?date >= xsd:date("2012-12-31")`
	`  ?date <= xsd:date("2013-01-01") )`
	`}`

Comparing two Emotion Mining algorithms by comparing the discrepancies in the results obtained using both.	`SELECT ?source1 ?algo1 (GROUP_CONCAT(?cat1)`
	`  as ?cats1)`
	`WHERE {`
	`  ?set1 onyx:extractedFrom ?source1.`
	`  ?analysis1 prov:generated ?set1;`
	`  onyx:algorithm ?algo1.`
	`  ?set1 onyx:hasEmotion`
	`  [ onyx:hasEmotionCategory ?cat1 ].`
	`  FILTER EXISTS{`
	`  ?set2 onyx:extractedFrom ?source1.`
	`  ?analysis2 prov:generated ?set2.`
	`  ?set2 onyx:hasEmotion`
	`  [ onyx:hasEmotionCategory ?cat2 ].`
	`  FILTER ( ?set1 != ?set2).`
	`  FILTER ( ?cat2 != ?cat1 ).`
	`  }`
	`}`
	`GROUP BY ?source1 ?algo1`
	`ORDER BY ?source1`

Table 2: Example SPARQL queries with Onyx

adaptation of a well-known Emotion Analysis tool to output Onyx, Synesketch [2,], and the translation of EmotionML resources to Onyx and vice versa.

For the EmotionML part, the evaluation process is split into two parts: transforming the EmotionML categories into a semantic format, and representing EmotionML cases with Onyx. The result of the former can be seen in [23], which has been used as namespace (emlonyx) in the translation of an EmotionML example in Table 3. The specification of EmotionML is public, including its XML schema, which eased the process of mapping it to Onyx. We have focused especially on representing EmotionML emotions in Onyx.

Synesketch is a library and application that detects emotions in English texts and can generate images that reflect those emotions. Originally written in Java, it has been unofficially ported to several programming languages (including PHP), which shows the interest of the community in this tool. The aim of the PHP port was, among others, to offer a public endpoint for emotion analysis, which later had to be taken down due to misuse. The relevance of this tool and its Open Source license were the leading factors in choosing this tool. Our approach has been to develop a proof-of-concept web service that performs Emotion Analysis using Synesketch's emotion analysis. The service can be accessed via a REST API and its results are presented in Onyx, using the RDF format.

The Synesketch library uses the big-6 emotional model, which comprises: happiness, sadness, fear, anger, disgust and surprise. Each of those emotions are present in the input text with a certain weight that ranges from 0 to 1. Additionally, it has two attributes more that correspond to the general emotional valence (positive, negative or neutral) and the general emotional weight. In other words, these attributes together show how "positive", "negative" or "neutral" the overall emotion is.

To represent the big-6 emotion category in Onyx we used EmotionML's big-6 category, which we previously mapped to Onyx. The Synesketch weight directly mapped to hasEmotionIntensity in Onyx.

However, the General Emotional valence and weight do not directly match any Onyx property or class. To solve it, we simply added an AggregatedEmotion with the PositiveEmotion, NeutralEmotion or NegativeEmotion category (as defined by WordNet-Affect) depending on the value of the valence. The general emotional weight is then the intensity of this AggregatedEmotion, just like in the other cases.

The final result is a REST service that is publicly available at our website[3].

EmotionML	**Onyx**

`<emotionml xmlns="http://.../emotionml" xmlns:meta="http://.../metadata" category-set="http://.../#everyday-categories">`
`<info>`
`<classifiers:classifier classifiers:name="GMM"/>`
`</info>`
`<emotion>`
`<category name="Disgust" value="0.82"/>`
'Come, there is no use in crying like that!'
`</emotion>`
said Alice to herself rather sharply;
`<emotion>`
`<category name="Anger" value="0.57"/>`
'I advise you to leave off this minute!'
`</emotion>`
`</emotionml>`
	:Set1 a onyx:EmotionSet;
	onyx:extractedFrom "Come, there is no use in crying like that! said Alice to herself rather sharply; I advice you to live off this minute!";
	onyx:hasEmotion :Emo1
	onyx:hasEmotion :Emo2
	:Emo1 a onyx:Emotion;
	onyx:hasEmotionCategory emlonyx:disgust;
	onyx:hasEmotionIntensity 0.82;
	onyx:hasEmotionText "Come, there's no use in crying like that!"
	:Emo2 a onyx:Emotion;
	onyx:hasEmotionCategory emlonyx:anger;
	onyx:hasEmotionIntensity 0.57;
	onyx:hasEmotionText "I advice you to leave off this minute!"
	:Analysis1 a onyx:EmotionAnalysis;
	onyx:algorithm "GMM";
	onyx:usesEmotionModel emlonyx:everyday-categories;
	prov:generated Set1.

Table 3: Representation of EmotionML with Onyx

[3] http://demos.gsi.dit.upm.es/onyxemote/

5 Conclusions and Future Work

With this work we have introduced an option to represent Emotions that takes advantage of the work conducted in the field of Semantic Web. This ontology presents characteristics that are particularly beneficial for any process of Emotion Analysis. Onyx provides a structured format for Emotion Analysis. It addresses the problem of supporting heterogeneous categories of emotions, and new categories and features can be added, using the recommended taxonomy to link them and retain compatibility.

We also presented how Onyx would be used in several scenarios. Furthermore, we adapted some of the existent resources and services to Onyx, making them publicly available.

Although this paper is focused on Emotion Analysis, emotive information can also be directly provided by users. Either given explicitly or extracted via an automated process (Emotion Analysis), the information they represent is the same. A single ontology should thus cover both scenarios. This is possible with Onyx, as we demonstrate in this paper.

We would like to note that our proposal is compatible with EMO, since EMO can be easily mapped to Onyx using the property usesEmotionModel. The situation is similar with the proposal of Lopez et al. [15], which focuses on emotions instead of affects in general. The integration with HEO will be investigated. Onyx's and HEO's Emotion classes are very similar overall, but follow different approaches in several aspects.

With all this in mind, we consider that using Onyx to represent Emotion Mining results is highly beneficial.

As part of the future plans for Onyx, it will be actively used in the EUROSENTIMENT[4] project, whose aims is to create a language resource pool for Sentiment Analysis. Together with Marl [29] and Lemon [17], they will be the standard formats for representation of lexicons and results. Therefore all the services provided in the frame of the EUROSENTIMENT project will export emotional information using Onyx. Marl has already been integrated in NIF 2.0[5] to represent opinions, and efforts will be made to integrate Onyx as well for emotions.

There is also room for experimentation emotion composition and inference using tools such as SPIN[6]. It is possible to infer complex emotions whenever other simple emotions are present, and vice versa. The same techniques could be used to work with different emotion models

Finally, our research group will use the integration with EmotionML to develop intelligent personal agents that benefit from the potential of the Semantic Web.

[4] `http://eurosentiment.eu`
[5] `http://persistence.uni-leipzig.org/nlp2rdf/ontologies/nif-core/nif-core.html`
[6] `http://spinrdf.org/`

6 Acknowledgements

This work was partially funded by the EUROSENTIMENT FP7 Project (Grant Agreement no: 296277)

References

1. Humaine Emotion Annotation and Representation Language (EARL): Proposal., June 2006, `http://emotion-research.net/projects/humaine/earl/proposal#Dialects`.
2. Synesketch: Free Open-Source Software for Textual Emotion Recognition and Visualization, June 2006, `http://emotion-research.net/projects/humaine/earl/proposal#Dialects`.
3. Facebook Open Graph API, June 2013, `http://developers.facebook.com/docs/opengraph/`.
4. Ben Adida, Mark Birbeck, Shane McCarron, and Steven Pemberton. RDFa in XHTML: Syntax and processing. *Recommendation, W3C*, 2008.
5. Kazuyuki Ashimura, Paolo Baggia, Felix Burkhardt, Alessandro Oltramari, Christian Peter, and Enrico Zovato. EmotionML vocabularies, May 2012, `http://www.w3.org/TR/2012/NOTE-emotion-voc-20120510/`.
6. Paolo Baggia, Felix Burkhardt, Catherine Pelachaud, Christian Peter, and Enrico Zovato. Emotion Markup Language (EmotionML) 1.0, April 2013,`http://www.w3.org/TR/emotionml/`.
7. Sigal G. Barsade. The ripple effect: Emotional contagion and its influence on group behavior. *Administrative Science Quarterly*, 47(4):644–675, 2002.
8. Damian Borth, Tao Chen, Rongrong Ji, and Shih-Fu Chang. SentiBank: large-scale ontology and classifiers for detecting sentiment and emotions in visual content. In *Proceedings of the 21st ACM international conference on Multimedia*, MM '13, pages 459–460, New York, NY, USA, 2013. ACM.
9. Erik Cambria, Catherine Havasi, and Amir Hussain. SenticNet 2: A semantic and affective resource for opinion mining and sentiment analysis. In *FLAIRS Conference*, pages 202–207, 2012.
10. Erik Cambria, Andrew Livingstone, and Amir Hussain. The hourglass of emotions. In *Cognitive Behavioural Systems*, pages 144–157. Springer, 2012.
11. Paul Ekman. Basic emotions. *Handbook of cognition and emotion*, 98:45–60, 1999.
12. Paul Groth and Luc Moreau. Prov-O W3C Recommendation, April 2013, `http://www.w3.org/TR/prov-o/`.
13. Asunción Gómez-Pérez. Evaluation of ontologies. *International Journal of Intelligent Systems*, 16(3):391–409, 2001.
14. Janna Hastings, Werner Ceusters, Barry Smith, and Kevin Mulligan. Dispositions and processes in the emotion ontology. In *ICBO*, 2011.
15. Juan Miguel López, Rosa Gil, Roberto García, Idoia Cearreta, and Nestor Garay. Towards an ontology for describing emotions. In *Emerging Technologies and Information Systems for the Knowledge Society*, pages 96–104. Springer, 2008.
16. John McCrae, Dennis Spohr, and Philipp Cimiano. Linking lexical resources and ontologies on the semantic web with lemon. In *The Semantic Web: Research and Applications*, pages 245–259. Springer, 2011.

17. John McCrae, Dennis Spohr, and Philipp Cimiano. Linking lexical resources and ontologies on the semantic web with Lemon. In Grigoris Antoniou, Marko Grobelnik, Elena Simperl, Bijan Parsia, Dimitris Plexousakis, Pieter Leenheer, and Jeff Pan, editors, *The Semantic Web: Research and Applications*, volume 6643 of *Lecture Notes in Computer Science*, pages 245–259. Springer Berlin Heidelberg, 2011.
18. Alan Mislove, Massimiliano Marcon, Krishna P. Gummadi, Peter Druschel, and Bobby Bhattacharjee. Measurement and analysis of online social networks. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, IMC '07, pages 29–42, New York, NY, USA, 2007. ACM.
19. Luc Moreau, Ben Clifford, Juliana Freire, Joe Futrelle, Yolanda Gil, Paul Groth, Natalia Kwasnikowska, Simon Miles, Paolo Missier, Jim Myers, Beth Plale, Yogesh Simmhan, Eric Stephan, and Jan Van den Bussche. The open provenance model core specification (v1.1). *Future Generation Computer Systems*, 27(6):743 – 756, 2011.
20. Bo Pang and Lillian Lee. Opinion mining and sentiment analysis. *Foundations and trends in information retrieval*, 2(1-2):1–135, 2008.
21. Robert Plutchik. *Emotion: A psychoevolutionary synthesis*. Harper & Row New York, 1980.
22. Jesse J Prinz. Gut reactions: A perceptual theory of emotion. 2004.
23. J. Fernando Sànchez-Rada and Carlos A. Iglesias. EmotionML categories for Onyx, July 2013, `http://gsi.dit.upm.es/ontologies/onyx/emotionml`.
24. J. Fernando Sànchez-Rada and Carlos A. Iglesias. WordNet-Affect SKOS Taxonomy, May 2013,`http://gsi.dit.upm.es/ontologies/wnaffect/`.
25. Marc Schröder, Laurence Devillers, Kostas Karpouzis, Jean-Claude Martin, Catherine Pelachaud, Christian Peter, Hannes Pirker, Björn Schuller, Jianhua Tao, and Ian Wilson. What should a generic emotion markup language be able to represent? In *Affective Computing and Intelligent Interaction*, pages 440–451. Springer, 2007.
26. Marc Schröder, Hannes Pirker, and Myriam Lamolle. First suggestions for an emotion annotation and representation language. In *Proceedings of LREC*, volume 6, pages 88–92. Citeseer, 2006.
27. Carlo Strapparava and Alessandro Valitutti. Wordnet-affect: an affective extension of wordnet. In *Proceedings of LREC*, volume 4, pages 1083–1086, 2004.
28. Giovanni Tummarello, Renaud Delbru, and Eyal Oren. Sindice. com: Weaving the open linked data. In *The Semantic Web*, pages 552–565. Springer, 2007.
29. Adam Westerski, Carlos A. Iglesias, and Fernando Tapia Rico. Linked opinions: Describing sentiments on the structured web of data. In *4th international workshop Social Data on the Web (SDoW2011)*, Bonn, Germany, October 2011.

3.1.2 Onyx: A Linked Data Approach to Emotion Representation

Title	Onyx: A Linked Data Approach to Emotion Representation
Authors	Sánchez-Rada, J. Fernando and Iglesias, Carlos A.
Journal	Information Processing & Management
Impact factor	JCR 2016 2.391 (Q1)
ISSN	0306-4573
Publisher	
Volume	52
Year	2016
Keywords	emotion analysis, Emotionml, EmotionML, Lexical resources, linked data, Linked data, Linked Data, Ontology, onyx, Provenance
Pages	99–114
Online	http://www.sciencedirect.com/science/article/pii/S030645731500045X
Abstract	Extracting opinions and emotions from text is becoming more and more important, especially since the advent of micro-blogging and social networking. Opinion mining has become particularly popular and now gathers many public services, datasets and lexical resources. Unfortunately, there are few available lexical and semantic resources for emotion recognition that could foster the development of new emotion aware services and applications. Some of the barriers for developing such resources are the diversity of emotion theories and the absence of a common vocabulary to express emotion. This article presents a semantic vocabulary, called Onyx, intended to provide support to represent emotions in lexical resources and emotion analysis services. It follows a linguistic Linked Data approach, it is aligned with the Provenance Ontology, and it has been integrated with lemon, an increasingly popular RDF model for representing lexical entries. This approach also means a new and interesting way to work with different theories of emotion. As part of our work, Onyx has been aligned with EmotionML and WordNet-Affect.
Problem solved	

Onyx: A Linked Data Approach to Emotion Representation

J. Fernando Sánchez-Rada, Carlos A. Iglesias

Intelligent Systems Group
Departamento de Ingeniería de Sistemas Telemáticos
Universidad Politécnica de Madrid (UPM)

Abstract

Extracting opinions and emotions from text is becoming increasingly important, especially since the advent of micro-blogging and social networking. Opinion mining is particularly popular and now gathers many public services, datasets and lexical resources. Unfortunately, there are few available lexical and semantic resources for emotion recognition that could foster the development of new emotion aware services and applications. The diversity of theories of emotion and the absence of a common vocabulary are two of the main barriers to the development of such resources. This situation motivated the creation of Onyx, a semantic vocabulary of emotions with a focus on lexical resources and emotion analysis services. It follows a linguistic Linked Data approach, it is aligned with the Provenance Ontology, and it has been integrated with the Lexicon Model for Ontologies (lemon), a popular RDF model for representing lexical entries. This approach also means a new and interesting way to work with different theories of emotion. As part of this work, Onyx has been aligned with EmotionML and WordNet-Affect.

Keywords: emotion analysis, ontology, provenance, Linked Data, EmotionML, lexical resources

1. Introduction

With the rise of social media, more and more users are sharing their opinions and emotions online [1]. The increasing volume of information and number of users are drawing the attention of researchers and companies alike, which seek not only academical results but also profitable applications such as brand monitoring. As a result, many tools and services have been created to enrich or make sense out of human generated content. Unfortunately, they are isolated data silos or tools that use very different annotation schemata. Even worse, the scarce available resources are also suffering from the heterogeneity of formats and models of emotion, making it hard to combine different resources.

Linked Data can change this situation, with its lingua franca for data representation as well as a set of tools to process and share such information. Plenty

Email addresses: jfernando@dit.upm.es (J. Fernando Sánchez-Rada), cif@gsi.dit.upm.es (Carlos A. Iglesias)

of services have already embraced the Linked Data concepts and are providing tools to interconnect the previously closed silos of information [2]. In fact, the Linked Data approach has proven useful for fields like Opinion Mining. Some schemata offer semantic representation of opinions [3], allowing richer processing and interoperability.

Emotions have a crucial role in our lives, and even change the way we communicate [1]. They can be passed on just like any other kind of information, in what some authors call emotional contagion [4]. That is a phenomenon that is clearly visible in social networks [5]. Public APIs make it relatively easy to study the social networks and their information flow. For this very reason Social Network Analysis is an active field [6], with Emotion Mining as one of its components. On the other hand, the growing popularity of services like microblogging will inevitably lead to services that exchange and use emotion in their interactions. Some social sites are already using emotions natively, giving their users the chance to share emotions or use them in queries. A noteworthy example is Facebook, which recently updated the way its users can share personal statuses.

Nevertheless, the impact of emotion analysis goes well beyond social networks. For instance, there are a variety of systems whose only human-machine communication is purely text-based. Such systems can use the emotive information to change their behavior and responses [1].

Furthermore, there are many sources that can be used for sentiment analysis beyond pure text, including video and audio. Multimodal analysis, or making use of several of these sources, is an active field that gathers experts from different disciplines. A unified schema and appropriate tooling would open up new possibilities in this field.

Lastly, combining subjective information from emotion analysis with facts already published as Linked Data could enable a wide array of new services. This would require a widely accepted Linked Data representation for emotions, which does not exist yet. In this paper we present Onyx, a new vocabulary that aims to bridge this gap and allow for interoperable tools and resources. We also provide a set of example applications, additional vocabularies to use existing models, and multilingual resources that use Onyx to annotate emotion.

The rest of this paper is structured as follows. Section 2 introduces the technologies that Onyx is based upon, as well as the challenges related to emotion analysis and creating a standard model for emotions, including a succinct overview of the formats currently in use. Section 3 covers the Onyx ontology in detail, including some vocabularies or models of emotions, and several examples. Section 4 presents the results of our evaluation of the Ontology, focusing on the coverage of current formats like Emotion Markup Language (EmotionML). Finally, Section 5 completes this paper with conclusions and future work.

2. Enabling technologies

2.1. Models for emotions and emotion analysis

To work with emotions and reason about them, we first need to have a solid understanding and model of emotions. This, however, turns out to be a rather complex task. It is comprised of two main components: modeling (including categorization) and representation.

Confusingly, the terms opinion, sentiment, emotion, feeling and affect are commonly used interchangeably. Throughout this article we follow the terminology by Cambria et al. [7] where opinion mining and sentiment analysis are focused on polarity detection and emotion recognition, respectively.[1]

There are also several models for emotions, ranging from the most simplistic and ancient that come from Chinese philosophers to the most modern theories that refine and expand older models [9, 10]. The literature on the topic is vast, and it is out of the scope of this paper to reproduce it. The recent work by Cambria et al. [11] contains a comprehensive state of the art on the topic, as well as an introduction to a novel model, *the Hourglass of emotions*, inspired by Plutchik's studies [12]. Plutchik's model is a model of categories that has been extensively used [13, 14, 15] in the area of emotion analysis and affective computing, relating all the different emotions to each other in what is called the wheel of emotions.

All the existing motions are mainly divided in two groups: discrete and dimensional models. In discrete models, emotions belong to one of a predefined set of categories, which varies from model to model. In dimensional models, an emotion is represented by the value in different axes or dimensions. A third category, mixed models, merges both views.

Other models are more general and model affects, including emotions as a subset. One of them is the work done by Strapparava and Valitutti in WordNet-Affect [16], an affective lexicon on top of WordNet. WordNet-Affect comprises more than 300 affective labels linked by concept-superconcept relationships, many of which are considered emotions. What makes this categorization interesting is that it effectively provides a taxonomy of emotions. It both gives information about relationships between emotions and makes it possible to decide the level of granularity of the emotions expressed. Section 3.2.1 discusses how we formalized this taxonomy using SKOS, and converted that taxonomy into an Onyx vocabulary.

Despite all, there does not seem to be a universally accepted model for emotions [17]. This complicates the task of representing emotions. In a discussion regarding EmotionML, Schroder et al. pose that *given the fact that even emotion theorists have very diverse definitions of what an emotion is, and that very different representations have been proposed in different research strand, any attempt to propose a standard way of representing emotions for technological contexts seems doomed to fail* [18]. Instead they claim that the markup should offer users choice of representation, including the option to specify the affective state that is being labeled, different emotional dimensions and appraisal scales. The level of intensity completes their definition of an affect in their proposal.

EmotionML [19] is one of the most notable general-purpose emotion annotation and representation languages. It was born from the efforts made for Emotion Annotation and Representation Language (EARL) [20, 17] by Human-Machine Interaction Network on Emotion (HUMAINE). EARL originally included 48 emotions divided into 10 different categories. EmotionML offers twelve vocabularies for categories, appraisals, dimensions and action tendencies. A vocabulary is a set of possible values for any given attribute of the emotion. There is a complete description of those vocabularies and their

[1] A more detailed terminology discussion can be found in Munezero et al. [8].

computer-readable form available [21].

In the field of Semantic Technologies, Grassi introduced Human Emotion Ontology (HEO), an ontology for human emotions meant for annotating emotions in multimedia data. We discuss some differences between Onyx and HEO in Section 5. Another work worth mentioning is that of Hastings et al. [22] in Emotion Ontology (EMO), an ontology that tries to reconcile the discrepancies in affective phenomena terminology. It is, however, too general to be used in the context of emotion analysis: it provides a qualitative notion of emotions, when a quantitative one would be needed.

For Opinion Mining we find the Marl vocabulary [3]. Marl was designed to annotate and describe subjective opinions expressed in text. In essence, it provides the conceptual tools to annotate opinions and results from Opinion Mining in an open and sensible format.

2.2. Linguistic Linked Open Data (LLOD)

LLOD [23, 24] is an initiative that promotes the use of linked data technologies for modeling, publishing and interlinking linguistic resources. The main benefit of using linked data principles to model linguistic resources is that it provides a graph-based model that allows representing different kinds of linguistic resources (such as lexical-semantic resources, linguistic annotations or corpora) in a uniform way, thus supporting querying across resources. The creation of an LLOD cloud is a cooperative task that is been managed by several communities, such as Open Linguistics Working Group (OWLG) of the Open Knowledge Foundation and Ontology-Lexica Community Group (OntoLex) of W3C. As a result of this activity, an initial LLOD is currently available as shown in Figure 1, where several types of resources have been identified: lexical-semantic resources (e.g. machine readable dictionaries, semantic networks, semantic knowledge bases, ontologies and terminologies), annotated corpora and linguistic annotations.

With regards to modeling lexical-semantic resources, *Lexicon Model for Ontologies (lemon)* [25] proposes a framework for modeling and publishing lexicon and machine-readable dictionaries as linked data. *lemon* provides a bridge between the most influential lexical-semantic resources, WordNet [26] and DBPedia [27]. With regards to annotated corpora, there are two initiatives [28], POWLA [29] and NLP Interchange Format (NIF) [30] that enable to link lexical-semantic resources to corpora. Finally, Ontologies of Linguistic Annotation (OLiA) [31] are a repository of annotation terminology for various linguistic phenomena that can be used in combination with POWLA, NIF or *lemon*. OLiA ontologies allow to represent linguistic annotations in corpora, grammatical specifications in dictionaries, and their respective meaning within the LLOD cloud in an operable way.

The main benefits of modeling linguistic resources as linked data include [24] interoperability and integration of linguistic resources, unambiguous identification of elements of linguistic description, unambiguous links between different resources, possibility to annotate and query across distributed resources and availability of mature technological infrastructure.

2.3. W3C's Provenance

Provenance is information about entities, activities, and people involved in producing a piece of data or thing, which can be used to form assessments about

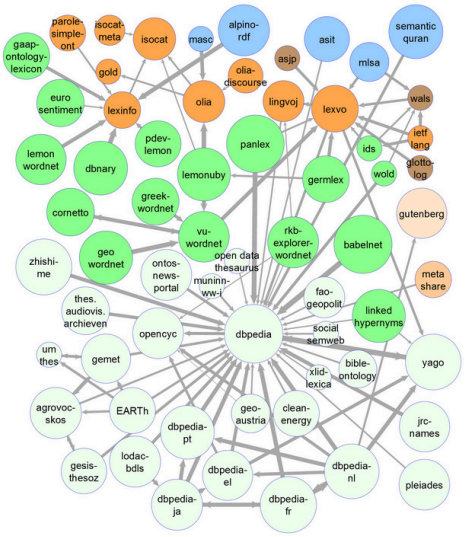


Figure 1: LLOD cloud diagram by Open Linguistics Working Group on May 2014 available at `http://linguistics.okfn.org/`

its quality, reliability or trustworthiness. The PROV Family of Documents defines a model, corresponding serializations and other supporting definitions to enable the inter-operable interchange of provenance information in heterogeneous environments such as the Web [32]. It includes a full-fledged ontology that other ontologies like Onyx can link to. Figure 2 shows the very basic classes in PROV-O, which should be enough to understand the role of Provenance in Onyx. The complete ontology is covered by the PROV-O Specification [33].

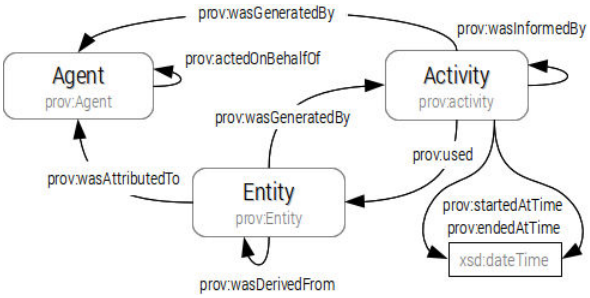


Figure 2: Simple overview of the basic classes in the Provenance Ontology [33]

As we can see, Agents take part in Activities to transform Entities (data)

into different Entities (modified data). This process can be aggregation of information, translation, adaptation, etc. In our case, this activity is an emotion analysis, which turns plain data into semantic emotion information.

There are many advantages to adding provenance information in emotion analysis in particular, as different algorithms may produce different results.

3. Onyx ontology and vocabularies

This section gives a comprehensive view of the ontology and is structured in three parts. Subsection 3.1 presents the ontology in full. Subsection 3.2 shows how different vocabularies are represented with the ontology, using three known models of emotion. Lastly, Subsection 3.3 exemplifies how to annotate data using the ontology and vocabularies from the previous section.

3.1. Onyx ontology

Onyx is a vocabulary that models emotions and the emotion analysis process itself. It can be used represent the results of an emotion analysis service or the lexical resources involved (e.g. corpora and lexicons). This vocabulary can connect results from different providers and applications, even when different models of emotions are used.

At its core, the ontology has three main classes: *Emotion*, *EmotionAnalysis* and *EmotionSet*. In a standard emotion analysis, these three classes are related as follows: an *EmotionAnalysis* is run on a source (generally text, e.g. a status update), the result is represented as one or more *EmotionSet* instances that contain one or more *Emotion* instances each.

The model of emotions in Onyx is very generic, which reflects the lack of consensus on modeling and categorizing emotions. An advantage of this approach is that the representation and psychological models are decoupled.

The *EmotionAnalysis* instance contains information about the source (e.g. dataset) from which the information was taken, the algorithm used to process it, and the emotion model followed (e.g. Plutchik's categories). Additionally, it can make use of Provenance to specify the Agent in charge of the analysis, the resources used (e.g. dictionaries), and other useful information.

An *EmotionSet* contains a group of emotions found in the text or in one of its parts. As such, it contains information about: the original text (*extractedFrom*); the exact excerpt that contains the emotion or emotions (*emotionText*); the person that showed the emotions (*sioc:has_creator*); the entity that the emotion is related to (*describesObject*); the concrete part of that object it refers to (*describesObjectPart*); the feature about that part or object that triggers the emotion (*describesFeature*); and, lastly, the domain detected. All these properties are straightforward, but a note should be given about the domain property. Different emotions could have different interpretations in different contexts (e.g., fear is positive when referred to a thriller, but negative when it comes to cars and safety).

When several *EmotionSet* instances are related, an *AggregatedEmotionSet* can be created that links to all of them. *AggregatedEmotionSet* is a subclass of *EmotionSet* that contains additional information about the original *EmotionSet* instances it aggregates. For instance, we could aggregate all the emotions related to a particular movie, or all the emotions shown by a particular user, and still be able to trace back to the original individual emotions.

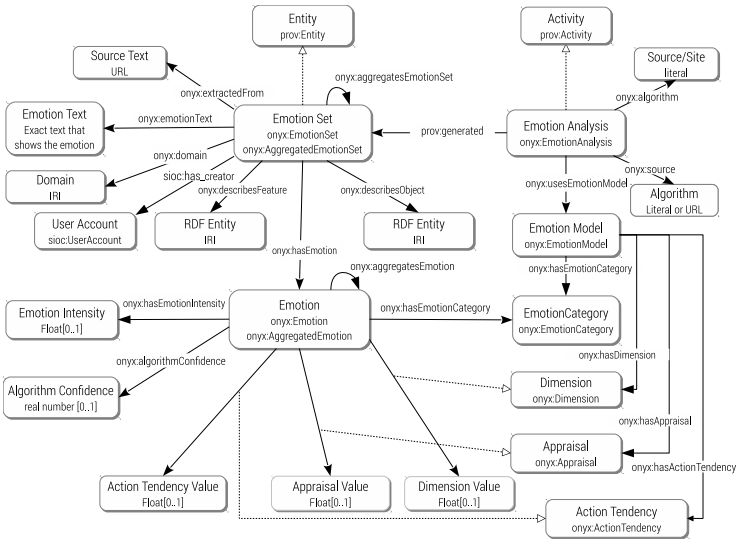


Figure 3: Class diagram of the Onyx ontology.

Onyx's *Emotion* model includes: *EmotionCategory* which is a specific category of emotion (e.g. "sadness", although more than one could be specified), linked through the *hasEmotionCategory* property; the emotion intensity via *hasEmotionIntensity*; action tendencies related to this emotion, or actions that are triggered by the emotion; appraisals and dimensions. Lastly, specific appraisals, dimensions and action tendencies can be defined by sub-classing *Appraisal*, *Dimension* and *ActionTendency*, whose value should be a float number.

On top of that generic model we have included two different models: the WordNet-Affect taxonomy, and the EmotionML vocabularies for categories, dimensions and appraisals, which are detailed in Section 3.2.

Although emotional models and categories differ in how they classify or quantify emotions, they describe different aspects of the same complex phenomenon emotion [34]. Hence, there are equivalence relationships between different categories or emotions in different models. To state such equivalence between emotion categories in Onyx one can use the properties defined in SKOS[2] such as *skos:exactMatch* or *skos:closeMatch*. This approach falls short when dealing with dimensional emotional theories or complex category theories. Since dimensional models are widely used in practice, Section 4.4 covers how to deal with this issue in detail.

Within a single model, it is also possible that two separate emotions, when found simultaneously, imply a third one. For instance, "thinking of the awful things I've done makes me want to cry" might reveal sadness and disgust,

[2] `http://www.w3.org/TR/skos-reference/#mapping`

EmotionAnalysis	
Property	**Description**
source	Identifies the source of the user generated content
algorithm	Emotion analysis algorithm that was used
usesEmotionModel	Link to the emotion model used, which defines the categories, dimensions, appraisals, etc.
EmotionSet	
Property	**Description**
domain	The specific domain in which the EmotionAnalysis was carried out
algorithmConfidence	Numeric value that represents the predicted accuracy of the result
extractedFrom	Text or resource that was subject to the analysis
hasEmotion	An Emotion that is shown by the EmotionSet. An EmotionExpression may contain several Emotions.
Emotion	
Property	**Description**
hasEmotionCategory	The type of emotion, defined by an instance of the Emotion Ontology as specified in the corresponding EmotionAnalysis
hasEmotionIntensity	Degree of intensity of the emotion
emotionText	Fragment of the EmotionSet's source that contained emotion information

Table 1: Main properties in the ontology. The specification contains the full description of all properties: `http://www.gsi.dit.upm.es/ontologies/onyx`

which together might be interpreted as remorse. Some representations would refer to remorse as a complex emotion. Onyx purposely does not include the notion of complex emotions. It follows the same approach as EmotionML in this respect, as HUMAINE EARL included this distinction between simple and complex emotions, but it was not included in the EmotionML specification. This simplifies the ontology and avoids discussion about the definition of complex emotions, since there are several possible definitions of a complex emotion, and different levels of emotions (e.g. the Hourglass of Emotions model). One possible way to deal with such situation is to add an *AggregatedEmotion* that represents remorse to the *EmotionSet*, linking it to the primary emotions with the *aggregatesEmotion* property.

Table 1, contains a comprehensive list of the properties associated with each of these classes. Figure 3 shows a complete overview of all these classes and their properties.

To group all the attributes that correspond to a specific emotion model, we created the *EmotionModel* class. Each EmotionModel will be linked to the different categories it contains (*hasEmotionCategory*), the *Appraisal* or *Dimension* instances it introduces (through *hasAppraisal* and *hasDimension*), etc.

Having a formal representation of the categories and dimensions proves very useful when dealing with heterogeneous datasets in emotion analysis. In addition to being necessary to interpret the results, this information can be used to filter out results and for automation.

3.2. Vocabularies

An *EmotionModel*, or at least an *EmotionCategory*, has to be defined in order to make a valid annotation. Annotators can define their own ad-hoc models and categories, but the Linked Data approach dictates that vocabularies and entities should be reused when appropriate. Hence, we offer several *EmotionModel* vocabularies that can be used with Onyx.

As of this writing, we have modeled the quite extensive WordNet-Affect taxonomy as an *EmotionModel*, to be used as the reference for categorical representation. We also ported the main vocabularies defined for EmotionML [21], and created a model based on the The Hourglass of Emotions [11]. A list of vocabularies with a detailed explanation is publicly available[3]. Onyx's Github repository[4] contains the tools used to generate all these models.

3.2.1. WordNet-Affect

WordNet-Affect [16] contains a subset of synsets suitable to represent affective concepts. Each synset is given one or more affective labels (a-labels) or categories. These labels labels are linked via concept/superconcept relationships. We processed the list of labels in WordNet-Affect 1.1 and generated a SKOS taxonomy. This taxonomy and its specification are available on our website[5]. In this specification we included a navigable tree with all the affects and their relationships. This tree makes it trivial to select an affect that represents the desired emotion. Figure 4 shows part of the tree, with some nodes collapsed.

The full taxonomy contains 305 affects, 291 of which are related to emotions. The RDF version of the taxonomy also includes an *EmotionModel* that contains these 291 affects as *EmotionCategory* entities.

Besides providing a good starting point for other ontologies, the resulting taxonomy also serves as reference for mapping between several different ontologies in the future.

3.2.2. EmotionML

EmotionML does not include any emotion vocabulary in itself. However, the Multimodal Interaction Working Group released a series of vocabularies that cover the most frequent models of emotions [21]. Users can either define their own vocabularies or reuse one of the existing ones.

We have developed a tool that generates an *EmotionModel* model from a vocabulary definition, including all its dimension, category, appraisal or action tendency entries. Using this tool, we have processed the vocabularies released by the Multimodal Interaction Working Group.

EmotionML has four types of vocabularies, according to the type of characteristic of the emotion phenomenon they represent: emotion categories, emotion

[3] `http://www.gsi.dit.upm.es/ontologies/onyx/vocabularies`
[4] `https://github.com/gsi-upm/onyx`
[5] `http://gsi.dit.upm.es/ontologies/wnaffect/`

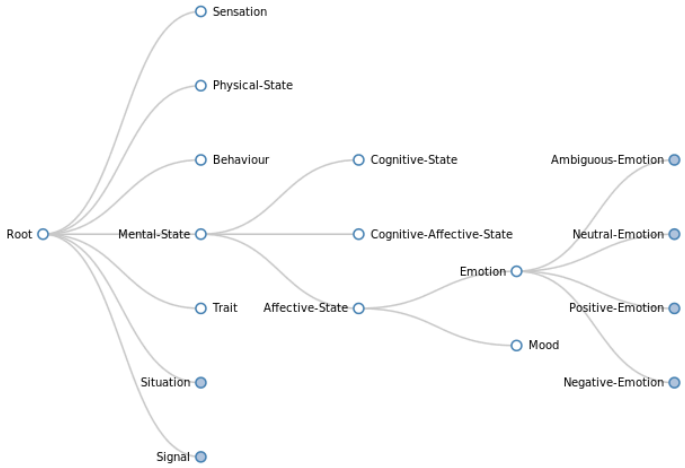


Figure 4: Small part of the full WordNet-Affect labels taxonomy. Blue nodes are contracted.

Listing 1: Excerpt of the models from EmotionML in Onyx

```
emoml:big6 a onyx:EmotionModel ;
    onyx:hasEmotionCategory emoml:big6_anger,
                            emoml:big6_disgust,
                            emoml:big6_fear,
                            emoml:big6_happiness,
                            emoml:big6_sadness,
                            emoml:big6_surprise .

emoml:ema a onyx:EmotionModel ;
    onyx:hasAppraisal emoml:ema_adaptability,
                        emoml:ema_agency,
                        emoml:ema_blame,
    ...

emoml:frijda a onyx:EmotionModel ;
    onyx:hasActionTendency emoml:frijda_agonistic,
                           emoml:frijda_approach,
                           emoml:frijda_attending,
    ...
    onyx:hasEmotionCategory emoml:frijda_anger,
                            emoml:frijda_arrogance,
                            emoml:frijda_desire,

emoml:pad a onyx:EmotionModel ;
    onyx:hasDimension emoml:pad_arousal,
                            emoml:pad_dominance,
                            emoml:pad_pleasure .
```

dimensions, appraisals and action tendencies. If an emotion model addresses several of these characteristics, there will be an independent vocabulary for each. For instance, Frijda's model defines action tendencies and categories, which results in the *frijda-categories* and *frijda-action-tendencies vocabularies*. In Onyx, instead of following this approach, we opted for adding all characteristics in the same model. This results in cleaner URIs, and helps represent the emotion model as a whole.

With these vocabularies it is possible to translate EmotionML resources into Onyx for their use in the Semantic Web. Table 2 contains an example of how EmotionML resources can be translated to Onyx.

EmotionML

```
<emotionml
  xmlns="http://.../emotionml"
  xmlns:meta="http://.../metadata"
  category-set="http://.../#everyday-categories">
<info>
  <classifiers:classifier
    classifiers:name="GMM"/>
</info>
<emotion>
 <category name="Disgust" value="0.82"/>
  'Come, there is no use in crying like that!'
</emotion>
  said Alice to herself rather sharply;
<emotion>
 <category name="Anger" value="0.57"/>
  'I advise you to leave off this minute!'
</emotion>
</emotionml>
```

Onyx

```
:Set1 a onyx:EmotionSet;
  onyx:extractedFrom "Come, there is no use in crying like that! said Alice
      to herself rather sharply; I advice you to live off this minute!";
  onyx:hasEmotion :Emo1
  onyx:hasEmotion :Emo2
:Emo1 a onyx:Emotion;
  onyx:hasEmotionCategory
    emoml:disgust;
  onyx:hasEmotionIntensity 0.82;
  onyx:hasEmotionText "Come, there's no use in crying like that!"
:Emo2 a onyx:Emotion;
  onyx:hasEmotionCategory emoml:anger;
  onyx:hasEmotionIntensity 0.57;
  onyx:hasEmotionText "I advice you to leave off this minute!"
:Analysis1 a onyx:EmotionAnalysis;
  onyx:algorithm "GMM";
  onyx:usesEmotionModel emoml:everyday-categories;
  prov:generated Set1.
```

Table 2: Example translation of an EmotionML resource to Onyx. emoml: <`http://www.gsi.dit.upm.es/ontologies/onyx/vocabularies/emotionml/ns#`>

3.2.3. The Hourglass of Emotions

The Hourglass of Emotions [11] is an interesting example of mixed models, including both dimensions and categories. Based on the paper by Cambria et al. we have created a basic model in Onyx, which includes the four dimensions and

24 first-level emotions, and 32 second-level emotions. Using the generated Onyx vocabulary, we can perform simple experiments with this interesting model. Nonetheless, a complete representation would include the relationships between the different categories and the dimensions, or restrictions.

Listing 2: Excerpt from the definition of the Hourglass of Emotions model in Onyx

```
hg:HourglassModel a onyx:EmotionModel ;
    onyx:hasDimension hg:Pleasantness ,
                      hg:Attention ,
                      hg:Sensitivity ,
                      hg:Aptitude ;
    onyx:hasEmotionCategory hg:ecstasy ,
                      hg:vigilance ,
                      hg:rage ,
    ...
                      hg:coercion .

hg:Pleasantness a onyx:EmotionDimension.
hg:Attention a onyx:EmotionDimension.
hg:Sensitivity a onyx:EmotionDimension.
hg:Aptitude a onyx:EmotionDimension.

hg:ecstasy a onyx:EmotionCategory.
hg:vigilance a onyx:EmotionCategory.
hg:rage a onyx:EmotionCategory.
...
hg:coercion a onyx:EmotionCategory.
```

3.3. Examples

After this introduction to the ontology, we will present several use cases for it. The examples should give a better understanding of the whole ontology. Rather than exhaustive and complex real life applications, these examples are simple self-contained showcases of the capabilities of semantic emotion annotation using Onyx. For the sake of brevity, we will omit the prefix declaration in the examples.

4. Applications

This section examines how Onyx has been applied in several use cases, such as modeling and publication of emotion lexicons (Section 4.1), annotation of emotion in corpora (Sect. 4.2), enabling interoperability of emotion services (Sect. 4.3) or mapping and composition heterogeneous emotion representations (Section 4.4).

4.1. Emotion annotation of lexical entries

Annotated lexical resources are the bases for analysis of emotions in text. These resources currently available can be classified into opinion and emotion lexicons.

Opinion lexicons supply a polarity value for a given lexical entry. Some examples of opinion lexicons are SentiWordNet [36], SenticNet 1.0 [37], Multi-Perspective Question Answering (MPQA) [38] or GermanPolarityClues [39].

Case	Turtle Representation

An example Emotion-Analysis.	`:customAnalysis`
	`  a onyx:EmotionAnalysis;`
	`  onyx:algorithm "SimpleAlgorithm";`
	`  onyx:usesEmotionModel wna:WNAModel.`

Processing "I lost one hour today because of the strikes!!", by the user JohnDoe	`:result1`
	`  a onyx:EmotionSet;`
	`  prov:wasGeneratedBy :customAnalysis;`
	`  sioc:has_creator [`
	`  sioc:UserAccount`
	`  <http://blog.example.com/JohnDoe>. ];`
	`  onyx:hasEmotion [`
	`  onyx:hasEmotionCategory wna:anger;`
	`  onyx:hasEmotionIntensity :0.9 ];`
	`  onyx:emotionText "I lost one hour today because of the strikes!!" ;`
	`  dcterms:created "2013-05-16T19:20:30+01:00"^^dcterms:W3CDTF.`

Example of annotation of a lexical entry using Onyx and *lemon* [35].	`:fifa`
	`  a lemon:Lexicalentry;`
	`  lemon:sense [`
	`  lemon:reference wn:synset-fear-noun-1;`
	`  onyx:hasEmotion`
	`  [ oyx:hasEmotionCategory wna:fear. ].`
	`  ];`
	`  lexinfo:partOfSpeech lexinfo:noun.`

Table 3: Representation with Onyx

Case	Query

Finding all the users that did not feel good during last New Year's Eve, and the exact emotions they felt.	`SELECT DISTINCT ?creator ?cat`
	`WHERE {`
	`  ?set onyx:hasEmotion`
	`  [ onyx:hasEmotionCategory ?cat];`
	`  dcterms:created ?date;`
	`  sioc:has_creator ?creator.`
	`  ?cat skos:broaderTransitive* wna:negative-emotion.`
	`  FILTER( ?date >= xsd:date("2012-12-31") ? date <= xsd:date("2013-01-01") )`
	`}`

Comparing two Emotion Mining algorithms by comparing the discrepancies in the results obtained using both.	`SELECT ?source1 ?algo1 (GROUP_CONCAT(?cat1) as ?cats1)`
	`WHERE {`
	`  ?set1 onyx:extractedFrom ?source1.`
	`  ?analysis1 prov:generated ?set1;`
	`  onyx:algorithm ?algo1.`
	`  ?set1 onyx:hasEmotion`
	`  [ onyx:hasEmotionCategory ?cat1 ].`
	`  FILTER EXISTS{`
	`  ?set2 onyx:extractedFrom ?source1.`
	`  ?analysis2 prov:generated ?set2.`
	`  ?set2 onyx:hasEmotion`
	`  [ onyx:hasEmotionCategory ?cat2 ].`
	`  FILTER ( ?set1 != ?set2).`
	`  FILTER ( ?cat2 != ?cat1 ).`
	`  }`
	`}`
	`GROUP BY ?source1 ?algo1`
	`ORDER BY ?source1`

Table 4: Example SPARQL queries with Onyx

Emotion lexicons supply an emotion description for a given lexical entry. Some examples of emotion lexicons are WordNet-Affect [16], the Affective Norms for English Words (ANEW) [40], EmoLex [41] or SenticNet 3.0 [42].

In this section we present the application of Onyx for annotating lexical entries with emotion representations. This application has been carried out in the context of the Eurosentiment R&D project where a LLOD approach has been followed [43]. In particular, lexical entries described with the *lemon* model have been extended with sentiment and emotion features using Marl and Onyx, respectively, as illustrated in Listing 3.

Listing 3: Lexicon expressed in *lemon* extended with sentiment and emotion features.

```
le:literature_en a lemon:Lexicon;
   lemon:language "en";
   lemon:entry lee:book, lee:terrifying.

lee:sense/terrifying_3 a lemon:Sense;
   lemon:reference dbp:Horror_fiction;
   lemon:reference "00111760";
   lexinfo:partOfSpeech lexinfo:adjective;
   lemon:context  lee:sense/book;
   marl:hasPolarity marl:Positve;
   marl:polarityValue 0.7;
   onyx:hasEmotionSet [
       onyx:hasEmotion [
           onyx:hasEmotionCategory wna:fear;
           onyx:hasEmotionIntensity 1.
       ] ;
   ] .

lee:book a lemon:LexicalEntry;
   lemon:canonicalForm [ lemon:writtenRep "book"@en ];
   lemon:sense [ lemon:reference wn:synset-book-noun-1;
                 lemon:reference dbp:Book. ];
   lexinfo:partOfSpeech lexinfo:noun.

lee:terrifying a lemon:LexicalEntry;
   lemon:canonicalForm [ lemon:writtenRep "terrifying"@en ];
   lemon:sense  lee:sense/terrifying_3;
   lexinfo:partOfSpeech lexinfo:adjective.
```

In this example, we illustrate how a lexical entry (terrifying) has a positive polarity value (0.7) and a emotion category (fear) when referring to the word book. In other words, a terrifying book is linked to feeling fear, but it usually represents a positive quality. In contrast, terrifying is a negative quality when referred to news. For a more in-depth explanation of the format, see [44, 43].

The main benefits of this approach are:

- Lexical entries are aligned with WordNet [26]. This enables interoperability with other affect lexicons, such as WordNet-Affect, SentiWordNet or GermanPolarityClues are also aligned with WordNet. Moreover, WordNetDomains has been used to define the domain of the lexical entry.
- Lexical entries are aligned with Linked Data entities in datasets such as DBPedia. None of the reviewed affect lexicons provide this feature yet, except for SenticNet 3.0.
- Most of the affect lexicons assign prior polarities or emotions to lexical entries, with the exception of SenticNet, that uses multi-word expressions (i.e. small room). The use of multi-word expressions is also possible in

Lexicons		
Language	Domains	#Entities
German	General	107417
English	Hotel,Electronics	8660
Spanish	Hotel,Electronics	1041
Catalan	Hotel,Electronics	1358
Portuguese	Hotel,Electronics	1387
French	Hotel,Electronics	651

Table 5: Summary of the lexicons in the LRP

Corpora		
Language	Domains	#Entities
English	Hotel,Electronics	22373
Spanish	Hotel,Electronics	18191
Catalan	Hotel,Electronics	4707
Portuguese	Hotel,Electronics	6244
French	Electronics	22841

Table 6: Summary of the corpora in the LRP

lemon (e.g. Siamese_cat), which provides a formalism for describing its decomposition into their component words.

- Onyx is extensible and the same formalism can be used for different emotion descriptions and it is not tied to a particular emotion representation, as the other affect lexicons. Following the EmotionML model, Onyx uses pluggable ontologies (vocabularies) that complement the central ontology.

This formalism has been used to generate the Eurosentiment dataset as detailed in [44, 45]. This dataset is composed of fourteen domain-specific opinion and emotion lexicons covering six languages (German, English, Spanish, Catalan, Portuguese and French) and two domain (Hotels and Electronics) as shown in Table 5.

4.2. Emotion annotations in corpora

One common use case for affective technologies [19] is the annotation of material involving emotionality, such as texts, videos or speech recordings.

Onyx has been used for emotion annotation of corpora as described in [44, 46, 45][6]. An overview of this lexical resources is provided in Table 6.

These corpora were used as gold standard for the evaluation of the services in the Eurosentiment [45] pool. The evaluation material is publicly available[7] and can be used as an example of evaluating semantic emotion analysis services using semantic resources in Onyx.

[6] A tool for translating other formats to Eurosentiment is available at `http://eurosentiment.readthedocs.org/en/latest/corpusconverter.html`

[7] `https://github.com/EuroSentiment/evaluation`

4.3. Emotion service specification

Defining a common service API and format is important for interoperability between services and to boost the ecosystem of emotion analysis services. This section shows how this can be achieved through a combination of NIF, originally intended for NLP services, and Onyx for annotation of emotion.

NLP Interchange Format (NIF) 2.0 [30] defines a semantic format and API for improving interoperability among natural language processing services. The classes to represent linguistic data are defined in the NIF Core Ontology. All ontology classes are derived from the main class *nif:String* which represents strings of Unicode characters. One important subclass of *nif:String* is the *nif:Context* class. It represents the whole string of the text and is used to calculate the indices of the substrings. There are other classes (*nif:Word*, *nif:Sentence*, *nif:Phrase*) for representing partitions of a text. NIF individuals are identified by URIs following a *nif:URIScheme* which restricts URI's syntax. NIF can be extended via vocabularies modules. It uses Marl for sentiment annotations and Onyx have been proposed as a NIF vocabulary for emotions.

In addition, NIF defines an input and output format for REST web services in the NIF 2.0 public API specification. This specification defines a set of parameters that should be supported by NIF compliant services.

Listing 4 shows the output of a service call with the input parameter value "My IPad is an awesome device"[8]

The main benefit of using NIF for emotion services is that NIF compliant services can be easily combined as shown in the NIF combinator [47][9], where well known NLP tools are combined thanks to the use of NIF wrappers.

To demonstrate the creation of an emotion analysis service using this specification, we developed a proof-of-concept web service on top of the Synesketch [48] library.

Synesketch is a library and application that detects emotions in English texts and generates images that reflect those emotions. Originally written in Java, it has been unofficially ported to several programming languages (including PHP), which shows the interest of the community in this tool. The aim of the PHP port was, among others, to offer a public endpoint for emotion analysis, which later had to be taken down due to misuse. The relevance of this tool and its Open Source license were the leading factors in choosing this tool. The service can be accessed via a REST API and its results are presented in Onyx, using the RDF format.

The Synesketch library uses the six categories of emotion proposed by Paul Ekman [9]. This model is included among the vocabularies of EmotionML under the name big6, and can be represented in Onyx as shown in Section 3.2.2. Each emotion is present in the input text with a certain weight that ranges from 0 to 1. Additionally, it has two attributes more that correspond to the general emotional valence (positive, negative or neutral) and the general emotional weight. These two attributes together show how *positive*, *negative* or *neutral* the overall emotion is.

[8]More details about the service output of sentiment and emotion services based on NIF, Marl and Onyx can be found at http://eurosentiment.readthedocs.org/en/latest/format/servicesformat.html

[9]The NIF combinator is available at http://demo.nlp2rdf.aksw.org/

Listing 4: Service output expressed in NIF using the vocabulary Onyx.

```
{
  "@context": [ "http://example.com/context.jsonld" ],
    "analysis": [{
      "@id": "http://example.com/analyse",
      "@type": [ "onyx:EmotionAnalysis" ],
      "dc:language": "en",
      "onyx:maxEmotionIntensity": 1.0,
      "onyx:minEmotionIntensity": 0.0
      "prov:wasAssociatedWith": "http://www.gsi.dit.upm.es/"
          }],
    "entries": [{
      "@id": "http://example.com/analyse?input=My%20ipad%20is
          %20an%20awesome%20device",
      "dc:language": "es",
      "opinions": [],
      "emotions": [{
        "onyx:aboutObject": "http://dbpedia.org/page/IPad"
        "prov:generatedBy": "http://example.com/analyse",
        "onyx:hasEmotion": [{
          "onyx:hasEmotionCategory": "wna:liking",
          "onyx:hasEmotionIntensity": 0.7
          }, {
          "onyx:hasEmotionCategory": "wna:approval",
          "onyx:hasEmotionIntensity": 0.1 }]}],
      "nif:isString": "My ipad is an awesome device",
      "prov:generatedBy": "http://example.com/analyse",
      "strings": [{
        "@id": "http://example.com/analyse?input=My%20ipad%20
            is%20an%20awesome%20device#char=3,6",
        "itsrdf:taIdentRef": "http://dbpedia.org/page/IPad",
        "nif:anchorOf": "ipad" }]}]
}
```

The Synesketch weight directly maps to *hasEmotionIntensity* in Onyx. However, the general emotional valence and weight do not directly match any Onyx property or class. To solve it, we simply add an *AggregatedEmotion* to the *EmotionSet* with the *PositiveEmotion*, *NeutralEmotion* or *NegativeEmotion* category (as defined by WordNet-Affect) depending on the value of the valence. The general emotional weight is then the intensity of this *AggregatedEmotion*, just like in the other cases.

The final result is a REST service that is publicly available at our website[10].

Several other implementations have been developed in the context of the Eurosentiment project, mainly as wrappers of already existing resources. The majority of them uses the WordNet-Affect categories, although there are some that required the specification of ad-hoc categories.

4.4. Emotion mapping and rules

Some of the potential benefits of the use of ontologies [49] are: mapping between different emotion representations, the definition of the relationship between concepts in an emotion description and emotion composition. Each of these topics is a subject of study in its own right. However, we want to illustrate how Onyx's semantic approach could be used in this direction. In particular, we will focus on SPARQL Inference Notation (SPIN)[11] rules.

[10] `http://demos.gsi.dit.upm.es/onyxemote/`
[11] `http://spinrdf.org/spin.html`

SPIN is a powerful tool to create rules and logical constraints to any entity using standard SPARQL queries. It does this by using a set of RDF classes and properties defined in its specification. For our purposes, the simplest use of SPIN is to attach a SPARQL query (rule from now on) to a certain class, so that it is performed on entity creation, update or deletion. What is interesting about SPIN is that it has mechanisms to generalize this procedure, and to create templates from these rules so that they can be applied to several classes. An Open Source Java API[12] is also publicly available which can be used to test the examples in this section.

Exploring the full potential of SPIN is out of the scope of this paper. However, we will cover two ways in which Onyx and SPIN can be used together to provide more flexibility than any non-semantic approach could. For the sake of clarity, we will only include the relevant rules that should be used in each case, rather than the complete SPIN RDF excerpt.

The first example (Listing 5) shows the rule that should be added to the *EmotionSet* class in order to infer a complex Plutchik emotion from two basic ones. In particular, it annotates an emotion with the Optimism category if it is already annotated with Anticipation and Joy. The query also shows how to add new entities and specifying their URIs using a random identifier and the base URI.

Listing 5: Automatic composition of emotions. Anticipation and Joy result in Optimism.

```
INSERT {
    ?this onyx:hasEmotion ?emotion .
    ?emotion a onyx:Emotion .
    ?emotion a onyx:AggregatedEmotion .
    ?emotion onyx:hasEmotionCategory plutchik:Optimism .
}
WHERE {
    ?this onyx:hasEmotion _:0 .
    ?this onyx:hasEmotion _:1 .
    _:0 onyx:hasEmotionCategory plutchik:Joy .
    _:1 onyx:hasEmotionCategory plutchik:Anticipation .
    BIND (IRI(CONCAT("http://example.org/id/#Emotion", SUBSTR
        (str(RAND()), 3, 16))) AS ?emotion) .
}
```

The second example (Listing 6) assigns an emotional category based on dimensional values. It takes as an example the Hourglass of Emotions [11] model where second-level emotions can be expressed as a combination of two sentic levels. In this case, a positive level of attention and pleasantness results in optimism. Its dimensions and the main emotional categories have been represented with Onyx (Section 3.2.3).

Finally, by using these two rules together and a simple mapping of both hourglass' and Plutchik's Optimism categories, it is possible to find optimistic results among entries annotated using Plutchik's basic categories and entries annotated using the dimensions from the Hourglass of Emotions.

[12] `http://topbraid.org/spin/api/`

Listing 6: Applying categories based on the dimensions of an emotion.

```
INSERT {
    ?this onyx:hasEmotionCategory hg:Optimism .
}
WHERE {
    ?this hg:Pleasantness ?p .
    ?this hg:Attention ?a .
    FILTER ( ?p > 0 && ?a > 0 )
```

5. Conclusions and future work

With this work we have introduced the Onyx ontology, which can represent emotions taking advantage of the work conducted in the Web of Data (Linked Data) and in emotion research community (EmotionML). The ontology is extendable through pluggable vocabularies that enable its adaptation to different emotion models and application domains.

This paper discusses several applications and use cases, such as the emotion annotation of lexical entries and corpora, the specification of inter-operable emotion analysis services as well as the definition of emotion mappings. To this end, Onyx has been linked with vocabularies such as *lemon*, NIF and the Provenance Ontology. One remarkable example is the role of Onyx in the R&D European project Eurosentiment[13], whose aim is the creation of a language resource pool for Sentiment Analysis. As a result, a set of lexical resources are publicly available annotated with Onyx.

A key factor for the adoption of proposals such as Onyx is its us and extension by the emotion research community. To this end, the W3C Community Group *Linked Data Models for Emotion and Sentiment Analysis*[14] has been set up with participants from industry and academia. We hope that it will contribute to the evolution of Onyx, shaping it to address the issues arising from feedback of the community. The next step for Onyx is to go beyond the applications in this paper (Section 4) and be leveraged in state-of-the-art sentiment analysis applications. As a start point, it could be used in any of the several sentiment analysis challenges available. In particular, it is a perfect fit for semantic sentiment analysis challenges such as SemEval[15].

For the most part, Onyx relies on the main elements introduced by EmotionML. That seems to be the case with HEO as well, which explains the similarities between both ontologies. Given these similarities, we will discuss some of the differences between both ontologies that justify the use of Onyx.

First and foremost, Onyx intends to be a model as generic as possible. It has been integrates with other ontologies, When in doubt about a property or concept, we chose to leave it outside the ontology and link to other vocabularies (e.g. OpenAnnotation[50], NIF [47]). It also integrates perfectly with the provenance ontology, adding great value to the ontology. Although HEO also provides a generic model of emotion, it is tailored to a specific use case and includes concepts that would now be better expressed with other ontologies.

[13] `http://eurosentiment.eu`

[14] `http://www.w3.org/community/sentiment/`

[15] SemEval's sentiment analysis track: `http://alt.qcri.org/semeval2015/index.php?id=tasks`

For instance, Open Annotation or Prov-O for annotation. The fact that Onyx follows an approach analogous to Marl also facilitates the integration of Opinion and Emotion, which is crucial for sentiment analysis.

There are other practical differences between the wo, such as action tendencies and appraisal being properties in Onyx. This reduces the number of nodes necessary for annotation, and makes annotations more convenient.

These differences prove that HEO and Onyx are actually very different. Not only on the ontological level, but also in approach. We believe Onyx was a missing piece in the Linked Data puzzle, it integrates with other ontologies and covers aspects that no other ontology for emotions did.

Moreover, the presented applications show the applicability and usefulness of the ontology. Using the concepts shown in Section 4.4, it would be possible to combine resources that use different emotion models. For instance, an application could leverage the power of WordNet-Affect [16], EmoLex [15] (Plutchik's categories) and DepecheMood [51] (Ekman's categories).

As future work, we will study how emotion synthesis and emotional embodied conversational agents can be applied in e-learning. In particular, our aim is to explore how conversational agents can benefit from integration of semantic, emotion and user models for improving user engagement. There are two interesting aspects in this sense. The first one is using a behavioral model based on emotions to interact with students on a deeper level. For this, we will build on our experience with conversational agents and make use of Onyx annotations to reason about emotions. The second one is trying to gain a better understanding of students. This analysis will build on the concepts in Section 4.4 to use various models that characterize different aspects of emotions.

6. Acknowledgements

This work was partially funded by the EUROSENTIMENT FP7 Project (Grant Agreement no: 296277). The authors want to thank the reviewers for their thorough and helpful feedback, which improved the quality of this paper. Lastly, we thank Paul Buitelaar and Gabriela Vulcu as well for their contribution to the representation of lexical information using *lemon*.

References

[1] B. Pang, L. Lee, Opinion mining and sentiment analysis, Foundations and trends in information retrieval 2 (1-2) (2008) 1–135.

[2] G. Tummarello, R. Delbru, E. Oren, Sindice. com: Weaving the open linked data, in: The Semantic Web, Springer, 552–565, 2007.

[3] A. Westerski, C. A. Iglesias, F. Tapia, Linked Opinions: Describing Sentiments on the Structured Web of Data, in: Proceedings of the 4th International Workshop Social Data on the Web, vol. Vol-830, 13–23, URL `onlineatCEUR-WS.org/Vol-830`, 2011.

[4] S. G. Barsade, The Ripple Effect: Emotional Contagion and its Influence on Group Behavior, Administrative Science Quarterly 47 (4) (2002) 644–675.

[5] A. D. Kramer, J. E. Guillory, J. T. Hancock, Experimental evidence of massive-scale emotional contagion through social networks, Proceedings of the National Academy of Sciences (2014) 201320040URL http://www.pnas.org/content/early/2014/05/29/1320040111.short.

[6] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, B. Bhattacharjee, Measurement and analysis of online social networks, in: Proceedings of the 7th ACM SIGCOMM conference on Internet measurement, IMC '07, ACM, New York, NY, USA, ISBN 978-1-59593-908-1, 29–42, 2007.

[7] E. Cambria, B. Schuller, Y. Xia, C. Havasi, New Avenues in Opinion Mining and Sentiment Analysis, Intelligent Systems, IEEE 28 (2) (2013) 15–21.

[8] M. Munezero, C. Montero, E. Sutinen, J. Pajunen, Are They Different? Affect, Feeling, Emotion, Sentiment, and Opinion Detection in Text, Affective Computing, IEEE Transactions on 5 (2) (2014) 101–111.

[9] P. Ekman, Basic emotions, Handbook of cognition and emotion 98 (1999) 45–60.

[10] J. J. Prinz, Gut reactions: A perceptual theory of emotion, Oxford University Press, 2004.

[11] E. Cambria, A. Livingstone, A. Hussain, The hourglass of emotions, in: Cognitive Behavioural Systems, Springer, 144–157, 2012.

[12] R. Plutchik, Emotion: A psychoevolutionary synthesis, Harper & Row New York, 1980.

[13] D. Borth, T. Chen, R. Ji, S.-F. Chang, SentiBank: large-scale ontology and classifiers for detecting sentiment and emotions in visual content, in: Proceedings of the 21st ACM international conference on Multimedia, MM '13, ACM, New York, NY, USA, ISBN 978-1-4503-2404-5, 459–460, URL http://doi.acm.org/10.1145/2502081.2502268, 2013.

[14] E. Cambria, C. Havasi, A. Hussain, SenticNet 2: A Semantic and Affective Resource for Opinion Mining and Sentiment Analysis., in: FLAIRS Conference, 202–207, 2012.

[15] S. M. Mohammad, P. D. Turney, Crowdsourcing a Word-Emotion Association Lexicon 29 (3) (2013) 436–465.

[16] C. Strapparava, A. Valitutti, WordNet-Affect: an affective extension of WordNet, in: Proceedings of LREC, vol. 4, 1083–1086, 2004.

[17] M. Schröder, H. Pirker, M. Lamolle, First suggestions for an emotion annotation and representation language, in: Proceedings of LREC, vol. 6, 88–92, 2006.

[18] M. Schröder, L. Devillers, K. Karpouzis, J.-C. Martin, C. Pelachaud, C. Peter, H. Pirker, B. Schuller, J. Tao, I. Wilson, What should a generic emotion markup language be able to represent?, in: Affective Computing and Intelligent Interaction, Springer, 440–451, 2007.

[19] P. Baggia, C. Pelachaud, C. Peter, E. Zovato, Emotion Markup Language (EmotionML) 1.0 W3C Recommendation, Tech. Rep., W3C, URL http://www.w3.org/TR/emotionml/, 2014.

[20] H. N. of Excellence, HUMAINE Emotion Annotation and Representation Language (EARL): Proposal., Tech. Rep., HUMAINE Network of Excellence, URL http://emotion-research.net/projects/humaine/earl/proposal#Dialects, 2006.

[21] K. Ashimura, P. Baggia, F. Burkhardt, A. Oltramari, C. Peter, E. Zovato, EmotionML vocabularies, Tech. Rep., W3C, URL http://www.w3.org/TR/2012/NOTE-emotion-voc-20120510/, 2012.

[22] J. Hastings, W. Ceusters, B. Smith, K. Mulligan, Dispositions and Processes in the Emotion Ontology, in: Proc. ICBO, 71–78, 2011.

[23] C. Chiarcos, S. Hellmann, S. Nordhoff, Towards a linguistic linked open data cloud: The Open Linguistics Working Group, TAL 52 (3) (2011) 245–275.

[24] C. Chiarcos, J. McCrae, P. Cimiano, C. Fellbaum, Towards open data for linguistics: Linguistic linked data, in: New Trends of Research in Ontologies and Lexical Resources, Springer, 7–25, 2013.

[25] P. Buitelaar, P. Cimiano, J. McCrae, E. Montiel-Ponsoda, T. Declerck, Ontology lexicalisation: The lemon perspective, in: Workshop at 9th International Conference on Terminology and Artificial Intelligence (TIA 2011), 33–36, 2011.

[26] C. Fellbaum, WordNet, Wiley Online Library, 1998.

[27] C. Bizer, J. Lehmann, G. Kobilarov, S. Auer, C. Becker, R. Cyganiak, S. Hellmann, DBpedia-A crystallization point for the Web of Data, Web Semantics: science, services and agents on the world wide web 7 (3) (2009) 154–165.

[28] C. Chiarcos, S. Hellmann, S. Nordhoff, Linking linguistic resources: Examples from the open linguistics working group, in: Linked Data in Linguistics, Springer, 201–216, 2012.

[29] C. Chiarcos, POWLA: Modeling linguistic corpora in OWL/DL, in: The Semantic Web: Research and Applications, Springer, 225–239, 2012.

[30] S. Hellmann, Integrating Natural Language Processing (NLP) and Language Resources using Linked Data, Ph.D. thesis, Universität Leipzig, 2013.

[31] C. Chiarcos, Ontologies of Linguistic Annotation: Survey and perspectives., in: LREC, 303–310, 2012.

[32] L. Moreau, B. Clifford, J. Freire, J. Futrelle, Y. Gil, P. Groth, N. Kwasnikowska, S. Miles, P. Missier, J. Myers, B. Plale, Y. Simmhan, E. Stephan, J. Van den Bussche, The Open Provenance Model core specification (v1.1), Future Generation Computer Systems 27 (6) (2011) 743 – 756, ISSN 0167-739X, URL http://www.sciencedirect.com/science/article/pii/S0167739X10001275.

[33] P. Groth, L. Moreau, Prov-O W3C Recommendation, Tech. Rep., W3C, URL http://www.w3.org/TR/prov-o/, 2013.

[34] M. Schröder, The SEMAINE API: Towards a Standards-Based Framework for Building Emotion-Oriented Systems, Advances in Human-Computer Interaction 2010.

[35] J. McCrae, D. Spohr, P. Cimiano, Linking Lexical Resources and Ontologies on the Semantic Web with Lemon, in: G. Antoniou, M. Grobelnik, E. Simperl, B. Parsia, D. Plexousakis, P. Leenheer, J. Pan (Eds.), The Semantic Web: Research and Applications, vol. 6643 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, ISBN 978-3-642-21033-4, 245–259, 2011.

[36] A. Esuli, F. Sebastiani, Sentiwordnet: A publicly available lexical resource for opinion mining, in: Proceedings of LREC, vol. 6, 417–422, 2006.

[37] E. Cambria, R. Speer, C. Havasi, A. Hussain, SenticNet: A Publicly Available Semantic Resource for Opinion Mining., in: AAAI Fall Symposium: Commonsense Knowledge, vol. 10, 02, 2010.

[38] J. Wiebe, T. Wilson, C. Cardie, Annotating expressions of opinions and emotions in language, Language resources and evaluation 39 (2-3) (2005) 165–210.

[39] U. Waltinger, GermanPolarityClues: A Lexical Resource for German Sentiment Analysis., in: Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC), 1638–1642, 2010.

[40] M. M. Bradley, P. J. Lang, Affective norms for English words (ANEW): Instruction manual and affective ratings, Tech. Rep., Citeseer, 1999.

[41] S. Mohammad, P. D. Turney, Crowdsourcing a Word-Emotion Association Lexicon, Computation Intelligence 29 (3) (2013) 436–465.

[42] E. Cambria, D. Olsher, D. Rajagopal, SenticNet 3: a common and commonsense knowledge base for cognition-driven sentiment analysis, in: Twenty-eighth AAAI conference on artificial intelligence, 1515–1521, 2014.

[43] P. Buitelaar, M. Arcan, C. A. Iglesias, J. F. Sánchez-Rada, C. Strapparava, Linguistic Linked Data for Sentiment Analysis, in: 2nd Workshop on Linked Data in Linguistics (LDL-2013): Representing and linking lexicons, terminologies and other language data, Pisa, Italy, 1–8, 2013.

[44] V. Gabriela, B. Paul, N. Sapna, P. Bianca, A. Mihael, C. Barry, S. J. Fernando, I. C. A., Generating Linked-Data based Domain-Specific Sentiment Lexicons from Legacy Language and Semantic Resources, in: Proceedings of the 5th International Workshop on Emotion, Social Signals, Sentiment and Linked Open Data (ES3LOD), co-located with LREC 2014, Reykjavik, Iceland, 6–9, 2014.

[45] J. F. Sánchez-Rada, G. Vulcu, C. A. Iglesias, P. Buitelaar, EUROSENTIMENT: Linked Data Sentiment Analysis, in: J. v. O. Matthew Horridge, Marco Rospocher (Ed.), Proceedings of the 13th International Semantic

Web Conference ISWC 2014 Posters and Demonstrations Track, 145–148, URL http://ceur-ws.org/Vol-1272/paper_116.pdf, 2014.

[46] V. Gabriela, B. Paul, N. Sapna, P. Bianca, A. Mihael, C. Barry, S. J. Fernando, I. C. A., Linked-Data based Domain-Specific Sentiment Lexicons, in: Proceedings of 3rd Worshop on Linked Data in Linguistics: Multilingual Knowledge Resources and Natural Language Processing, co-located with LREC 2014, 77–81, 2014.

[47] S. Hellmann, J. Lehmann, S. Auer, M. Nitzschke, NIF Combinator: Combining NLP Tool Output, in: Proceedings of the 18th International Conference on Knowledge Engineering and Knowledge Management, EKAW'12, Springer-Verlag, Berlin, Heidelberg, 446–449, 2012.

[48] U. Krcadinac, P. Pasquier, J. Jovanovic, V. Devedzic, Synesketch: An Open Source Library for Sentence-based Emotion Recognition, IEEE Transactions on Affective Computing 4 (3) (2013) 312–325.

[49] M. Schröder, The SEMAINE API: A Component Integration Framework for a Naturally Interacting and Emotianally Competent Embodied Conversational Agent, Ph.D. thesis, Universität des Saarlandes, 2011.

[50] P. Ciccarese, S. Soiland-Reyes, T. Clark, Web Annotation as a First-Class Object, Internet Computing, IEEE 17 (6) (2013) 71–75, ISSN 1089-7801.

[51] J. Staiano, M. Guerini, DepecheMood: a Lexicon for Emotion Analysis from Crowd-Annotated News, arXiv preprint arXiv:1405.1605 URL http://arxiv.org/abs/1405.1605, 00002.

3.1.3 Towards a Common Linked Data Model for Sentiment and Emotion Analysis

Title	Towards a Common Linked Data Model for Sentiment and Emotion Analysis
Authors	Sánchez-Rada, J. Fernando and Schuller, Björn and Patti, Viviana and Buitelaar, Paul and Vulcu, Gabriela and Bulkhardt, Felix and Clavel, Chloé and Petychakis, Michael and Iglesias, Carlos A.
Proceedings	Proceedings of the LREC 2016 Workshop Emotion and Sentiment Analysis (ESA 2016)
ISBN	
Year	2016
Keywords	emotion, linked data, sentiment
Pages	48–54
Online	http://www.lrec-conf.org/proceedings/lrec2016/workshops/LREC2016Workshop-ESA_Proceedings.pdf
Abstract	The different formats to encode information currently in use in sentiment analysis and opinion mining are heterogeneous and often custom tailored to each application. Besides a number of existing standards, there are additionally still plenty of open challenges, such as representing sentiment and emotion in web services, integration of different models of emotions or linking to other data sources. In this paper, we motivate the switch to a linked data approach in sentiment and emotion analysis that would overcome these and other current limitations. This paper includes a review of the existing approaches and their limitations, an introduction of the elements that would make this change possible, and a discussion of the challenges behind that change.

Towards a Common Linked Data Model for Sentiment and Emotion Analysis

J. Fernando Sánchez-Rada, Björn Schuller, Viviana Patti, Paul Buitelaar, Gabriela Vulcu, Felix Burkhardt, Chloé Clavel, Michael Petychakis, Carlos A. Iglesias,
Linked Data Models for Emotion and Sentiment Analysis W3C Community Group.
internal-sentiment@w3.org
jfernando, cif@dit.upm.es, Universidad Politécnica de Madrid, Spain
schuller@ieee.org, University of Passau, Germany and Imperial College London, UK
patti@di.unito.it, Dipartimento di Informatica, University of Turin, Italy.
paul.buitelaar,gabriela.vulcu@insight-centre.org, Insight Centre for Data Analytics at NUIG, Ireland
Felix.Burkhardt@telekom.de, Telekom Innovation Laboratories, Germany
chloe.clavel@telecom-paristech.fr, LTCI, CNRS, Télécom ParisTech, Université Paris-Saclay, France
mpetyx@epu.ntua.gr, National Technical University of Athens

Abstract

The different formats to encode information currently in use in sentiment analysis and opinion mining are heterogeneous and often custom tailored to each application. Besides a number of existing standards, there are additionally still plenty of open challenges, such as representing sentiment and emotion in web services, integration of different models of emotions or linking to other data sources. In this paper, we motivate the switch to a linked data approach in sentiment and emotion analysis that would overcome these and other current limitations. This paper includes a review of the existing approaches and their limitations, an introduction of the elements that would make this change possible, and a discussion of the challenges behind that change.

Keywords: sentiment, emotion, linked data

1. Introduction

As Internet access becomes ubiquitous, more and more websites and applications allow us to share our opinions with the rest of the world. This information has drawn the attention of researchers and industry alike. Researchers see this as an opportunity to collect information about society. For industry, it means quick and unobtrusive feedback from their customers. For private individuals, it can be of interest how the public "sentiment" towards them or their ideas, comments, and contributions reflect on the internet.

However, humans are not capable of processing the ever growing flow of information. As a consequence, sentiment and emotion analysis have received increased support and attention. Many tools that offer automated transformation of unstructured data into structured information have emerged. The provided content analysis functionalities may vary from brand impact based on its social media presence, trend analytics possibly accompanied with predictions for future trends, sentiment identification over a brand or a product.

Unfortunately, the different formats to encode information currently in use are heterogeneous and often custom tailored to each application. The biggest contender is Emotion Markup Language (EmotionML) (see Sec. 4.1.). EmotionML provides a common representation in many scenarios and has been widely adopted by the community.

However, there are still plenty of open challenges not fully covered by EmotionML, as it was solely developed to represent emotional states on the basis of suggested and user-defined vocabularies. Sentiment analysis has not been one of the 39 use cases that motivated EmotionML[1]. Also, a bridge to the semantic web and linked data has been discussed, but been postponed due to the neccessity to reduce complexity for the first version.

In this paper, we motivate the switch to a linked data approach in sentiment analysis that would overcome these and other current limitations. We introduce the elements that would make this change possible and discuss the challenges behind that change.

The rest of this paper is structured as follows. Section 2. contains a brief overview of the terminology in the field; Section 3. introduces the main applications of Sentiment and Emotion Analysis; Section 4. briefly discusses the state of the art in data representation and formats in sentiment analysis; Section 5. presents recent public projects related to sentiment and emotion analysis in any modality; Section 6. explains how a linked data approach would allow more complex applications of sentiment analysis; Section 7. reviews current models and formats that a common linked data representation could be based on; Section 8. exemplifies how current applications would highly benefit from a linked data approach; finally, we draw conclusions from the above.

2. Terminology

The literature of natural language processing differs from the one of affective computing in the terminology used for defining opinion/sentiment/emotion phenomena(Clavel and Callejas, 2015). Indeed, the natural language processing community more frequently uses opinion, sentiment and affect while the affective computing community tends to prefer the word emotion and provides in-depth studies of the term emotion and its specificity according to other linked phenomena such as moods, attitudes, affective dispositions and interpersonal stances (Scherer, 2005). The

[1] https://www.w3.org/2005/Incubator/emotion/XGR-emotion/#AppendixUseCases

distinction between opinion, sentiment and affect is not always clear in the Natural Language Processing (NLP) community (Ishizuka, 2012). Some studies consider sentiment analysis in a broader sense including the analysis of sentiments, emotions and opinions (Chan and Liszka, 2013; Ortigosa et al., 2014a) and consider positive vs. negative distinction as the study of sentiment polarity. Other studies consider sentiment as the affective part of opinions (Kim and Hovy, 2004). Another point of view is also given in Krcadinac et al. (Krcadinac et al., 2013) which states that sentiment analysis concerns positive vs. negative distinction while affect analysis or emotion recognition focus on more fine-grained emotion categories. However, we can refer to Munezero (Munezero et al., 2014) for in-depth reflections of the differences between affect, emotion, sentiment and opinion from a NLP point of view. To sum up, they claim that affects have no expression in language, that emotions are briefer than sentiment and that opinions are personal interpretations of information and are not necessarily emotionally charged unlike sentiments. Other approaches (Martin and White, 2005) prefer to use the general term attitudes to gather three distinct phenomena: affect (personal reaction referring to an emotional state), judgment (assigning quality to individuals according to normative principles) and appreciations (evaluation of an object, e.g. a product or a process).
In the scope of this paper, we use the term 'Sentiment and Emotion Analysis' to cover the range of tecniques to detect subjectivity and emotional state.

3. Applications of Emotion and Sentiment Analysis

Sentiment analysis is now an established field of research and a growing industry (Liu, 2012). There are many applications for sentiment analysis as well as for emotion analysis. It is often used in social media monitoring, tracking customer attitudes towards brands, towards politicians etc Moreover, it is also practical for use in business analytics. Sentiment analysis is in demand because of its efficiency and it can provide an quick overview based on the analysis of humanly impossible to analyse data sources. Thousands of text documents can be processed for sentiment in terms of seconds as opposed to large amounts of time humans would need to make sense out of hotel reviews for example.
Below we categorize the sentiment analysis application in different areas of service. At the public service level we look at sentiment analysis approaches for e-learning systems, tracking opinions about politicians and identification of violent social movements in social media. For businesses and organizations sentiment analysis is used in products benchamrcking, brand reputation and ad placement. From the individual's perspective we are looking at decision making based on opinions about products and services as well as identifying communities and individuals with similar interests and opinions.

1. Public service
   (a) E-learning environments (Ortigosa et al., 2014b): Sentiment and emotion analysis information can be used by adaptive e-learning systems to support personalized learning, by considering the user's emotional state when recommending him/her the most suitable tasks to be tackled at each time. Also, the students' sentiments towards a course serve as useful feedback for teachers.
   (b) Tracking public opinions about political candidates: Recently, with every political campaign, it has become a standard practice to see the public opinion from social media or other sources about each candidate.
   (c) Radicalization and recruitment detection (Zimbra and Chen, 2012): Sentiment analysis is used for detection of violent social movement groups.
2. Businesses and organizations
   (a) Market analysis and benchmark products and services: Businesses spend a huge amount of money to find consumer opinions using consultants, surveys and focus groups, etc
   (b) Affective user interfaces (Nasoz and Lisetti, 2007): An example is in the automotive domain where human-computer interaction is enhanced through Adaptive Intelligent User Interfacesthat are able to recognize users' affective states (i.e., emotions experienced by the users) and responding to those emotions by adapting to the current situation via an affective user model.
   (c) Ads placements: A popular way of monetize online is add placement. Sentiment and emotion analysis is exploited in various ways to a) place ads in key social media content, b) place ads if one praises a product or c) place ads from a competitor if one criticizes a product.
3. Individuals
   (a) Make decisions to buy products or to use services.
   (b) Find collectives and individuals with similar interests and opinions.

4. State of the Art

This section introduces works that are relevant either because they aim to provide a common language and framework to represent emotional information (as is the case of EmotionML), or because they they provide a specific representation of affects and emotions.

4.1. EmotionML

EmotionML (Burkhardt et al., 2016) is W3C recommendation to represent emotion related states in data processing systems. It was developed as a XML schema by a subgroup of the W3C MMI (Multimodal Interaction) Working Group chaired by Deborah Dahl in a first version from approximately 2005 until 2013, most of this time the development was lead by Marc Schröder. It is possible to use EmotionML both as a standalone markup and as a plug-in annotation in different contexts. Emotions can be represented

in terms of four types of descriptions taken from the scientific literature: categories, dimensions, appraisals, and action tendencies, with a single <emotion> element containing one or more of such descriptors. The following snippet exemplifies the principles of the EmotionML syntax.

```
<graysentencered redidred=blue"
    bluesent1blue"black>
blackDoblack blackIblack blackhave
    black blacktoblack blackgoblack
    blacktoblack blacktheblack
    blackdentistblack?
black</graysentenceblack>
black<grayemotionred redxmlnsred=blue"
    bluehttpblue://bluewwwblue.bluew3
    blue.blueorgblue/2009/10/
    blueemotionmlblue"red redcategory
    red-redsetred=blue"bluehttp
    blue://.../bluexmlblue#blueeveryday
    blue-bluecategoriesblue"black>
black<graycategoryred rednamered=blue"
    blueafraidblue"red redvaluered=
    blue"blue0.4blue"/black>
black<grayreferencered redrolered=blue"
    blueexpressedByblue"red redurired=
    blue"blue#bluesent1blue"/black>
black</grayemotionblack>
```

Since there is no single agreed-upon vocabulary for each of the four types of emotion descriptions, EmotionML provides a mandatory mechanism for identifying the vocabulary used in a given <emotion>. Some vocabularies are suggested by the W3C (Ashimura, Kazuyuki et al., 2014) and to make EmotionML documents interoperable users are encouraged to use them.

4.2. WordNet Affect

WordNet Affect (Strapparava et al., 2004) is an effort to provide lexical representation of affective knowledge. It builds upon WordNet, adding a new set of tags to a selection of synsets to annotate them with affective information. The affective labels in WordNet Affect were generated through a mix of manual curation and automatic processing. Labels are related to one another in the form of a taxonomy. Then, a subset of all WordNet synsets were annotated with such labels, leveraging the structure and information of WordNet. Hence, the contribution of WordNet Affect is twofold: a rich categorical model of emotions based on WordNet, and the linking of WordNet synsets to such affects.

4.3. Chinese Emotion Ontology

The Chinese Emotion Ontology (Yan et al., 2008) was developed to help understand, classify and recognize emotions in Chinese. The ontology is based on HowNet, the Chinese equivalent of WordNet. The ontology provides 113 categories of emotions, which resemble the WordNet taxonomy and the authors also relate the resulting ontology with other emotion categories. All the categories together contains over 5000 Chinese verbs.

4.4. Emotive Ontology

Sykora et al. (Sykora et al., 2013) propose an ontology-based mechanism to extract fine-grained emotions from informal messages, such as those found on Social Media.

5. Relevant Projects

This section presents some recent note-worthy projects linked to emotion or sentiment analysis in any of its different modalities.

5.1. ArsEmotica

ArsEmotica (Bertola and Patti, 2016) is an application framework where semantic technologies, linked data and natural language processing techniques are exploited for investigating the emotional aspects of cultural heritage artifacts, based on user generated contents collected in art social platforms. Tha aim of ArsEmotica is to detect emotion evoked by artworks from online collections, by analyzing social tags intended as textual traces that visitors leave for commenting artworks on social platforms. The approach is ontology-driven: given a tagged resource, the relation with the evoked emotions is computed by referring to an ontology of emotional categories, developed within the project and inspired by the well-known Plutchik's model of human emotions (Plutchik and Conte, 1997). Detected emotions are meant to be the ones which better capture the affective meaning that visitors, collectively, give to the artworks. The ArsEmotica Ontology (AEO) is encoded in OWL and incorporates, in a unifying model, multiple ontologies which describe different aspects of the connections between media objects (e.g. artworks), persons and emotions. The ontology allows to link art reviews, or excerpts thereof, to specific emotions. Moreover, due to the need of modeling the link among words in a language and the emotions they refer to, AEO integrates with LExical Model for Ontologies (lemon) to provide the lexical model (Patti et al., 2015). Where possible and relevant, linkage to external repositories of the LOD (e.g. DBpedia) is provided.

5.2. EuroSentiment

The aim of the EuroSentiment project [2] was to provide a shared language resource pool, a marketplace dedicated to services and resources useful in multilingual Sentiment Analysis. The project focused on adapting existing lexicons and corpora to a common linked data format. The format for lexicons is based on a combination of lemon (for lexical concepts), Marl (opinion/sentiment) and Onyx (emotions). Each entry in the lexicon is described with part of speech information, morphosyntactic information, links to DBpedia and WordNet and sentiment information of the entry was identified as a sentiment word. The format for corpora uses NIF instead of lemon, while keeping the combination of Onyx and Marl for subjectivity. The results of the project include: a semantic enriching pipeline for lexical resources, a set of lexicons and corpora for sentiment and emotion analysis; conversion tools from legacy non-semantic formats; an extension of the NIF format and API for web services; and, lastly, the implementation of said

[2] http://eurosentiment.eu

API in different programming languages, which helps developers develop and deploy semantic sentiment and emotion analysis services in minutes.

5.3. MixedEmotions

The MixedEmotions project [3] plans to continue the work started in the EuroSentiment project, investigating other media (image and sound) in many languages in the sentiment analysis context. Its aim is to develop novel multilingual multi-modal Big Data analytics applications to analyse a more complete emotional profile of user behavior using data from mixed input channels: multilingual text data sources, A/V signal input (multilingual speech, audio, video), social media (social network, comments), and structured data. Commercial applications (implemented as pilot projects) are in Social TV, Brand Reputation Management and Call Centre Operations. Making sense of accumulated user interaction from different data sources, modalities and languages is challenging and yet to be explored in fullness in an industrial context. Commercial solutions exist but do not address the multilingual aspect in a robust and large-scale setting and do not scale up to huge data volumes that need to be processed, or the integration of emotion analysis observations across data sources and/or modalities on a meaningful level. MixedEmotions thus implements an integrated Big Linked Data platform for emotion analysis across heterogeneous data sources, different languages and modalities, building on existing state of the art tools, services and approaches to enable the tracking of emotional aspects of user interaction and feedback on an entity level.

5.4. SEWA

The European Sentiment Analysis in the Wild (SEWA) project [4] deploys and capitalises on existing state-of-the-art methodologies, models and algorithms for machine analysis of facial, vocal and verbal behaviour to realise naturalistic human sentiment analysis "in the wild". The project thus develops computer vision, speech processing, and machine learning tools for automated understanding of human interactive behaviour in naturalistic contexts for audio and visual spatiotemporal continuous and discrete analysis of sentiment, liking and empathy.

5.5. OPENER

OpeNER (Open Polarity Enhanced Name Entity Recognition) is a aims to to provide a set of free Natural Language Processing tools free that are easy to use, adapt and integrate in the workflow of Academia, Research and Small and Medium Enterprise. OpeNER uses the KAF (Bosma et al., 2009) annotation format, with ad-hoc elements to represent sentiment and emotion features. The results of the project include a corpus of annotated reviews and a Linked Data node that exposes this information.

[3] http://mixedemotions-project.eu
[4] http://www.sewaproject.eu/

6. Motivation for a Linked Data Approach

Currently, there are many commercial social media text analysis tools, such as Lexalytics [5], Sentimetrix [6] and Engagor [7] that offer sentiment analysis functionalities from text. There are also a lot of social media monitoring tools that generate statistics about presence, influence power, customer/followers engagement, which are presented in intuitive charts on the user's dashboard. Such tools indicatively are Hootsuite, Klout and Tweetreach which are specialized on Twitter analytics. However, such solutions are quite generic, are not integrated in the process of product development or in product cycles and definitely are not trained under domain-specific terminology, idioms and characteristics. Industry-specific approaches are also available (Aldahawi and Allen, 2013; Abrahams et al., 2012), but still they are not easily configured under integrated, customizable solutions. Opinion mining and trend prediction over social media platforms are emerging research directions with great potential, with companies offering such services tending not to disclose the methodologies and algorithms they use to process data. The academic community has also shown interest into these domains (Pang and Lee, 2008). Some of the most popular domains are User Generated Reviews as well as Twitter mining, particularly due to the availability of information without restriction access (Aiello et al., 2013). An enormous amount of tweets is created daily, Twitter is easily accessible which means that there are available twitter data from people with different background (ethnical, cultural, social), there are tweets in many different languages and finally there is a large variety of discussed topics.

Encoding this extra information is beyond the capabilities of any of the existing formats for sentiment analysis. This is hindering the appearance of applications that make deep sense of data. A Linked Data approach would enable researchers to use this information, as well as other rich information in the Linked Data cloud. Furthermore, it would make it possible to infer new knowledge based on existing reusable vocabularies.

An interesting aspect of analysing social media is that there are many features in the source beyond pure text that can be exploited. Using these features we could gain deeper knowledge and understanding of the user generated content, and ultimately train a system to look for more targeted characteristics. Such a system would be more accurate in processing and categorizing such content. Among the extra features in social media, we find the name of the users who created the content, together with more information about their demographics and other social activities. Moreover users can interact, start conversations over a posted comment, and express their agreement or disagreement either by providing textual responses or explicitly through "thumbs-up" functionalities. Apart from the actual content, it is also the context in which it was created that can serve as a rich source of information and be used to generate more powerful data analytics and lead to smarter company deci-

[5] https://www.lexalytics.com/
[6] http://www.sentimetrix.com/
[7] http://www.engagor.com/

sions. [8]

7. Semantic Models and Vocabularies

This section describes models and vocabularies that can be used to model sentiment and emotion in different scenarios, including annotation of lexical resources (lemon) and NLP services (NIF).

7.1. Marl

Marl is a vocabulary to annotate and describe subjective opinions expressed on the web or in particular Information Systems. This opinions may be provided by the user (as in online rating and review systems), or extracted from natural text (sentiment analysis). Marl models opinions on the aspect and feature level, which is useful for fine grained opinions and analysis.
Marl follows the Linked Data principles as it is aligned with the Provenance Ontology. It also takes a linguistic Linked Data approach: it is aligned with the Provenance Ontology, it represents lexical resources as linked data, and has been integrated with lemon (Section 7.4.).

7.2. Onyx

Onyx (Sánchez-Rada and Iglesias, 2016) is a vocabulary for emotions in resources, services and tools. It has been designed with services and lexical resources for Emotion Analysis in mind. What differentiates Onyx from other vocabularies in Section 4. is that instead of adhering to a specific model of emotions, it provides the concepts to formalize different emotion models. These models are known as vocabularies in Onyx's terminology, following the example of EmotionML. A number of commonly used models have already been integrated and published as linked data [9].
The list includes all EmotionML vocabularies (Ashimura, Kazuyuki et al., 2014), WordNet-Affect labels and the hourglass of emotions (Cambria et al., 2012).
A tool for limited two-way conversion between Onyx representation and EmotionML markup is available, using a specific mapping.
Just like Marl, Onyx is aligned with the Provenance Ontology, and can be used together with lemon in lexical resources.

7.3. NLP Interchange Format (NIF)

NLP Interchange Format (NIF) 2.0 (Hellmann, 2013) defines a semantic format and an API for improving interoperability among natural language processing services.
NIF can be extended via vocabularies modules. It uses Marl for sentiment annotations and Onyx have been proposed as a NIF vocabulary for emotions.

7.4. lemon

lemon is a proposed model for modelling lexicon and machine-readable dictionaries and linked to the Semantic Web and the Linked Data cloud. It was designed to meet the following challenges RDF-native form to enable leverage of existing Semantic Web technologies (SPARQL, OWL, RIF etc.). Linguistically sound structure based on LMF to enable conversion to existing offline formats. Separation of the lexicon and ontology layers, to ensure compatibility with existing OWL models. Linking to data categories, in order to allow for arbitrarily complex linguistic description. In particular, the LexInfo vocabulary is aligned to lemon and ISOcat. A small model using the principle of least power - the less expressive the language, the more reusable the data. Lemon was developed by the Monnet project as a collaboration between: CITEC at Bielefeld University, DERI at the National University of Ireland, Galway, Universidad Politécnica de Madrid and the Deutsche Forschungszentrum für Künstliche Intelligenz.

8. Application

This section contains a noncomprehensive list of popular tools that would potentially benefit from the integration of a unified Linked Data model.

8.1. GATE

GATE (General architecture for Text Engineering) (Cunningham et al., 2009) is an open source framework written entirely in JAVA that can be used for research and commercial applications under the GNU license. It is based on an extensible plugin-architecture and processing resources for several languages are already provided. It can be very useful to manually and automatically annotate text and do subsequential sentiment analysis based on gazetteer lookup and grammar rules as well as machine learning, a support vector machine classifier is already integrated as well as interfaces to linked open data, e.g. DBPedia.

8.2. Speechalyzer

Speechalizer (Burkhardt, 2012) is a java library for the daily work of a 'speech worker', specialized in very fast labeling and annotation of large audio datasets. Includes EmotionML import and export functionality.

8.3. openSMILE

The openSMILE tool enables you to extract large audio feature spaces in realtime for emotion and sentiment analysis from audio and video. It is written in C++ and is available as both a standalone commandline executable as well as a dynamic library (A GUI version is to come soon). The main features of openSMILE are its capability of on-line incremental processing and its modularity. Feature extractor components can be freely interconnected to create new and custom features, all via a simple configuration file. New components can be added to openSMILE via an easy plugin interface and a comprehensive API. openSMILE is free software licensed under the GPL license and is currently available via Subversion in a pre-release state [10].

9. W3C Community Group

The growing interest in the application of Linked Data in the field of Emotion and Sentiment Analysis has motivated

[8] http://www.alchemyapi.com/api/sentiment-analysis

[9] http://www.gsi.dit.upm.es/ontologies/onyx/vocabularies/

[10] http://sourceforge.net/projects/opensmile/

the creation of the W3C Sentiment Analysis Community Group (CG)[11]. The community group is a public forum for experts and practicians from different fields related to Emotion and Sentiment Analysis, as well as semantic technologies. In particular, the community group intends to gather the best practices in the field. Existing vocabularies for emotion and sentiment analysis are thoroughly investigated and taken as a starting point for discussion in the CG. However, its aim is not to publish specifications but rather to identify the needs and pave the way.
It further deals with the requirements beyond text-based analysis, i.e. emotion/sentiment analysis from images, video, social network analysis, etc.

10. Conclusions

Sentiment and Emotion Analysis is a trending field, with a myriad of potential applications and projects exploiting it in the wild. In recent years several European projects have dealt with sentiments and emotions in any of its modalities, such as SEWA and OpeNER. However, as we have explained in this paper, there are several open challenges that need to be addressed. A Linked Data approach would address several of those challenges, as well as foster research in the field and adoption of its technologies. The fact that projects such as ArsEmotica or EuroSentiment have already introduced semantic technologies to deal with similar problems supports this view. Nevertheless, to guarantee the success and adoption of the new approach, we need common vocabularies and best practices for their use. This work is a first step in this direction, which will be continued by the community in the upcoming years with initiatives such as the Linked Data Models for Emotion and Sentiment Analysis W3C Community Group.

11. Acknowledgements

This joint work has been made possible by the existence of the Linked Data Models for Emotion and Sentiment Analysis W3C Community. The work by Carlos A. Iglesias and J. Fernando Sánchez-Rada has been partially funded by the European Union through projects EuroSentiment (FP7 grant agreement #296277) and MixedEmotions (H2020 RIA grant agreement #644632). The work of Björn Schuller has been partially funded by the European Union as part of the SEWA project (H2020 RIA grant agreement #654094)

Abrahams, A. S., Jiao, J., Wang, G. A., and Fan, W. (2012). Vehicle defect discovery from social media. *Decision Support Systems*, 54(1):87–97.

Aiello, L. M., Petkos, G., Martin, C., Corney, D., Papadopoulos, S., Skraba, R., Goker, A., Kompatsiaris, I., and Jaimes, A. (2013). Sensing trending topics in twitter. *Multimedia, IEEE Transactions on*, 15(6):1268–1282.

Aldahawi, H. A. and Allen, S. M. (2013). Twitter mining in the oil business: A sentiment analysis approach. In *Cloud and Green Computing (CGC), 2013 Third International Conference on*, pages 581–586. IEEE.

[11] https://www.w3.org/community/sentiment/

Ashimura, Kazuyuki, Baggia, Paolo, Oltramari, Alessandro, Peter, Christian, and Ashimura, Kazuyuki. (2014). Vocabularies for EmotionML. 00004.

Bertola, F. and Patti, V. (2016). Ontology-based affective models to organize artworks in the social semantic web. *Information Processing & Management, Special issue on Emotion and Sentiment in Social and Expressive Media*, 52(1):139–162.

Bosma, W., Vossen, P., Soroa, A., Rigau, G., Tesconi, M., Marchetti, A., Monachini, M., and Aliprandi, C. (2009). KAF: a generic semantic annotation format. In *Proceedings of the GL2009 workshop on semantic annotation*. 00054.

Burkhardt, F., Schröder, M., Baggia, P., Pelachaud, C., Peter, C., and Zovato, E. (2016). Emotion markup language (emotionml) 1.0.

Burkhardt, F. (2012). Fast labeling and transcription with the speechalyzer toolkit. *Proc. LREC (Language Resources Evaluation Conference), Istanbul*.

Cambria, E., Livingstone, A., and Hussain, A. (2012). The hourglass of emotions. In *Cognitive behavioural systems*, pages 144–157. Springer. 00072 bibtex: Cambria2012.

Chan, C.-C. and Liszka, K. J. (2013). Application of Rough Set Theory to Sentiment Analysis of Microblog Data. In *Rough Sets and Intelligent Systems-Professor Zdzis\law Pawlak in Memoriam*, pages 185–202. Springer. 00000.

Clavel, C. and Callejas, Z. (2015). Sentiment analysis: from opinion mining to human-agent interaction. *Affective Computing, IEEE Transactions on*, PP(99):1–1, to appear.

Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V., Ursu, C., Dimitrov, M., Dowman, M., Aswani, N., Roberts, I., Li, Y., and others. (2009). *Developing Language Processing Components with GATE Version 5:(a User Guide)*. University of Sheffield. 00111.

Hellmann, S. (2013). *Integrating Natural Language Processing (NLP) and Language Resources using Linked Data*. Ph.D. thesis, Universität Leipzig. 00002.

Ishizuka, M. (2012). Textual affect sensing and affective communication. In *Cognitive Informatics & Cognitive Computing (ICCI* CC), 2012 IEEE 11th International Conference on*, pages 2–3. IEEE. 00004.

Kim, S.-M. and Hovy, E. (2004). Determining the sentiment of opinions. In *Proceedings of the 20th international conference on Computational Linguistics*, page 1367. Association for Computational Linguistics. 01159.

Krcadinac, U., Pasquier, P., Jovanovic, J., and Devedzic, V. (2013). Synesketch: An open source library for sentence-based emotion recognition. *Affective Computing, IEEE Transactions on*, 4(3):312–325. 00011.

Liu, B. (2012). *Sentiment Analysis and Opinion Mining*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers.

Martin, J. R. and White, P. R. (2005). *The Language of Evaluation. Appraisal in English*. Palgrave Macmillan Basingstoke and New York.

Munezero, M., Montero, C. S., Sutinen, E., and Pajunen, J. (2014). Are they different? affect, feeling, emotion, sentiment, and opinion detection in text. *Affective Computing, IEEE Transactions on*, 5(2):101–111. 00012.

Nasoz, F. and Lisetti, C. L. (2007). Affective user modeling for adaptive intelligent user interfaces. In *Human-Computer Interaction. HCI Intelligent Multimodal Interaction Environments, 12th International Conference, HCI International 2007, Beijing, China, July 22-27, 2007, Proceedings, Part III*, pages 421–430.

Ortigosa, A., Martín, J. M., and Carro, R. M. (2014a). Sentiment analysis in Facebook and its application to e-learning. *Computers in Human Behavior*, 31:527–541. 00051.

Ortigosa, A., Martín, J. M., and Carro, R. M. (2014b). Sentiment analysis in facebook and its application to e-learning. *Computers in Human Behavior*, 31:527 – 541.

Pang, B. and Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and trends in information retrieval*, 2(1-2):1–135.

Patti, V., Bertola, F., and Lieto, A. (2015). Arsemotica for arsmeteo.org: Emotion-driven exploration of online art collections. In *Proceedings of the Twenty-Eighth International Florida Artificial Intelligence Research Society Conference, FLAIRS 2015, Hollywood, Florida. May 18-20, 2015*, pages 288–293.

Plutchik, R. E. and Conte, H. R. (1997). *Circumplex models of personality and emotions*. American Psychological Association. 00258.

Scherer, K. R. (2005). What are emotions? And how can they be measured? *Social science information*, 44(4):695–729. 01546.

Strapparava, C., Valitutti, A., and others. (2004). WordNet Affect: an Affective Extension of WordNet. In *LREC*, volume 4, pages 1083–1086. 00859.

Sykora, M. D., Jackson, T. W., O'Brien, A., and Elayan, S. (2013). Emotive ontology: Extracting fine-grained emotions from terse, informal messages. *Computer Science and Information Systems Journal*. 00011.

Sánchez-Rada, J. F. and Iglesias, C. A. (2016). Onyx: A Linked Data approach to emotion representation. *Information Processing & Management*, 52(1):99–114, January. 00003.

Yan, J., Bracewell, D. B., Ren, F., and Kuroiwa, S. (2008). The Creation of a Chinese Emotion Ontology Based on HowNet. *Engineering Letters*, 16(1):166–171. 00022.

Zimbra, D. and Chen, H. (2012). Scalable sentiment classification across multiple dark web forums. In *2012 IEEE International Conference on Intelligence and Security Informatics, ISI 2012, Washington, DC, USA, June 11-14, 2012*, pages 78–83.

3.1.4 A Linked Data Model for Multimodal Sentiment and Emotion Analysis

Title	A Linked Data Model for Multimodal Sentiment and Emotion Analysis
Authors	Sánchez-Rada, J. Fernando and Iglesias, Carlos A. and Gil, Ronald
Proceedings	4th Workshop on Linked Data in Linguistics: Resources and Applications
DOI	10.18653/v1/W15-4202
Year	2015
Keywords	iglesias, sanchezrada
Pages	11–19
Online	http://aclweb.org/anthology/W/W15/W15-4202.pdf
Abstract	The number of tools and services for sentiment analysis is increasing rapidly. Unfortunately, the lack of standard formats hinders interoperability. To tackle this problem, previous works propose the use of the NLP Interchange Format (NIF) as both a common semantic format and an API for textual sentiment analysis. However, that approach creates a gap between textual and sentiment analysis that hampers multimodality. This paper presents a multimedia extension of NIF that can be leveraged for multimodal applications. The application of this extended model is illustrated with a service that annotates online videos with their sentiment and the use of SPARQL to retrieve results for different modes.

A Linked Data Model for Multimodal Sentiment and Emotion Analysis

J. Fernando Sánchez-Rada
Grupo de Sistemas Inteligentes
Universidad Politécnica de Madrid
jfernando@dit.upm.es

Carlos A. Iglesias
Grupo de Sistemas Inteligentes
Universidad Politécnica de Madrid
cif@dit.upm.es

Ronald Gil
Massachusetts Institute of Technology
rongil@mit.edu

Abstract

The number of tools and services for sentiment analysis is increasing rapidly. Unfortunately, the lack of standard formats hinders interoperability. To tackle this problem, previous works propose the use of the NLP Interchange Format (NIF) as both a common semantic format and an API for textual sentiment analysis. However, that approach creates a gap between textual and sentiment analysis that hampers multimodality. This paper presents a multimedia extension of NIF that can be leveraged for multimodal applications. The application of this extended model is illustrated with a service that annotates online videos with their sentiment and the use of SPARQL to retrieve results for different modes.

1 Introduction

With the rise of social media and crowdsourcing, the interest in automatic means of extraction and aggregation of user opinions (Opinion Mining) and emotions (Emotion Mining) is growing. This tendency is mainly focused on text analysis, the cause and consequence of this being that the tools for text analysis are getting better and more accurate. As is often the case, these tools are heterogeneous and implement different formats and APIs. This problem is hardly new or limited to sentiment analysis, it is also present in the Natural Language Processing (NLP) field. In fact, both fields are closely related: textual sentiment analysis can be considered a branch of NLP. Looking at how NLP deals with heterogeneity and interoperability we find NIF, a format for NLP services that solves these issues. Unfortunately, *NLP Interchange Format* (NIF) (Hellmann et al., 2013) is not enough to annotate sentiment analysis services. Fortunately, it can be extended, by exploiting the extensibility of semantic formats. Using this extensibility and already existing ontologies for the sentiment and emotion domains, the R&D Eurosentiment project recently released a model that extends NIF for sentiment analysis (Buitelaar et al., 2013).

However, the Eurosentiment model is bound to textual sentiment analysis, as NIF focuses on annotation of text. The R&D MixedEmotions project aims at bridging this gap by providing a Big Linked Data Platform for multimedia and multilingual sentiment and emotion analysis. Hence, different modes (e.g. images, video, audio) require different formats. Format heterogeneity becomes problematic when different modes coexist or when the text is part of other media. Some examples of this include working with text extracted from a picture with OCR, or subtitles and transcripts of audio and video. This scenario is not uncommon, given the maturity of textual sentiment analysis tools.

In particular, this paper focuses on video and audio sources that contain emotions and opinions, such as public speeches. We aim to represent that information in a linked data format, linking the original source with its transcription and any sentiments or emotions found in any of its modes. Using the new model it is possible to represent and process multimodal sentiment information using a common set of tools.

The rest of the paper is structured as follows: Section 2 covers the background for this work; Section 3 presents requirements for semantic annotation of sentiment in multimedia; Section 4

Proceedings of the 4th Workshop on Linked Data in Linguistics (LDL-2015), pages 11–19,
Beijing, China, July 31, 2015. ©2015 Association for Computational Linguistics and Asian Federation of Natural Language Processing

introduces the bases for sentiment analysis using NIF and delves into the use of NIF for media other than text; Section 5 exemplifies the actual application of the new model with a prototype and semantic queries; Section 6 is dedicated to related work; lastly, Section 7 summarises the conclusions drawn from our work and presents possible lines of work.

2 Background

2.1 Annotation based on linked data

Annotating is the process of associating metadata with multimedia assets. Previous research has shown that annotations can benefit from compatibility with linked data technologies (Hausenblas, 2007).

The W3C Open Annotation Community Group has worked towards a common RDF-based specification for annotating digital resources. The group intends to reconcile two previous proposals: the Annotation Ontology (Ciccarese et al., 2011) and the Open Annotation Collaboration (OAC) (Haslhofer et al., 2011). Both proposals incorporate elements from the earlier Annotea model (Kahan et al., 2002). The Open Annotation Ontology (Robert Sanderson and de Sompel, 2013) provides a general description mechanism for sharing annotation between systems based on an RDF model. An annotation is defined by two relationships: *body*, the annotation itself, and *target*, the asset that is annotated. Both body and target can be of any media type. In addition, parts of the body or target can be identified by using Fragment Selectors (oa:FragmentSelector) entities. W3C Fragment URIs (Tennison, 2012) can be used instead, although the use of Fragment Selectors is encouraged. The vocabulary defines fragment selectors for text (oa:Text), text segments plus passages before or after them (oa:TextQuoteSelector), byte streams (oa:DataPositionSelector), areas (oa:AreaSelector), states (oa:State), time moments (oa:TimeState) and request headers (oa:RequestHeaderState). Finally, Open Annotation (OA) ontology defines how annotations are published and transferred between systems. The recommended serialisation format is JSON-LD.

Another research topic has been the standardisation of linguistic annotations in order to improve the interoperability of NLP tools and resources. The main proposals are Linguistic Annotation Framework (LAF) and NIF 2.0. The ISO Specification LAF (Ide and Romary, 2004) and its extension Graph Annotation Format (GrAF) (Ide and Suderman, 2007) define XML serialisation of linguistic annotation as well as RDF mappings. NIF 2.0 (Hellmann et al., 2013) follows a pragmatic approach to linguistic annotations and is focused on interoperability of NLP tools and services. It is directly based on RDF, Linked Data and ontologies, and it allows handling structural interoperability of linguistic annotations as well as semantic interoperability. NIF 2.0 Core ontology provides classes and properties to describe the relationships between substrings, text and documents by assigning URIs to strings. These URIs can then be used as subjects in RDF easily annotated. NIF builds on current best practices for counting strings and creating offsets such as LAF. NIF uses Ontologies for Linguistic Annotation (OLiA) (Chiarcos, 2012) to provide stable identifiers for morpho-syntactical annotation tag sets. In addition to the core ontology, NIF defines Vocabulary modules as an extension mechanism to achieve interoperability between different annotation layers. Some of the defined vocabularies are Marl (Westerski et al., 2011) and Lexicon Model for Ontologies (lemon) (Buitelaar et al., 2011).

As discussed by Hellmann (Hellmann, 2013), the selection of the annotation scheme comes from the domain annotation requirements and the trade-off among granularity, expressiveness and simplicity. He defines different profiles with this purpose. The profile NIF simple can express *the best estimate* of an NLP tool in a flat data model, with a low number of triples. An intermediate profile called NIF Stanbol allows the inclusion of alternative annotations with different confidence as well as provenance information that can be attached to the additionally created URN for each annotation. This profile is integrated with the semantic content management system Stanbol (Westenhaler, 2014). Finally, the profile NIF OA provides the most expressive model but requires more triples and creates up to four new URNs per annotation, making it more difficult to query.

Finally, we review Fusepool since they propose an annotation model that combines OA and NIF. Fusepool (Westenhaler, 2014) is an R&D project whose purpose is to digest and turn data from different sources into linked data to make data interoperable for reuse. One of the tasks of this

project is to define a new Enhancement Structure for the semantic content management system Apache Stanbol (Bachmann-Gmür, 2013). Fusepool researchers' main design considerations with OA is for it to define a very expressive model capable of very complex annotations. This technique comes with the disadvantage of needing a high amount of triples to represent lower level NLP processing, which in turn complicates the queries necessary to retrieve simple data.

2.2 Eurosentiment Model

The work presented here is partly based on an earlier work (Buitelaar et al., 2013) developed within the Eurosentiment project. The Eurosentiment model proposes a linked data approach for sentiment and emotion analysis, and it is based on the following specifications:

- Marl (Westerski et al., 2011) is a vocabulary designed to annotate and describe subjective opinions expressed on the web or in information systems
- Onyx (Sanchez-Rada and Iglesias, 2013) is built on the same principles as Marl to annotate and describe emotions, and provides interoperability with Emotion Markup Language (EmotionML) (Schröder et al., 2011)
- lemon (Buitelaar et al., 2011) defines a lexicon model based on linked data principles which has been extended with Marl and Onyx for sentiment and emotion annotation of lexical entries
- NIF 2.0 (Hellmann et al., 2013) which defines a semantic format and API for improving interoperability among natural language processing services

The way these vocabularies have been integrated is illustrated in the example below, where we are going to analyse the sentiment of an opinion ("Like many Paris hotels, the rooms are too small") posted in TripAdvisor. In the Eurosentiment model, *lemon* is used to define the lexicon for a domain and a language. In our example, we have to generate this lexicon for the hotel domain and the English language[1]. A reduced lexicon for Hotels in English (le:hotel_en) is shown in Listing 1 for illustration purposes. The lexicon is composed of a set of lexical entries (prefix lee). Each lexical entry is semantically disambiguated and provides a reference to the syntactic variant (in the example the canonical form) and the senses. The example shows how the senses have been extended to include sentiment features. In particular, the sense small_1 in the context of room_1 has associated a negative sentiment. That is, "small room" is negative (while small phone could be positive, for example).

```
lee:sense/small_1 a lemon:Sense;
 lemon:reference "01391351";
 lexinfo:partOfSpeech lexinfo:adjective;
 lemon:context lee:sense/room_1;
 marl:polarityValue "-0.5"^^xsd:double;
 marl:hasPolarity marl:negative.

le:hotel_en a lemon:Lexicon;
 lemon:language "en";
 lemon:topic ed:hotel;
 lemon:entry lee:room, lee:Paris, lee:
     small.

lee:room a lemon:LexicalEntry;
 lemon:canonicalForm [ lemon:writtenRep
     "room"@en ];
 lemon:sense [ lemon:reference wn:synset
     -room-noun-1;
      lemon:reference dbp:Room ];
 lexinfo:partOfSpeech lexinfo:noun.

lee:Paris a lemon:LexicalEntry;
  lemon:canonicalForm [ lemon:writtenRep
       "Paris"@en ];
  lemon:sense [ lemon:reference dbp:
      Paris;
      lemon:reference wn:synset-room-noun
          -1 ];
 lexinfo:partOfSpeech lexinfo:noun.

lee:small a lemon:LexicalEntry;
 lemon:canonicalForm [ lemon:writtenRep
     "small"@en ];
 lemon:sense lee:sense/small_1;
 lexinfo:partOfSpeech lexinfo:adjective.
```

Listing 1: Sentiment analysis expressed with Eurosentiment model.

The Eurosentiment model uses NIF in combination with Marl and Onyx to provide a standardised service interface. In our example, let us assume the opinion has been published at http://tripadvisor.com/myhotel. NIF follows a linked data principled approach so that different tools or services can annotate a text. To this end, texts are converted to RDF literals and an URI is generated so that annotations can be defined for that text in a linked data way. NIF offers different URI Schemes to identify text fragments inside a

[1]The reader interested in how this domain specific lexicon can be generated can consult (Vulcu et al., 2014).

string, e.g. a scheme based on RFC5147 (Wilde and Duerst, 2008), and a custom scheme based on context. In addition to the format itself, NIF 2.0 defines a REST API for NLP services with standardised parameters. An example of how these ontologies are integrated is illustrated in Listings 2, 3 and 4.

```
<http://tripadvisor.com/myhotel#char
    =0,49>
  rdf:type nif:RDF5147String , nif:
      Context;
  nif:beginIndex "0";
  nif:endIndex "49";
  nif:sourceURL <http://tripadvisor.com/
      myhotel.txt>;
  nif:isString "Like many Paris hotels,
      the rooms are too small";
  marl:hasOpinion <http://tripadvisor.
      com/myhotel/opinion/1>.
```

Listing 2: NIF + Marl output of a service call http://eurosentiment.eu?i=Like many Paris hotels, the rooms are too small

```
<http://tripadvisor.com/myhotel/opinion
    /1>
  rdf:type marl:Opinion;
  marl:describesObject dbp:Hotel;
  marl:describesObjectPart dbp:Room;
  marl:describesFeature  "size";
  marl:polarityValue "-0.5";
  marl:hasPolarity: http://purl.org/marl
      /ns#Negative.
```

Listing 3: Sentiment analysis expressed with Eurosentiment model.

```
<http://eurosentiment.eu/analysis/1>
  rdf:type marl:SentimentAnalysis;
  marl:maxPolarityValue "1";
  marl:minPolarityValue "-1";
  marl:algorithm "dictionary-based";
  prov:used le:hotel_en;
  prov:wasAssociatedWith http://dbpedia.
      org/resource/UPM.
```

Listing 4: Sentiment analysis expressed with Eurosentiment model.

3 Requirements for semantic annotation of sentiment in multimedia resources

The increasing need to deal with human factors, including emotions, on the web has led to the development of the W3C specification EmotionML (Schröder et al., 2011). EmotionML aims for a trade-off between practical applicability and scientific well-foundedness. Given the lack of agreement on a finite set of emotion descriptors, EmotionML follows a plug-in model where emotion vocabularies can be defined depending on the application domain and the aspect of emotions to be focused.

EmotionML (Schröder et al., 2011) uses Media URIs to annotate multimedia assets. Temporal clipping can be specified either as Normal Play Time (npt) (Schulzrinne et al., 1998), as SMPTE timecodes (Society of Motion Picture and Television Engineers, 2009), or as real-world clock time (Schulzrinne et al., 1998).

During the definition of the EmotionML specification, the Emotion Incubator group defined 39 individual use cases (Schröder et al., 2007) that could be grouped into three broad types: manual annotation of materials (e.g. annotation of videos, speech recordings, faces or texts), automatic recognition of emotions from sensors and generation of emotion-related system responses. Based on these uses cases as well as others identified in the literature (Grassi et al., 2011), a number of requirements have been identified for the annotation of multimedia assets based on linked data technologies:

- **Standards compliance**. Emotion annotations should be based on linked data technologies such as RDF or W3C Media Fragment URI. Unfortunately, EmotionML has been defined in XML. Nevertheless, as commented above, the vocabulary Onyx provides a linked data version of EmotionML that can be used instead. Regarding the annotation framework, OA covers the annotation of multimedia assets while NIF only supports the annotation of textual sources.
- **Trace annotation of time-varying signals**. The time curve of properties scales (e.g. arousal or valence) should be preserved. To this end, EmotionML defines two mechanisms. The element *trace* allows the representation of the time evolution of a dynamic scale value based on a periodic sampling of values (i.e. one value every 100ms at 10 Hz). In case of aperiodic sampling, separate emotion annotations should be used. The current version of the ontologies we use does not support trace annotations.
- **Annotations of multimedia fragments**. Fragments of multimedia assets should be enabled. To this end, EmotionML uses Media

URIs to be able to annotate temporal interval or frames. As presented above, NIF provides a compact scheme for textual fragment annotation, but it does not cover multimedia fragments. In contrast, OA supports the annotation of multimedia fragments using a number of triples.

- **Collaborative and multi-modal annotations**. Emotion analysis of multimedia assets may be performed based on different combination of modalities (i.e. full body video, facial video, each with or without speech or textual transcription). Thus, interoperability of emotion annotations is essential. Semantic web technologies provide a solid base for distributed, interoperable and shareable annotations, with proposals such as OA and NIF.

4 Linked Data Annotation for Multimedia Sentiment Analysis

One of the main goals of NIF is interoperability between NLP tools. For this, it uses a convention to assign URIs to parts of a text. Since URIs are unique, different tools can analyse the same text independently, and one may use the URIs later to combine the information from both.

These URIs are constructed with a combination of the URI of the source of the string (its context), and a unique identifier for that string within that particular context. A way to assign that identifier is called a URI scheme. Strings belong to different classes, according to the scheme used to generate its URI. The currently available schemes are: ContextHashBasedString, OffsetBasedString, RFC5147String and ArbitraryString. The usual scheme is *RFC5147String*.

For instance, for a context `http://example.com`, its content may be "This is a test", and the *RFC5147String* `http://example.com#char=5,7` would refer to the "is" part within the context.

However, to annotate multimedia sources indexing by characters is obviously not possible. We need a different way to uniquely refer to a fragment.

Among the different possible approaches to identify media elements, we propose to follow the same path as the Ontology for Media Resources (Lee et al., 2012) and use the Media Fragments URI W3C recommendation (Troncy et al., 2012). The recommendation specifies how to refer to a specific fragment or subpart of a media resource. URIs follow this scheme:

`<scheme>:<part>[?<q>][#<frag.>]`

Where `<scheme>` is the specific scheme or protocol (e.g. http), `part` is the hierarchical part (e.g. example.com), `q` is the query (e.g. user=Nobody), and `frag` is the piece we are interested in: the multimedia fragment (e.g. t=10).

Since the Media Fragments URI schema is very similar to those already used in NIF and follows the same philosophy, we have extended NIF to include it. The result is Figure 1.

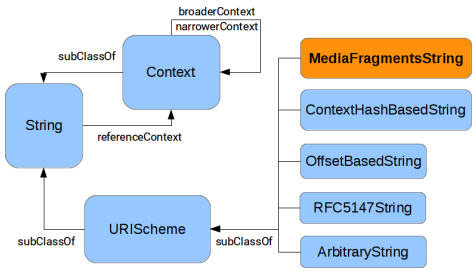


Figure 1: By extending the URI Schemes of NIF, we make it possible to use multimedia sources in NIF, and refer to their origin using the Media Fragments recommendation.

Using this URI Scheme and the NIF notation for sentiment analysis, the results from a service that analyses both the captions from a YouTube video and the video comments would look like the document in Listing 5. In this way, we fulfill the requirements previously identified in Sect. 3. This example is, in fact, the aggregation of three different kinds of analysis: textual sentiment analysis on comments (`CommentAnalysis`) and captions (`CaptionAnalysis`), and sentiment analysis based on facial expressions (`SmileAnalysis`). Each analysis would individually return a document similar to that of the example, with only the fields corresponding to that particular analysis.

The results can be summarised as follows: a youtube video (`http://youtu.be/W07PoKUD-Yk`) is tagged as positive overall based on facial expressions (`OpinionS01`); the section of the video from second 108 to second 110 (`http://youtu.be/W07PoKUD-Yk#t=108,110`) reflects negative sentiment judging by the captions (`OpinionT01`);

lastly, the video has a comment (`http://www.youtube.com/comment?lc=<CommentID>`) that reflects a positive opinion (`OpinionC01`).

The JSON-LD context in Listing 6 provides extra information the semantics of the document, and has been added for completeness.

```
{
"analysis": [{
 "@id": "SmileAnalysis",
 "@type": "marl:SentimentAnalysis",
 "marl:algorithm": "AverageSmiles"
 }, {
 "@id": "CaptionAnalysis",
 "@type": "marl:SentimentAnalysis",
 "marl:algorithm": "NaiveBayes"
 }, {
 "@id": "CommentAnalysis",
 "@type": "marl:SentimentAnalysis",
 "marl:algorithm": "NaiveBayes"
 }],
"entries": [{
 "@id": "http://youtu.be/W07PoKUD-Yk",
 "@type": [
  "nifmedia:MediaFragmentsString",
  "nif:Context"],
 "nif:isString": "<FULL Transcript>",
 "opinions": [{
  "@id": "_:OpinionS01",
  "marl:hasPolarity": "marl:Positive",
  "marl:polarityValue": 0.5,
  "prov:generatedBy": "SmileAnalysis"
  }],
 "sioc:hasReply": "http://
     ↪ www.youtube.com/comment?lc=<
     ↪ CommentID>",
 "strings": [{
  "@id": "http://youtu.be/W07PoKUD-Yk#t=
      ↪ 108,110",
  "@type": "nifmedia:
      ↪ MediaFragmentsString",
  "nif:anchorOf": "Family budgets under
      ↪ pressure",
  "opinions": [{
   "@id": "_:OpinionT01",
   "marl:hasPolarity": "marl:Negative",
   "marl:polarityValue": -0.3058,
   "prov:generatedBy": "CaptionAnalysis"
   }]
 }]
 }, {
 "@id": "http://www.youtube.com/comment?
     ↪ lc=<CommentID>",
 "@type": [
  "nif:Context", "nif:RFC5147String" ],
 "nif:isString": "He is well spoken",
 "opinions": [{
  "@id": "OpinionC01",
  "marl:hasPolarity": "marl:Positive",
  "marl:polarityValue": 1,
  "prov:generatedBy": "CommentAnalysis"
  }]
 }]
}
```

Listing 5: Service results are annotated on the fragment level with sentiment and any other property in NIF such as POS tags or entities.

```
{
 "marl": "http://www.gsi.dit.upm.es/
     ↪ ontologies/marl/ns#",
 "nif": "http://persistence.uni-
     ↪ leipzig.org/nlp2rdf/ontologies/
     ↪ nif-core#",
 "onyx": "http://www.gsi.dit.upm.es/
     ↪ ontologies/onyx/ns#",
 "nifmedia": "http://www.gsi.dit.upm.es/
     ↪ ontologies/nif/ns#",
 "analysis": {
    "@id": "prov:wasInformedBy"
 },
 "opinions": {
   "@container": "@list",
   "@id": "marl:hasOpinion",
   "@type": "marl:Opinion"
 },
 "entries": {
    "@id": "prov:generated"
 },
 "strings": {
    "@reverse": "nif:hasContext"
 }
}
```

Listing 6: JSON-LD context for the results necessary to give semantic meaning to the JSON in Listing 5.

5 Application

5.1 VESA: Online HTML5 Video Annotator

The first application to use NIF annotation for sentiment analysis of Multimedia sources is VESA, the Video Emotion and Sentiment Analysis tool. VESA is both a tool to run sentiment analysis of online videos, and a visualisation tool which shows the evolution of sentiment information and the transcript as the video is playing, using HTML5 widgets. The visualisation tool can run the analysis in real time (live analysis) or use previously stored results.

The live analysis generates transcriptions using the built-in Web Speech API in Google Chrome[2] while the video plays in the background. To improve the performance and accuracy of the transcription process, the audio is chunked in sentences (delimited by a silence). Then, each chunk is sent to a sentiment analysis service. As of this writing, users can choose sentiment analysis in Spanish or English, in a general or a financial domain, using different dictionaries.

The evolution of sentiment within the video is shown as a graph below the video in Figure 2. The

[2] `https://www.google.com/intl/en/chrome/demos/speech.html`

full transcript of the video allows users to check the accuracy of the transcription service.

The results from the service can be stored in a database, and can be later replayed. We also developed a Widget version of the annotator that can be embedded in other websites, and integrated in widget frameworks like Sefarad[3].

The project is completely open source and can be downloaded from its Github repository[4].

Figure 2: The graph shows the detected sentiment in the video over time, while the video keeps playing.

5.2 Semantic multimodal queries

This section demonstrates how it would be possible to integrate sentiment analysis of different modes using SPARQL. In particular, it covers two scenarios: fusion of results from different modes, and detection of complex patterns using information from several modes.

As discussed in Section 6, SPARQL has some limitations when it comes to querying media fragments. There are extensions to SPARQL that overcome those limitations. However, for the sake of clarity, this section will avoid those extensions. Instead, the examples assume that the original media is chunked equally for every mode. Every chunk represents a media fragment, which may contain an opinion.

[3] http://github.com/gsi-upm/Sefarad

[4] https://github.com/gsi-upm/video-sentiment-analysis

When different modes yield different sentiments or emotions, it is usually desirable to integrate all the results into a single one. The query in Listing 7 shows how to retrieve all the opinions for each chunk. These results can be fed to a fusion algorithm.

```
SELECT ?frag ?algo ?opinion ?pol WHERE {
  ?frag a nifmedia:MediaFragmentsString;
        marl:hasOpinion ?opinion.
  ?opinion marl:hasPolarity ?pol.
  ?algo prov:generated ?opinion.
}
```

Listing 7: Gathering all the opinions detected in a video.

Another possibility is that the discrepancies between different modes reveal useful information. For instance, using a cheerful tone of voice for a negative text may indicate sarcasm or untruthfulness. Listing 8 shows an example of how to detect such discrepancies. Note that it uses both opinions and emotions at the same time.

```
SELECT ?frag WHERE {
  ?frag a nifmedia:MediaFragmentsString;
        marl:hasOpinion ?opinion;
        onyx:hasEmotion ?emo.
  ?opinion prov:wasGeneratedBy
               _:TextAnalysis;
           marl:hasPolarity marl:
               Negative.
  ?emo prov:wasGeneratedBy
               _:AudioAnalysis;
       onyx:hasEmotionCategory wna:
           Cheerfulness.
}
```

Listing 8: Detecting negative text narrated with a cheerful tone of voice.

6 Related work

Semedi research group proposes the use of semantic web technologies for video fragment annotation (Morbidoni et al., 2011) and affective states based on the HEO (Grassi et al., 2011) ontology. They propose the use of standards, such as XPointer (Paul Grosso and Walsh, 2003) and Media Fragment URI (Troncy et al., 2012) for defining URIs for text and multimedia, respectively, as well as the Open Annotation Ontology (Robert Sanderson and de Sompel, 2013) for expressing the annotations. Their approach is similar to the one we have proposed, based on web standards and linked data to express emotion annotations. Our proposal has been aligned with the latest available specifications, which have been extended as presented in this article.

On the other hand, a better integration between multimedia and the linked data toolbox would be necessary. Working with multimedia fragments in plain SPARQL is not an easy task. More specifically, it is the relationship between fragments that complicates it, e.g. finding overlaps or contiguous segments. An extension to SPARQL by Kurz et al. (Kurz et al., 2014), SPARQL-MM, introduces convenient methods that allow these operations in a concise way.

7 Conclusions and future work

We have introduced the conceptual tools to describe sentiment and emotion analysis results in a semantic format, not only from textual sources but also multimedia.

Despite being primarily oriented towards analysis of texts extracted from multimedia sources, this approach can be used to apply other kinds of analysis, in a way similar to how NIF integrates results from different tools. However, more effort needs to be put into exploring different use cases and how they can be integrated in our extension of NIF for sentiment analysis in multimedia. This work will be done in the project MixedEmotions, where several use cases (Brand Monitoring, Social TV or Call Center Management) have been identified and involve multimedia analysis.

In addition, this discussion can be carried out in the *Linked Data Models for Emotion and Sentiment Analysis* W3C Community Group [5], where professionals and academics of the Semantic and sentiment analysis worlds meet and discuss the application of an interdisciplinary approach.

Regarding the video annotator, although the current version is fully functional, it could be improved in several ways. The main limitation is that its live analysis relies on the Web Speech API, and needs user interaction to set specific audio settings. We are studying other fully client-side approaches.

Acknowledgements

This research has been partially funded and by the EC through the H2020 project MixedEmotions (Grant Agreement no: 141111) and by the Spanish Ministry of Industry, Tourism and Trade through the project Calista (TEC2012-32457).

The authors also want to thank Berto Yáñez for his support and inspiring demo "Popcorn.js Sentiment Tracker".

[5] http://www.w3.org/community/sentiment/

References

Reto Bachmann-Gmür. 2013. *Instant Apache Stanbol*. Packt Publisher.

Paul Buitelaar, Philipp Cimiano, John McCrae, Elena Montiel-Ponsoda, and Thierry Declerck. 2011. Ontology lexicalisation: The lemon perspective. In *Workshop at 9th International Conference on Terminology and Artificial Intelligence (TIA 2011)*, pages 33–36.

Paul Buitelaar, Mihael Arcan, Carlos A Iglesias, J Fernando Sánchez-Rada, and Carlo Strapparava. 2013. Linguistic linked data for sentiment analysis. In *2nd Workshop on Linked Data in Linguistics (LDL-2013): Representing and linking lexicons, terminologies and other language data*, Pisa, Italy.

Christian Chiarcos. 2012. Ontologies of linguistic annotation: Survey and perspectives. In *LREC*, pages 303–310.

Paolo Ciccarese, Marco Ocana, Leyla Jael Garcia-Castro, Sudeshna Das, and Tim Clark. 2011. An open annotation ontology for science on web 3.0. *J. Biomedical Semantics*, 2(S-2):S4.

Marco Grassi, Christian Morbidoni, and Francesco Piazza. 2011. Towards semantic multimodal video annotation. In *Toward Autonomous, Adaptive, and Context-Aware Multimodal Interfaces. Theoretical and Practical Issues*, volume 6456 of *Lecture Notes in Computer Science*, pages 305–316. Springer Berlin Heidelberg.

Bernhard Haslhofer, Rainer Simon, Robert Sanderson, and Herbert Van de Sompel. 2011. The open annotation collaboration (oac) model. In *Multimedia on the Web (MMWeb), 2011 Workshop on*, pages 5–9. IEEE.

Michael Hausenblas. 2007. Multimedia vocabularies on the semantic web. Technical report, World Wide Web (W3C).

Sebastian Hellmann, Jens Lehmann, Sören Auer, and Martin Brümmer. 2013. Integrating nlp using linked data. In *The Semantic Web–ISWC 2013*, pages 98–113. Springer.

Sebastian Hellmann. 2013. *Integrating Natural Language Processing (NLP) and Language Resources using Linked Data*. Ph.D. thesis, Universität Leipzig.

Nancy Ide and Laurent Romary. 2004. International standard for a linguistic annotation framework. *Natural language engineering*, 10(3-4):211–225.

Nancy Ide and Keith Suderman. 2007. Graf: A graph-based format for linguistic annotations. In *Proceedings of the Linguistic Annotation Workshop*, LAW '07, pages 1–8, Stroudsburg, PA, USA. Association for Computational Linguistics.

J. Kahan, M.-R. Koivunen, E. Prud'Hommeaux, and R.R. Swick. 2002. Annotea: an open {RDF} infrastructure for shared web annotations. *Computer Networks*, 39(5):589 – 608.

Thomas Kurz, Sebastian Schaffert, Kai Schlegel, Florian Stegmaier, and Harald Kosch. 2014. Sparql-mm-extending sparql to media fragments. In *The Semantic Web: ESWC 2014 Satellite Events*, pages 236–240. Springer.

WonSuk Lee, Werner Bailer, Tobias Bürger, Pierre-Antoine Champin, Jean-Pierre Evain, Véronique Malaisé, Thierry Michel, Felix Sasaki, Joakim Söderberg, Florian Stegmaier, and John Strassner. 2012. Ontology for Media Resources 1.0. Technical report, World Wide Web Consortium (W3C), February. Available at http://www.w3.org/TR/mediaont-10/.

C. Morbidoni, M. Grassi, M. Nucci, S. Fonda, and G. Ledda. 2011. Introducing semlib project: semantic web tools for digital libraries. *International Workshop on Semantic Digital Archives-Sustainable Long-term Curation Perspectives of Cultural Heritage Held as Part of the 15th International Conference on Theory and Practice of Digital Libraries (TPDL), Berlin*.

Jonathan Marsh Paul Grosso, Eve Mater and Normal Walsh. 2003. W3C XPointer Framework. Technical report, World Wide Web Consortium (W3C), March. Available at http://www.w3.org/TR/xptr-framework/.

Pablo Ciccarese Robert Sanderson and Herbert Van de Sompel. 2013. W3C Open Annotation Data Model. Technical report, World Wide Web Consortium (W3C), February. Available at http://www.openannotation.org/spec/core/.

J Fernando Sanchez-Rada and Carlos Angel Iglesias. 2013. Onyx: Describing emotions on the web of data. In *ESSEM@ AI* IA*, pages 71–82. Citeseer.

Marc Schröder, Enrico Zovato, Hannes Pirker, Christian Peter, and Felix Burkhardt. 2007. W3C Emotion Incubator Group Report 10 July 2007. Technical report, W3C, July. Available at http://www.w3.org/2005/Incubator/emotion/XGR-emotion/.

Marc Schröder, Paolo Baggia, Felix Burkhardt, Catherine Pelachaud, Christian Peter, and Enrico Zovato. 2011. Emotionml – an upcoming standard for representing emotions and related states. In Sidney D'Mello, Arthur Graesser, Björn Schuller, and Jean-Claude Martin, editors, *Affective Computing and Intelligent Interaction*, volume 6974 of *Lecture Notes in Computer Science*, pages 316–325. Springer Berlin Heidelberg.

H. Schulzrinne, A. Rao, and R. Lanphier. 1998. Real Time Streaming Protocol (RTP). Technical Report RFC2326, IETF. Available at http://www.ietf.org/rfc/rfc2326.txt.

Society of Motion Picture and Television Engineers. 2009. SMPTE - RP 136. time and control codes for 24, 25 or 30 frame-per-second motion-picture systems - stabilized 2009. Technical report, SMPTE.

Jeni Tennison. 2012. Best practices for fragment identifiers and media type definitions. Technical report, World Wide Web (W3C).

Raphaël Troncy, Erik Mannens, Silvia Pfeiffer, and Davy Van Deursen. 2012. W3C Recommendation Media Fragments URI 1.0. Technical report, World Wide Web Consortium (W3C), September. Available at http://www.w3.org/TR/2012/REC-media-frags-20120925/.

Gabriela Vulcu, Paul Buitelaar, Sapna Negi, Bianca Pereira, Mihael Arcan, Barry Coughlan, J. Fernando Sánchez-Rada, and Carlos A. Iglesias. 2014. Generating Linked-Data based Domain-Specific Sentiment Lexicons from Legacy Language and Semantic Resources. In *th International Workshop on Emotion, Social Signals, Sentiment & Linked Open Data, co-located with LREC 2014*, Reykjavik, Iceland, May. LREC2014.

Rupert Westenhaler. 2014. Open Annotation and NIF 2.0 based Annotation Model for Apache Stanbol. In *Proceedings of ISWC 2014*.

Adam Westerski, Carlos A. Iglesias, and Fernando Tapia. 2011. Linked opinions: Describing sentiments on the structured web of data.

E. Wilde and M. Duerst. 2008. URI Fragment Identifiers for the text/plain Media Type. Technical Report RDF5147, Internet Engineering Task Force (IETF), April. Available at https://tools.ietf.org/html/rfc5147.

3.1.5 EUROSENTIMENT: Linked Data Sentiment Analysis

Title	EUROSENTIMENT: Linked Data Sentiment Analysis
Authors	Sánchez-Rada, J. Fernando and Vulcu, Gabriela and Iglesias, Carlos A. and Buitelaar, Paul
Proceedings	Proceedings of the ISWC 2014 Posters & Demonstrations Track a track within the 13th International Semantic Web Conference (ISWC 2014)
ISBN	
Volume	1272
Year	2014
Keywords	eurosentiment
Pages	145–148
Online	http://ceur-ws.org/Vol-1272/paper_116.pdf
Abstract	Sentiment and Emotion Analysis strongly depend on quality language resources, especially sentiment dictionaries. These resources are usually scattered, heterogeneous and limited to specific domains of application by simple algorithms. The EUROSENTIMENT project addresses these issues by 1) developing a common language resource representation model for sentiment analysis, and APIs for sentiment analysis services based on established Linked Data formats (lemon, Marl, NIF and ONYX) 2) by creating a Language Resource Pool (a.k.a. LRP) that makes available to the community existing scattered language resources and services for sentiment analysis in an interoperable way. In this paper we describe the available language resources and services in the LRP and some sample applications that can be developed on top of the EUROSENTIMENT LRP.

EUROSENTIMENT: Linked Data Sentiment Analysis

J. Fernando Sánchez-Rada[1], Gabriela Vulcu[2], Carlos A. Iglesias[1], and Paul Buitelaar[2]

[1] Dept. Ing. Sist. Telemáticos, Universidad Politécnica de Madrid,
{jfernando,cif}@gsi.dit.upm.es,
http://www.gsi.dit.upm.es
[2] Insight, Centre for Data Analytics at National University of Ireland, Galway
{gabriela.vulcu,paul.buitelaar}@insight-centre.org,
http://insight-centre.org/

Abstract. Sentiment and Emotion Analysis strongly depend on quality language resources, especially sentiment dictionaries. These resources are usually scattered, heterogeneous and limited to specific domains of application by simple algorithms. The EUROSENTIMENT project addresses these issues by 1) developing a common language resource representation model for sentiment analysis, and APIs for sentiment analysis services based on established Linked Data formats (lemon, Marl, NIF and ONYX) 2) by creating a Language Resource Pool (a.k.a. LRP) that makes available to the community existing scattered language resources and services for sentiment analysis in an interoperable way. In this paper we describe the available language resources and services in the LRP and some sample applications that can be developed on top of the EUROSENTIMENT LRP.

Keywords: Language Resources, Sentiment Analysis, Emotion Analysis, Linked Data, Ontologies

1 Introduction

This paper reports our ongoing work in the European R&D project EUROSENTIMENT, where we have created a multilingual Language Resource Pool (LRP) for Sentiment Analysis based on a Linked Data approach for modelling linguistic resources.

Sentiment Analysis requires language resources such as dictionaries that provide a sentiment or emotion value to each word. Just as words have different meanings in different domains, the associated sentiment or emotion also varies. Hence, every domain has its own dictionary. The information about what each domain represents or how the entries for each domain are related is usually undocumented or implied by the name of each dictionary. Moreover, it is common that dictionaries from different providers use different representation formats. Thus, it is very difficult to use different dictionaries at the same time.

In order to overcome these limitations, we have defined a Linked Data Model for Sentiment and Emotion Analysis, which is based on the combination of several vocabularies: the NLP Interchange Format (NIF) [1], to represent information about texts, referencing text in the web with unique URIs; the Lexicon Model for Ontologies (lemon) [2], to provide lexical information, and differentiate between different domains and senses of a word; Marl [5], to link lexical entries or senses with a sentiment; and Onyx [3], that adds emotive information.

The use of a semantic format not only eliminates the interoperability issue, but it also makes information from other Linked Data sources available for the sentiment analysis process. The EUROSENTIMENT LRP generates language resources from legacy corpora, linking them with other Linked Data sources, and shares this enriched version with other users.

In addition to language resources, the pool also offers access to sentiment analysis services with a unified interface and data format. This interface builds on the NIF Public API, adding several extra parameters that are used in Sentiment Analysis. Results are formatted using JSON-LD and the same vocabularies as for language resources. The NIF-compatible API allows for the aggregation of results from different sources.

The project documentation[3] contains further information about the EUROSENTIMENT format, APIs and tools.

2 Language Resources

The EUROSENTIMENT LRP contains a set of language resources (lexicons and corpora). The available EUROSENTIMENT language resources can be found here.[4] The user can see the domain and the language of each language resource. At the moment the LRP contains resources for electronics and hotel domains in six languages (Catalan, English, Spanish, French, Italian and Portuguese) and we are currently working on adding more language resources from other domains like telco, movies, food and music. Table 1 shows the number of reviews in each available corpus and the number of lexical entries in each available lexicon.

A detailed description of the methodology for creating the domain-specific sentiment lexicons and corpora to be added in the EUROSENTIMENT LRP was presented at LREC 2014 [4].

The EUROSENTIMENT demonstrator[5] shows how users can benefit from the LRP, including an interactive SPARQL query editor to access the resources and a faceted browser.

3 Sentiment Services

In addition to a model for language resources, EUROSENTIMENT also provides an API for sentiment and emotion analysis services. Several already existing ser-

[3] http://eurosentiment.readthedocs.org
[4] http://portal.eurosentiment.eu/home_resources
[5] http://eurosentiment.eu/demo

Lexicons

Language	Domains	#Entities
German	General	107417
English	Hotel,Electronics	8660
Spanish	Hotel,Electronics	1041
Catalan	Hotel,Electronics	1358
Portuguese	Hotel,Electronics	1387
French	Hotel,Electronics	651

Corpora

Language	Domains	#Entities
English	Hotel,Electronics	22373
Spanish	Hotel,Electronics	18191
Catalan	Hotel,Electronics	4707
Portuguese	Hotel,Electronics	6244
French	Electronics	22841

Table 1. Summary of the resources in the LRP

vices in different languages have been adapted to expose this API. Any user can benefit from these services, which are conveniently listed in the EUROSENTIMENT portal. At the moment, the following services are provided in several languages: language detection, domain detection, sentiment and emotion detection, and text analysis.

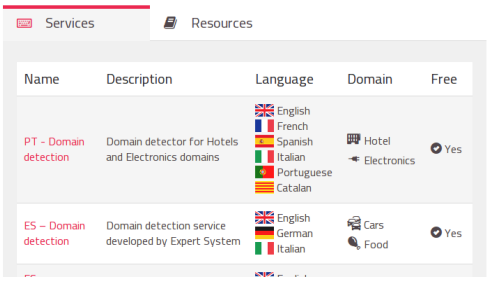


Fig. 1. The LRP provides a list of available services

4 Applications Using the LRP

To demonstrate the capabilities of the EUROSENTIMENT LRP, we open-sourced the code of several applications that make use of the services and resources of the EUROSENTIMENT LRP. The applications are written in different programming languages and are thoroughly documented. Using these applications as a template, it is straightforward to immediately start consuming the services and resources. The code can be found on the EUROSENTIMENT Github repositories.[6]

[6] http://github.com/eurosentiment

Fig. 2. Simple service that uses the resources in EUROSENTIMENT to analyse opinions in different languages and domains

Acknowledgements

This work has been funded by the European project EUROSENTIMENT under grant no. 296277

References

1. Hellmann, S., Lehmann, J., Auer, S., Nitzschke, M.: Nif combinator: Combining nlp tool output. In: Knowledge Engineering and Knowledge Management, pp. 446–449. Springer (2012)
2. McCrae, J., Spohr, D., Cimiano, P.: Linking lexical resources and ontologies on the semantic web with lemon. In: Antoniou, G., Grobelnik, M., Simperl, E., Parsia, B., Plexousakis, D., De Leenheer, P., Pan, J. (eds.) The Semantic Web: Research and Applications, Lecture Notes in Computer Science, vol. 6643, pp. 245–259. Springer Berlin Heidelberg (2011)
3. Sánchez-Rada, J.F., Iglesias, C.A.: Onyx: Describing emotions on the web of data. In: ESSEM@ AI* IA. pp. 71–82. Citeseer (2013)
4. Vulcu, G., Buitelaar, P., Negi, S., Pereira, B., Arcan, M., Coughlan, B., Sánchez-Rada, J.F., Iglesias, C.A.: Generating Linked-Data based Domain-Specific Sentiment Lexicons from Legacy Language and Semantic Resources. In: th International Workshop on EMOTION, SOCIAL SIGNALS, SENTIMENT & LINKED OPEN DATA, co-located with LREC 2014,. LREC2014, Reykjavik, Iceland (May 2014)
5. Westerski, A., Iglesias, C.A., Tapia, F.: Linked Opinions: Describing Sentiments on the Structured Web of Data. In: Proceedings of the 4th International Workshop Social Data on the Web (2011)

3.1.6 Linguistic Linked Data for Sentiment Analysis

Title	Linguistic Linked Data for Sentiment Analysis
Authors	Buitelaar, Paul and Arcan, Mihael and Iglesias, Carlos A. and Sánchez-Rada, J. Fernando and Strapparava, Carlo
Proceedings	2nd Workshop on Linked Data in Linguistics (LDL-2013): Representing and linking lexicons, terminologies and other language data. Collocated with the Conference on Generative Approaches to the Lexicon
ISBN	978-1-937284-77-0
Year	2013
Keywords	sentim
Pages	1–8
Online	https://www.aclweb.org/anthology/W/W13/W13-55.pdf
Abstract	In this paper we describe the specification of a model for the semantically interoperable representation of language resources for sentiment analysis. The model integrates 'lemon', an RDF-based model for the specification of ontology-lexica (Buitelaar et al. 2009), which is used increasingly for the representation of language resources as Linked Data, with 'Marl', an RDF-based model for the representation of sentiment annotations (Westerski et al., 2011; Sánchez-Rada et al., 2013).

Linguistic Linked Data for Sentiment Analysis

Paul Buitelaar, Mihael Arcan
DERI, Unit for NLP,
National University of Ireland, Galway
{paul.buitelaar, mihael.arcan}@deri.org

Carlos A. Iglesias, J. Fernando Sánchez-Rada
Dept. Ing. Sist. Telemáticos,
Univ. Politécnica de Madrid,
Spain
{cif, jfernando}@gsi.dit.upm.es

Carlo Strapparava
Human Language Technology
FBK, Italy
strappa@fbk.eu

1 Introduction

In this paper we describe the specification of a model for the semantically interoperable representation of language resources for sentiment analysis. The model integrates 'lemon', an RDF-based model for the specification of ontology-lexica (Buitelaar et al. 2009), which is used increasingly for the representation of language resources as Linked Data, with 'Marl', an RDF-based model for the representation of sentiment annotations (Westerski et al., 2011; Sánchez-Rada et al., 2013).

In the EuroSentiment project, the lemon/Marl model will be used to represent lexical resources for sentiment and emotion analysis such as SentiWordNet (Baccianella et al. 2010) and WordNet Affect[1] (Strapparava and Valitutti 2004), as well as other language resources such as sentiment annotated corpora, in a semantically interoperable way, using Linked data principles.

The representation of WordNet resources in lemon depends on a straightforward conversion of the WordNet data model, but importantly we introduce the use of URIs to uniquely and formally define structure and content of this WordNet based language resource. URIs are adopted from existing Linked Data resources, thereby further enhancing semantic interoperability. We further integrate a notion of domains into this representation in order to enable domain-specific definition of polarity for each lexical item.

The lemon model allows for the representation of all aspects of lexical information, including lexical sense (word meaning) and polarity, but also morphosyntactic features such as part-of-speech, inflection, etc. This kind of information is not provided by WordNet Affect but will be available from other language resources, including those available at EuroSentiment partners that can be easily integrated with the WordNet Affect information using lemon.

The representation of sentiment polarity uses concepts from Marl.

2 Motivation

Sentiment analysis is now an established field of research and a growing industry (Po et al. 2008). However, language resources for sentiment analysis are being developed by individual companies or research organisations and are normally not shared, with the exception of a few publicly available resources such as WordNet Affect and SentiWordNet. Domain-specific resources for multiple languages are potentially valuable but not shared, sometimes due to IP and licence considerations, but often because of technical reasons, including interoperability.

In the EuroSentiment project we envision instead a pool of semantically interoperable language resources for sentiment analysis, including domain-specific lexicons and annotated corpora. Sentiment analysis applications will be able to: access domain-specific polarity scores for individual lexical items in the context of semantically defined sentiment lexicons and corpora, or access and integrate complete language resources. Access may be restricted according to commercial considerations, with payment schedules in place, or may be partially free. A semantic service access layer will be put in place for this purpose.

3 The lemon Model

The lexicon model for ontologies (lemon) builds on previous work on standards for the representation of lexical resources, i.e., the Lexical Markup Framework (LMF[2]) but extends the underlying formal model and provides a native integration of lexica with domain ontologies. The lemon model is

[1] http://wndomains.fbk.eu/wnaffect.html

[2] http://www.lexicalmarkupframework.org/

described in detail in the lemon cookbook (McCrae et al. 2010). Here we provide a summary of its most prominent features, starting with the lemon core, which is organized around a core path as follows:

- *Ontology Entity*: URI of an ontology element to which a Lexical Form points, providing a possible linguistic realisation for that Ontology Entity
- *Lexical Sense*: functional object that links a Lexical Entry to an Ontology Entity, providing a sense-disambiguated interpretation of that Lexical Entry
- *Lexical Entry*: morpho-syntactic normalisation of one or more Lexical Form
- *Lexical Form*: morpho-syntactic variant of a Lexical Entry, including inflection, declination and syntactic variation
- *Representation*: standard written or phonetic representation for a Lexical Form

In addition, lemon has a number of modules that allow for further modelling. Currently defined modules are: linguistic description, phrase structure, morphology, syntax and mapping, variation.

The linguistic description module is concerned with the use of ISOcat data categories for describing lemon elements. Although lemon itself is a meta-model and therefore agnostic as regards the specific data category set used, we use a specific set of data categories in particular instances of the lemon model, such as LexInfo (Cimiano et al. 2011).

The phrase structure module is concerned with the modelling of lexical entries that are syntactically complex, such as phrases and clauses. The module provides tokenisation and phrase structure analysis to enable representation of the syntactic structure of such lexical entries.

The morphology module is concerned with the analysis and representation of inflectional and agglutinative morphology. The module allows the specification of regular inflections of words by use of Perl-like regular expressions, which greatly simplifies the creation of lexical entries for highly synthetic and inflectional languages.

The syntax and mapping module is concerned with a description of lexical 'predicates' (subcategorisation frames with syntactic arguments) and semantic predicates (properties with subject/object) on the ontology side and the mapping between them. The module allows a mapping to be specified as a one-to-one correspondence.

The variation module is concerned with a description of the relationships between the elements of a lemon lexicon, which are split into three classes: sense relations, lexical variations, form variations. Sense relations require a semantic context, such as translation. Lexical variations require a morphosyntactic context, such as plural. Form variations are all other variations, such as homographs.

An interesting aspect of lemon-based ontology lexicalisation is the use of URIs for uniquely identifying all objects defined by the lemon model (lexicons, lexical entries, words, phrases, forms, variants, senses, references, etc.), which can be linked and maintained in a flexible, modular and distributed way. The lemon model can therefore contribute significantly to the development of Lexical Linked Data (McCrae et al. 2011, Nuzzolese et al. 2011, McCrae et al. 2012), which in turn will greatly enhance distributed development, exchange, maintenance and use of lexical resources as well as of ontologies as they will be increasingly tightly integrated with lexical knowledge.

In the context of the EuroSentiment project we will exploit the lemon model exactly for this purpose: representing language resources for sentiment analysis in a Linked Data conform way (RDF-native form), enabling leverage of existing Semantic Web technologies (SPARQL, OWL, RIF etc.).

4 The Marl Sentiment Ontology

Marl is an ontology for annotating sentiment expressions, which will be used by the EuroSentiment service layer to describe the output of sentiment analysis services as well as by the resource layer to describe the sentiment properties of lexical entries. For this latter purpose in particular, the Marl ontology is used in combination with lemon as illustrated above.

The Marl ontology is a vocabulary designed for annotation and description of subjective opinions expressed in text. The goals of the Marl ontology are to:

- enable publishing raw data about opinions and the sentiments expressed in them
- deliver schema that will allow to compare opinions coming from different systems (polarity, topics and features)

• interconnect opinions by linking them to contextual information expressed from other popular ontologies or specialised domain ontologies.

The Marl ontology has been extended according to the needs of the EuroSentiment project. In particular, the main extension has been its alignment with the PROV-O Ontology (Lebo, 2013) in order to support provenance modelling. The PROV-O ontology is part of the PROV Family (Groth, 2012; Gil, 2012) that provides support for modelling and interchange of provenance on the Web and Information Systems.

Provenance is information about entities, activities and people involved in producing a piece of data or thing, which can be used to form assessment about its quality, reliability and trustworthiness. The main concepts of PROV are entities, activities and agents. Entities are physical or digital assets, such as web pages, spell checkers or, in our case, dictionaries or analysis services. Provenance records describe the provenance of entities, and an entity's provenance can refer to other entities. For example, a dictionary is an entity whose provenance refers to other entities such as lexical entries. Activities are how entities come into existence. For example, starting from a web page, a sentiment analysis activity creates an opinion entity describing the extracted opinions from that web page. Finally, agents are responsible for the activities and can be a person, a piece of software, an organisation or other entities. The Marl ontology has been aligned with the PROV ontology so that provenance of language resources can be tracked and shared.

Sentiment Analysis is an Activity that analyses a Source text according to an algorithm and produces an opinion about the entities described in the source text. The main features of the extracted opinion are the polarity (positive, neutral or negative), the polarity value or strength whose range is defined between a min and max value, and the described entity and feature of that opinion. Opinions can also be aggregated opinions of a set of users.

For a better understanding of the ontology itself, we present below the main classes and properties that form the ontology:

• *Opinion*: a subclass of the Provenance Entity that represents the results of a Sentiment Analysis process. Among its classes we find:

• *describesObject*: property that points to the object the opinion refers to.

• *describesObjectPart*: optional property, used whenever the opinion specifies the part of the object it refers to, not only the general object.

• *describesObjectFeature*: aspect of the object or part that the user is giving an opinion of.

• *hasPolarity*: polarity of the opinion itself, to be chosen from the available Opinion individuals.

• *polarityValue*: degree of the polarity. In other words, it represents how strong the opinion (independently of the polarity) is.

• *algorithmConfidence*: rating the analysis algorithm has given to this particular result. Can be interpreted as the accuracy or trustworthiness of the information

• *extractedFrom*: original source text or resource from which the opinion was extracted.

• *opinionText*: part of the source that was used in the sentiment analysis. That is, the part of the source that contained sentiment information.

• *domain*: context domain of the result. The same source can be analysed in different domains, which would lead to different results.

• *AggregatedOpinion*: when several opinions are equivalent, we can opt to aggregate them into an "AggregatedOpinion", which in addition to the properties we already covered, it presents these properties:

• *opinionCount*: the number of individual opinions this AggregatedOpinion represents.

• *Polarity*: base class to represent the polarity of the opinion. In every opinion, we will use an instance of this class. The base Marl ontology comes with three instances: Positive, Negative, Neutral

• *SentimentAnalysis*: in Marl, the process of sentiment analysis is also represented semantically, which allows us to understand the opinion data, trace it and keep several results by different algorithms, linking all of them to the process that created them. The main properties of each SentimentAnalysis class are: minPolarityValue: lower limit for polarity values in the opinions extracted via this analysis activity; maxPolarityValue: upper limit for polarity values in the opinions extracted via this analysis activity.

• *Algorithm*: algorithm that was used in the analysis. Useful to group opinions by extraction algorithm and compare them.

• *source*: site or source from which the opinion was extracted. There are two reasons behind this property: grouping by opinion source (e.g. opin-

ions from IMDB) and treating and interpreting opinions from the same source in the same manner.

An example application of the Marl ontology for a sentiment analysis service is shown in the Appendix. It is split in two: a view of the representation of the analysis (Fig 1), and a representation of the result (Fig 2).

5 Representation of WordNet Affect

In this section we describe how language resources based on the Princeton WordNet model (Miller 1995) such as WordNet Affect can be represented using lemon.

WordNet Affect is an extension of the WordNet database, including a subset of synsets suitable to represent affective concepts. Similarly to the extension related to domain labels, one or more affective labels (a-labels) are assigned to a number of WordNet synsets. In particular, the affective concepts representing emotional state are individuated by synsets marked with the a-label 'emotion'. The emotional categories are hierarchically organized in order to specialize synsets with a-label emotion and to distinguish synsets according to emotional valence. There are also other a-labels for concepts representing moods, situations eliciting emotions, or emotional responses[3].

Unique and independently established URIs for WordNet synsets allow for a distributed representation that enable Semantic Web based linking between and integration of WordNet based as well as other language resources. We illustrate this here with an example from WordNet Affect, using English based WordNet 3.0 URIs as defined by the Europeana project.

Consider the following example for the English noun 'fear' in WordNet and equivalent Italian synonyms taken from the Italian WordNet (i.e. this holds for any English aligned Wordnet) in WordNet Affect:

Princeton WordNet:

n#05590260 12 n 03 fear 0 fearfulness 0 fright 0 017 @ 05560878 n 0000 ! 05595229 n 0101 = 00080744 a 0000 = 00084648 a 0000 ~ 05590744 n 0000 ~ 05590900 n 0000 ~ 05591021 n 0000 ~ 05591212 n 0000 ~ 05591290 n 0000 ~ 05591377 n 0000 ~ 05591481 n 0000 ~ 05591591 n 0000 ~ 05591681 n 0000 ~ 05591792 n 0000 ~ 05592739 n 0000 ~ 05593389 n 0000 %p 10337259 n 0000 | an emotion experienced in anticipation of some specific pain or danger (usually accompanied by a desire to flee or fight)

WordNet Affect:

n#05590260 fifa paura spavento terrore timore | "una emozione che si prova prima di qualche specifico dolore o pericolo"
n#05590260 affective-label="negative-fear"
n#05590260 domain-label="Psychological_Features"

***lemon* transformation & integration:**

Using lemon we can represent and integrate information on the Italian synonyms, their links to the English based synset using Princeton WordNet URIs, and sentiment properties using Marl. Domain properties will be based on WordNet Domains[4]. The example illustrates the positive polarity of 'fear' in English (and 'fifa, paura, spavento, terrore' in Italian) in the context of 'horror movies' and negative polarity in the context of 'children movies'.

Declaration of namespaces used – *wn* declares WordNet 3.0 synsets, *lemon* declares the core lemon lexicon model, *lexinfo* declares specific properties for part-of-speech etc., *wd* declares domain categories, *marl* declares sentiment properties:

```
@prefix wn:
<http://semanticweb.cs.vu.nl/europeana/lod/purl/vocabularies/princeton/wn30/> .
@prefix lemon: <http://www.monnet-project.eu/lemon#> .
@prefix lexinfo:
<http://www.lexinfo.net/ontology/2.0/lexinfo#> .
@prefix wd: <http://www.eurosentiment.eu/wndomains/> .
@prefix marl: <http://purl.org/marl/ns#> .
```

Declaration of lexicon identifier, language and lexical entries:

```
:lexicon a lemon:Lexicon ;
  lemon:language "it" ;
  lemon:entry    :fifa,
                 :paura,
                 :spavento,
                 :terrore.
```

[3] A SKOS version of WordNet Affect is available from http://gsi.dit.upm.es/ontologies/wnaffect/

[4] http://wndomains.fbk.eu/

Declaration of lemma, sense (link to synset in WordNet 3.0, polarity and domain context) and part-of-speech of ‘fifa’:

```
:fifa a lemon:Lexicalentry ;
  lemon:canonicalForm [ lemon:writtenRep
"fifa"@it ] ;
  lemon:sense [ lemon:reference wn:synset-fear-noun-1;
          marl:polarityValue 0.375 ;
          marl:hasPolarity marl:positive ;
          lemon:context wd:horror_movies ] ;
  lemon:sense [ lemon:reference wn:synset-fear-noun-1;
           marl:polarityValue 0.375 ;
           marl:hasPolarity marl:negative ;
           lemon:context wd:children_movies ];
  lexinfo:partOfSpeech lexinfo:noun .
```

Declarations of lemma and part-of-speech of ‘paura, spavento, terrore, timore’:

```
:paura a lemon:Lexicalentry ;
   lemon:canonicalForm [ lemon:writtenRep
          "paura"@it ] ;
   lexinfo:partOfSpeech lexinfo:noun .

:spavento a lemon:Lexicalentry ;
   lemon:canonicalForm [ lemon:writtenRep
          "spavento"@it ] ;
   lexinfo:partOfSpeech lexinfo:noun .

:terrore a lemon:Lexicalentry ;
   lemon:canonicalForm [ lemon:writtenRep
          "terrore"@it ] ;
   lexinfo:partOfSpeech lexinfo:noun .

:timore a lemon:Lexicalentry ;
   lemon:canonicalForm [ lemon:writtenRep
          "timore"@it ] ;
   lexinfo:partOfSpeech lexinfo:noun .
```

Declarations of sense equivalence (synonymy) of ‘paura, spavento, terrore, timore’ with ‘fifa’:

```
:paura a lemon:LexicalSense ;
          lemon:equivalent :fifa.

:spavento a lemon:LexicalSense ;
          lemon:equivalent :fifa.

:terrore a lemon:LexicalSense ;
          lemon:equivalent :fifa.

:timore a lemon:LexicalSense ;
          lemon:equivalent :fifa..
```

6 Representation of Lexical and Sentiment Features

The examples discussed in the previous section showed the representation of WordNet based language resources with lemon. However also many other types of language resources exist, including sentiment dictionaries maintained by the EuroSentiment use case partners that define domain words with their polarity scores as well as inflectional variants, part-of-speech, etc. We can also represent such language resources using lemon combined with Marl, thereby making them interoperable with the lemon version of WordNet Affect as well as other lemon based language resources.

Consider the following example for the German noun ‘Einschlag’ (‘impact’) with lexical features (inflection, part-of-speech) and polarity score:

```
Einschlag    Einschlag NN negative -/-0.0048/- L
Einschlages Einschlag NN negative -/-0.0048/- L
Einschlags   Einschlag NN negative -/-0.0048/- L
Einschläge   Einschlag NN negative -/-0.0048/- L
Einschlägen Einschlag NN negative -/-0.0048/- L
```

Using lemon and Marl we can represent this and integrate it with additional information as follows:

Declaration of namespaces used – *wn* declares WordNet 3.0 synsets, *lemon* declares the core lemon lexicon model, *isocat* declares specific properties for part-of-speech etc. (*isocat* is part of the *lexinfo* model used in the previous example), *marl* declares sentiment properties:

```
@prefix wn:
<http://semanticweb.cs.vu.nl/europeana/lod/purl/vocabularies/princeton/wn30/> .
@prefix lemon: <http://www.monnet-project.eu/lemon#> .
@prefix isocat: <https://catalog.clarin.eu/isocat/interface/index.html> .
@prefix marl:
<http://gsi.dit.upm.es/ontologies/marl/ns#> .
```

Declaration of lexicon identifier, language and lexical entry:

```
:lexicon a lemon:Lexicon ;
   lemon:language "de" ;
   lemon:entry :Einschlag.
```

Declaration of lemma, sense (link to synset in WordNet 3.0, polarity), alternate forms (inflectional variants with features), part-of-speech and sentiment polarity:

```
:Einschlag
```

```
lemon:canonicalForm [
  lemon:writtenRep "Einschlag"@de ;
       isocat:DC-1297 isocat:DC-1883 ;
       # gender=masculine
       isocat:DC-1298 isocat:DC-1387 ;
       # number=singular
       isocat:DC-2720 isocat:DC-1331 ] ;
       # case=nominative
  lemon:sense [ lemon:reference
       wn:synset-impact-noun-1;
       marl:polarityValue 0.0048;
        marl:hasPolarity marl:negative ] ;
  lemon:altForm
    [ lemon:writtenRep "Einschlages"@de ;
       isocat:DC-1297 isocat:DC-1883 ;
       # gender=masculine
       isocat:DC-1298 isocat:DC-1387 ;
       # number=singular
       isocat:DC-2720 isocat:DC-1293 ] ;
       # case=genitive
    [ lemon:writtenRep "Einschlags"@de ;
       isocat:DC-1297 isocat:DC-1883 ;
       # gender=masculine
       isocat:DC-1298 isocat:DC-1387 ;
       # number=singular
       isocat:DC-2720 isocat:DC-1293 ] ;
       # case=genitive
    [ lemon:writtenRep "Einschläge"@de ;
       isocat:DC-1297 isocat:DC-1883 ;
       # gender=masculine
       isocat:DC-1298 isocat:DC-1354 ;
       # number=plural
       isocat:DC-2720 isocat:DC-1331 ] ;
       # case=nominative
    [ lemon:writtenRep "Einschlägen"@de ;
       isocat:DC-1297 isocat:DC-1883 ;
       # gender=masculine
       isocat:DC-1298 isocat:DC-1354 ;
       # number=plural
       isocat:DC-2720 isocat:DC-1265 ] ;
       # case=dative
isocat:DC-1345 isocat:DC-1333.
# partOfSpeech=noun.
```

7 Ongoing and Future Work

Sentiment Analysis aims at determining the attitude of the writer to some topic (positive, negative, neutral). Emotion analysis goes one step further and aims at determining the emotional or affective state of the writer when writing. In EuroSentiment, we have defined two vocabularies for annotating sentiment and emotion expressions, called Marl and Onyx, respectively. In this paper we focused on the representation of sentiment annotations with Marl. The definition and representation of emotion expressions with Onyx is ongoing work, with the objective of covering different theoretical models of emotions (Sánchez-Rada et al., 2013). Onyx will support the representation and use of several emotion taxonomies such as WordNet Affect or EmotionML

Our ongoing and future work is concerned also with the definition and implementation of a work flow that will enable the generation of domain-specific semantically interoperable lexica for sentiment analysis. The work flow will use lemon and Marl for the representation and integration of:

- WordNet Domains information on domain(s)
- domain entity information from DBpedia and/or other relevant semantic resources
- WordNet Affect information on synsets (using Onyx)
- morphosyntactic information (part-of-speech, inflection, …) from other language resources in the EuroSentiment Language Resource Pool
- SentiWordNet scores and/or automatically extracted domain sentiment scores

Given a particular sentiment analysis task domain, the approach is based on the analysis of a representative text collection for the purpose of entity identification, synset disambiguation, morphosyntactic analysis, and domain-specific polarity value extraction.

8 Conclusions

We presented a model for the specification, integration and use of language resources for sentiment analysis based on Linked Data principles.

The presented model is based directly on the lemon and Marl ontologies for the representation of Linked Data based lexical resources and sentiment expressions respectively. This work is now being extended so that emotion analysis is also addressed.

In the context of the EuroSentiment project the combined model will be used for the integrated and semantically interoperable representation of sentiment dictionaries and annotations. As a result, EuroSentiment will make available lexical resources based on this interoperable representation with the aim of fostering the development of services using sentiment analysis.

Acknowledgments

This work was partially funded by the EC for the FP7 project EuroSentiment under Grant Agreement 296277 and in part by a research grant from Science Foundation Ireland (SFI) under Grant Number SFI/12/RC/2289 for the INSIGHT project.

References

Baccianella, Stefano, Andrea Esuli and Fabrizio Sebastiani, “SENTIWORDNET 3.0: An Enhanced Lexical Re-source for Sentiment Analysis and Opinion Mining”, Proc. of LREC, 2010.

Buitelaar, Paul, Philipp Cimiano, Peter Haase, and Michael Sintek. "Towards linguistically grounded ontologies." In The Semantic Web: Research and Applications, pp. 111-125. Springer Berlin Heidelberg, 2009.

Cimiano, Philipp, Paul Buitelaar, John McCrae, and Michael Sintek. "LexInfo: A declarative model for the lexicon-ontology interface." Web Semantics: Science, Services and Agents on the World Wide Web 9, no. 1 (2011): 29-51.

Ekman, Paul. "Basic emotions." Handbook of cognition and emotion 98 (1999): 45-60.

Gil, Yolanda and Simon Miles, “PROV Model Primer”, W3C Working Draft, 11th December 2012, available at http://www.w3.org/TR/2012/WD-prov-primer-20121211/.

Groth, Paul and Luc Moreau, “PROV Overview”, W3C Working Draft, 11th December 2012, available at http://www.w3.org/TR/2012/WD-prov-overview-20121211/.

Lebo, Timothy, Satya Sahoo and Deborah McGuinness, “PROV-O: The PROV Ontology”, W3C Recommendation, 30th April 2013, available at http://www.w3.org/TR/prov-o/, 2013.

McCrae, John, Guadalupe Aguado-de-Cea, Paul Buitelaar, Philipp Cimiano, Thierry Declerck, Asunción Gómez-Pérez, Jorge Gracia et al. "Interchanging lexical resources on the semantic web." Language Resources and Evaluation 46, no. 4 (2012): 701-719.

McCrae, John, Dennis Spohr, and Philipp Cimiano. "Linking lexical resources and ontologies on the semantic web with lemon." In The Semantic Web: Research and Applications, pp. 245-259. Springer Berlin Heidelberg, 2011.

McCrae, John, Guadalupe Aguado-de-Cea, Paul Buitelaar, Philipp Cimiano, Thierry Declerck, Asunción Gómez Pérez, Jorge Gracia et al. "The lemon cookbook." (2010).

Miller, George A. "WordNet: a lexical database for English." Communications of the ACM 38.11 (1995): 39-41.

Nuzzolese A, Gangemi A, Presutti V (2011) Gathering lexical linked data and knowledge patterns from framenet. In: Proceedings of the sixth international conference on Knowledge capture, ACM, pp 41–48.

Pang, Bo and Lee, Lillian, “Opinion mining and sentiment analysis” Foundations and trends in information retrieval, 2008.

Prinz, Jesse. Gut Reactions: A Perceptual Theory of Emotion (Oxford: Oxford University Press, 2004): page 157

Sánchez-Rada, J. Fernando, Onyx Ontology Specification, V1.2 July 2013, available at http://www.gsi.dit.upm.es/ontologies/onyx/

Strapparava, Carlo, and Alessandro Valitutti. "WordNet-Affect: an affective extension of WordNet." In Proceedings of LREC, vol. 4, pp. 1083-1086. 2004.

Westerski, Adam, Carlos A. Iglesias and Fernando Tapia, “Linked Opinions: Describing Sentiments on the Structured Web of Data.” In Proceedings of the 4th International Workshop Social Data on the Web, 2011.

Westerski, Adam and Sánchez-Rada, J. Fernando, Marl Ontology Specification, V1.0 May 2013, available at http://www.gsi.dit.upm.es/ontologies/marl/

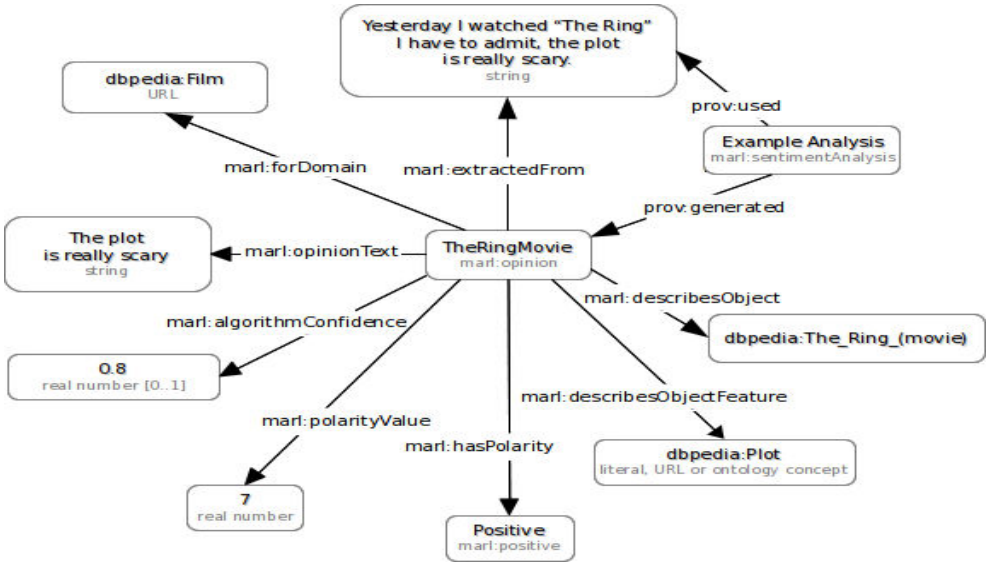


Figure 1: Example of a Sentiment Analysis activity representation

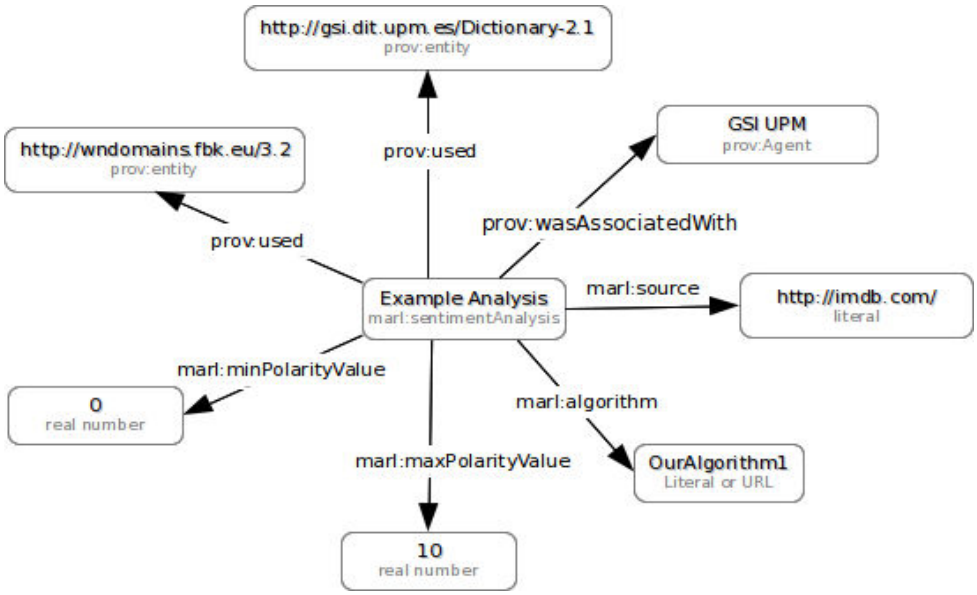


Figure 2: Example Sentiment Analysis result

3.1.7 A Linked Data Approach to Sentiment and Emotion Analysis of Twitter in the Financial Domain

Title	A Linked Data Approach to Sentiment and Emotion Analysis of Twitter in the Financial Domain
Authors	Sánchez-Rada, J. Fernando and Torres, Marcos and Iglesias, Carlos A. and Maestre, Roberto and Peinado, Raquel
Proceedings	Second International Workshop on Finance and Economics on the Semantic Web (FEOSW 2014)
ISBN	
Volume	1240
Year	2014
Keywords	emotions, finance, linked data, semantic, sentiment analysis
Pages	51–62
Online	http://ceur-ws.org/Vol-1240/
Abstract	Sentiment analysis has recently gained popularity in the financial domain thanks to its capability to predict the stock market based on the wisdom of the crowds. Nevertheless, current sentiment indicators are still silos that cannot be combined to get better insight about the mood of different communities. In this article we propose a Linked data approach for modelling sentiment and emotions about financial entities. We aim at integrating sentiment information from different communities for providers, and complements existing initiatives such as FIBO. The approach has been validated in the semantic annotation of tweets of several stocks in the Spanish stock market, including its sentiment information.

A Linked Data Approach to Sentiment and Emotion Analysis of Twitter in the Financial Domain

J. Fernando Sánchez-Rada[1], Marcos Torres[1], Carlos A. Iglesias[1], Roberto Maestre[2], and Esther Peinado[2]

[1] Universidad Politécnica de Madrid,
ETSI Telecomunicación, Avda. Complutense, 30, 28040 Madrid, Spain
[2] Paradigma Labs, Paradigma Tecnológico,
Avda. Europa, 26, Pozuelo de Alarcón, 28224 Madrid, Spain

Abstract. Sentiment analysis has recently gained popularity in the financial domain thanks to its capability to predict the stock market based on the wisdom of the crowds. Nevertheless, current sentiment indicators are still silos that cannot be combined to get better insight about the mood of different communities. In this article we propose a Linked Data approach for modelling sentiment and emotions about financial entities. We aim at integrating sentiment information from different communities or providers, and complements existing initiatives such as FIBO. The approach has been validated in the semantic annotation of tweets of several stocks in the Spanish stock market, including its sentiment information.

Keywords: linked data, semantic, finance, sentiment analysis, emotions

1 Introduction

The proliferation of user generated content in web sites and social networks, such as Facebook, TripAdvisor or Twitter, has lead to an increased awareness of the power of social networks for expressing opinions about products, services and even disasters. These so-called social sensors enable real time indexing of the social web with the aim of providing insight about the structure and activity of social networks. They provide a vast array of application possibilities, from monitoring brands or products to become early disaster warning systems [1].

In the financial field, social sensors can provide additional valuable information that complements other sources of information used in fundamental analysis, such as financial newspapers. In particular, sentiment analysis has been one of the most popular technologies to measure the investment mood. The sentiment stock market indicator has become a popular indicator that is provided together with the classical fundamental and technical stock market indicators [2]. Several websites provide the investor emotion index[3] or their sentiment, like AII Investor

[3] Market Emotion by CNN Money available at `http://money.cnn.com/data/fear-and-greed/`

Sentiment Survey[4], StockMarketSensor[5], or SentimentTrader[6], just to name a few.

In addition, recent research has shown that sentiment expressed in microblogging sites such as Twitter can be applied to predict daily changes in stock values [3,4].

Linked Data is another valuable resource that can provide financial analysts with an integration of available data sources in their activity [5]. Linked Data can provide a wide array of opportunities in the financial field. As reported by O'Riain et al. [6], depending on the information consumer needs, the integration and augmentation of financial information can lead to a significant benefit for financial and business analysis in tasks such as competitive analysis, fraud detection or figures comparison. It is also worth mentioning the recent trend towards open government and eGovernment data initiatives for public sector information, statistics data and economic indicators. The current status is promising, with a large volume of financial and economic data sets already available. Several researchers have shown this potential for different use cases, such as cross-lingual query of financial and business data from multiple sources [7,8], using social media in investment decisions [9,10] or enriching corporate financial reporting [11].

The aim of this article is the application of a Linked Data approach to expressing sentiments and emotions about financial concepts, which financial analysts can use to combine opinions expressed in different social media sites.

The article is arranged as follows. Sect. 2 gives an overview of the vocabularies we have defined for modelling sentiment and opinions as well as its interlinking with financial vocabularies such as FIBO [12]. Sect. 3 outlines our system design. Sect. 4 provides an overview of our experimental design and results. Sect. 5 expresses our conclusions and a brief discussion of future directions for this line of research.

2 Modeling Sentiment and Emotions as Linked Data

This section provides insight about the potential of Linked Data for accessing, interlinking and reasoning about business data sources. To leverage that power, it is necessary to have a robust representation model for sentiment in the financial context. Rather than creating an ad-hoc model, the Linked Data approach is to look for models for each domain and connect them. In particular, we will need a model for financial entities, a model for sentiment analysis results, and a model for microblogging messages. The following sections review the models (also referred to as ontologies or vocabularies) available in these domains, and Sect. 2.3 exemplifies the use of the final integrated model.

[4] AII Investor Sentiment Survey available at `http://www.aaii.com/sentimentsurvey`
[5] Available at `http://www.stockmarketsensor.com/`
[6] Available at `http://www.sentimentrader.com/`

2.1 Linked Data in the Financial Domain

Financial Industry Business Ontology (FIBO) [12] is a collaborative industry initiative to describe financial data standards using semantic technology. FIBO has been authored by Enterprise Data Management (EDM) council under the technical governance of the Object Management Group (OMG). FIBO has two distinct aspects: a business ontology and a presentation for business readability. FIBO is released in discrete ontologies by subject area: (i) Business Entities; (ii) Security, Loans, Derivatives and (iii) Corporate Actions and Transactions. At the time of this writing, only the first specification for Business Entities has been made public. The specification identifies a taxonomy of basic entities: Human Being, Legal Person, Organization and Legal Entity. This taxonomy is extended with other derived entities, such as Minor, Natural Person, Artificial Person (Company Limited by Guarantee, Legally Incorporated Partnership, Foundation or Incorporated Company), Formal Organization (Trust, Partnership or Incorporated Company) and Informal Organization. In addition, the ontology models concepts such as control and ownership.

Financial Exchange Framework Ontology (FEF) [13] is an ontology defined by International Financial Information Publishing (IFIP) Ltd. with the aim of providing an enterprise-wide publication and integration standard. FEF ontology provides support for modelling financial components and financial entities.

The FP7 FIRST Project (Large Scale Information Extraction and Integration Infrastructure for Supporting Financial Decision Making) has defined an ontology for sentiment analysis in financial domains [9,10]. The ontology identifies Orientation Term (OT), Financial Instrument (FI) and Indicator (I) and their relationships. In addition, the ontology conceptualises specialisations of FI (stocks and stock indexes), economic indicators, and relationships among them. Based on this ontology, the project FIRST has elaborated a set of ontologies for currencies, companies, financial instruments (stocks and stock indexes), funds, financial institutions, insurance companies and banks, available at FIRST project[7].

In its simple form, a FIBO definition would be a single triple. However, FIBO is a complete ontology that enables much more powerful assertions, as will be shown later.

2.2 Linked Opinions and Emotions about stocks

In this section we introduce two vocabularies, Marl and Onyx, that we have defined for providing a uniform vocabulary for expressing sentiments and emotions, respectively, according to linked data principles.

Marl [14] is a standardised data schema designed to annotate and describe subjective opinions expressed on the web or in particular Information Systems. Its aim is to show the benefits of publishing in the open, on the Web, the results of the opinion mining process in a structured form. On the road to achieving

[7] http://first.ijs.si/firstontology/

this, Marl attempts to answer the research question of to what extent opinion information can be formalised in a unified way.

Marl is the result of analysing the properties that characterise opinions expressed on the web or inside various IT systems. The final set of concepts proposed is shown in Fig. 1. It should be noted that opinions in Marl are meant to be linked to an entity. Such entity can be a FIBO Corporation, as described in the previous section. We will make use of this property in Section 2.3.

A detailed description of each particular property and an explanation of their meaning can be found in the vocabulary's specification [8].

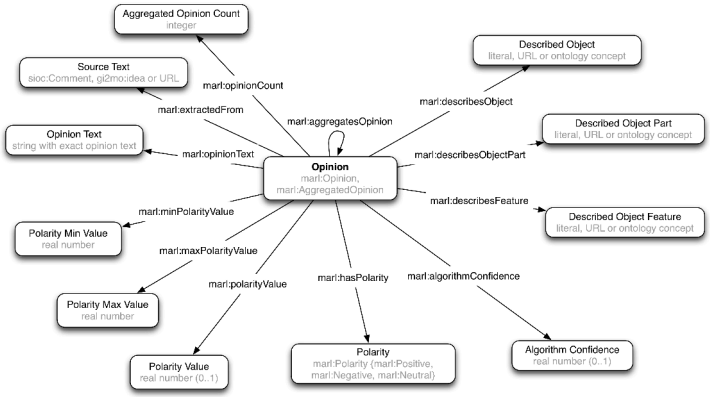


Fig. 1. Marl entities

Onyx [15] is a vocabulary to represent the Emotion Analysis process and its results, as well as annotating lexical resources for Emotion Analysis. It includes all the necessary classes and properties to provide structured and meaningful Emotion Analysis results, and to connect results from different providers and applications.

At its core, the Onyx ontology has three main classes: EmotionAnalysis, EmotionSet and Emotion. In a standard Emotion Analysis, these three classes are related as follows: an EmotionAnalysis is run on a source (generally in the form of text, e.g. a status update), the result is represented as one or more EmotionSet instances that contain one or more Emotion instances.

The specification of the Onyx vocabulary [9] contains an updated description of all its elements, with some usage examples.

[8] `http://www.gsi.dit.upm.es/ontologies/marl`
[9] `http://www.gsi.dit.upm.es/ontologies/onyx`

2.3 Using Linked Data

First of all, let us review a simplified version of the integration of all the elements that we described. To keep it as simple as possible, we will avoid any provenance information (such as who or how analised the twit to extract the opinion) or information about the post itself (author, date, etc.) This simplicity will not prevent us from harnessing the potential of Linked Data.

Listing 1.1. Simple representation using FIBO

```
ex:myOpinion a marl:Opinion;
             marl:hasPolarityValue marl:Positive;
             marl:describesObject ex:GSantander;
             marl:extractedFrom ex:twit1.
ex:twit1 a sioct:MicroblogPost;
         sioc:content "I like testing Grupo Santander".
ex:GSantander a fibo:IncorporatedCompany.
```

In this work, we have gathered thousands of posts from Twitter and stored them in a graph using a more complex version of this schema.

In order to provide a semantic representation of tweets, we have selected TwitLogic [16], which provides a vocabulary for tweets. The basic fields and their relationships are mainly RDF properties and classes taken from well-known sources like FOAF [17] or SIOC [18]. In this work we make use of FIBO to represent the entities of the financial domain. More specifically, we deal with Banks (Incorporated Companies) that have social presence and/or are mentioned by microblogging users. Marl and Onyx have been used for sentiment and emotion annotation, respectively. With this model, we can query all the opinions about a certain entity, statistics such as Positive/Negative ratio, and so on. Listing 1.3 shows an example that gets the count of positive and negative opinions about each entity.

However, the true potential of Linked Data comes into play when we use data from different sources. For instance, if there is another endpoint that contains opinions gathered from Twitter or other social networks, we can query their information seamlessly, provided they use Marl and FIBO as well.

If that example still seems uninteresting, we can also use disparate sources, such as DBpedia. DBpedia contains general information about many entities, which includes several corporations. To be able to query DBpedia, we just need to link our entities to a DBpedia entity. If we take our former example, this modification is as simple as:

Of course, this also involves named entity recognition techniques, which are covered in Section 3.2. Once this step is done, we can issue complex queries that answer questions such as: "What is the general opinion about Banks in Spain?", or "What is the relationship between year of incorporation and the number of opinions in social media?". Note that such queries could use advanced FIBO information, such as current contracts or date of incorporation.

Listing 1.2. Linking FIBO entities to DBpedia

```
ex:GSantander rdfs:seeAlso dbpedia:Santander_Group .
```

Listing 1.3. Query all positive opinions

```
PREFIX sioc: <http://rdfs.org/sioc/ns#>
PREFIX marl: <http://www.gsi.dit.upm.es/ontologies/marl/ns#>
SELECT ?entity
    COUNT(?negative_opinion) AS ?negative_opinions
    COUNT(?positive_opinion) AS ?positive_opinions
WHERE {
{
    ?positive_opinion marl:describesObject ?entity .
    ?positive_opinion marl:hasPolarity marl:Positive .
} UNION {
    ?negative_opinion marl:describesObject ?entity .
    ?negative_opinion marl:hasPolarity marl:Negative .
} } GROUP BY ?entity
```

3 Financial Twitter Tracker Architecture

In this section we describe the architecture of a prototype, called Financial Twitter Tracker, that we have developed for tracking the sentiment evolution of financial entities in Twitter. The core of the system is a semantic pipeline, described below, where tweets are retrieved and analysed. As a result, tweets are semantically annotated as stored in the semantic store Linked Media Framework (LMF) [19]. LMF also provides indexing capabilities based on Solr [20] full text indexing scalable solution. Finally a linked data visualisation framework called Sefarad[10] has been used in order to provide business analysts with a dashboard that assists them in their business decisions, as shown in Fig. 3.

The semantic pipeline for sentiment analysis consists of three tasks. First, the system connects to the Twitter API (Sect. 3.1) and retrieves tweets that match a list of predefined keywords. Then, a semantic analysis (Sect. 3.2 is carried out. Finally the sentiment analysis is done (Sect. 3.3).

3.1 Tweet retrieval

For the purpose of obtaining tweets we developed a wrapper over the services offered by the public Twitter API[11], concretely method search bounded by dates and keywords, which allows the retrieval of each and every tweet published within a particular day and regarding a particular topic. Given the data set of study, several related topics to financial world – such as banking, telecommunication, energy, to name a few – were established. Such data sets have been split according to different languages in order to increase performance and accuracy within the developed "sentiment analysis".

[10] Available at http://github.com/gsi-upm/Sefarad
[11] https://dev.twitter.com/docs/api/1.1/get/search/tweets

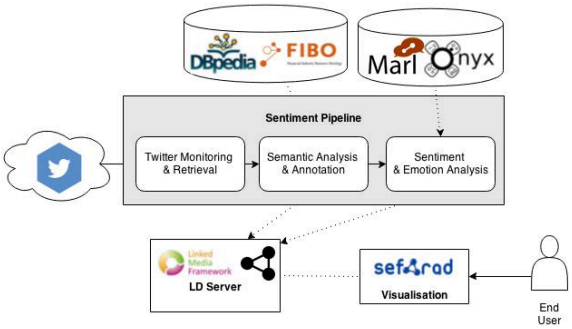


Fig. 2. Financial Twitter Tracker Architecture

3.2 Semantic Analysis and annotation

Data from Twitter is very heterogeneous, as it is used for different purposes (e.g. reviews, factual data, personal comments), covering different categories and subjects. Hence, it was necessary to carry out a categorization prior to the data analysis itself. With such filtering in mind, the Support Vector Machine system (SVM) was developed, taking into account the fact that it supports high dimensional data [21] and their suitability for classifying high volume of information using only support vectors which can be used in any distributed system [22,23,24] offering a great capability of cohesion and adaptation for the MapReduce paradigm. Several studies have proved that SVM provides better results than other techniques of classifications [25]. The system mentioned above has been trained throughout a random sampling of tweets tagged manually using Python with scikit [26] and numpy [27].

As POS-tagging, Treetagger [28] was chosen since it provides support for several languages. After acquiring a financial corpus for tracking a set of financial institutions, this corpus was cleaned, leaving aside irrelevant terms and stop words. Afterwards, collocations were extracted from the most frequent terms generating triplets with a structure domain-context-word (i.e. finance - profits - increasing). Once established these triplets, the following stage was to manually tag them by assigning a quantitative score to determine polarity and synset corresponding to WordNet 3.0 [29][10] basis. These triplets entitle the system to register texts providing scores thanks to the arrangements with WordNet, and leaning on MultiWordNet [30], WN-Affect [31], WN-Domains[32] and SentiWordNet[33]. For this goal, SentiWordNet has been extended in order to reasign scores for the finance domain. The method to enrich the lexicon stands out because its simplicity in terms of configuration, granting the chance of adding new languages easily or extending attached features (affects, domains, scores, etc.)

Another relevant aspect about the lexicon enrichment for its later storage and visualization was the extraction of entities by a NER based on Wikipedia, so that information is compared to the entities published by Wikipedia in order to work out the possible extraction from the text. Periodically the system brings the available information up to date with the new entries published on the online encyclopedia. Finally, that information is lined up with the financial ontology FIBO to provide data in a standardized way in accordance with the semantic web principles such as RDF/OWL, allowing the integration in other technical systems that adapts the given standard. Thanks to FIBO it is possible to provide a clear meaning - without ambiguities - for the financial terms.

3.3 Sentiment and Emotion Analysis

The last stage of the pipeline is in charge of the sentiment and emotion analysis. With a view to quantify the "sentiment" the procedure is to perform the arithmetic mean considering all the registered values recognized in the tweet and using simple rules like inverters (i.e. not). The emotion field can be extracted from the connection between triplets (aligned with WordNet 3.0) and WN-Affect. The outcome stems from the analysis of each tweet which was stored in a MongoDB NoSQL data base, which can handle high volume of information fulfilling the big data requirements of twitter processing.

3.4 Storage and visualisation

After the processing is done, all the triples are stored in an LMF instance, which provides SPARQL and Solr [20] endpoints. We built a generic visualisation framework, Sefarad, that uses these endpoints to display relevant information in any modern browser. This framework is modular and highly customisable. It already contains several plugins that use the power of D3 [12] to display the financial information in several ways. The plugins used, their configuration and location can be configured via an in-browser editor. One of its plugins allows the representation of public sentiment about each entity using Chernoff faces [34].

4 Experimentation

Throughout classification and Sentiment Analysis stages stages of the aforementioned pipeline, we performed experimentation with the obtained data. The classification step has been developed with an SVM trained for the recognition of two groups; finance and non-finance; which states whether the tweets are to continue to the next flow level or, on the contrary, are to be discarded.

Within Machine Learning there are two main discovery methods: supervised and unsupervised learning. In supervised learning, a series of manually tagged data are provided for the system training. On the unsupervised setting, it is the

[12] http://d3js.org/

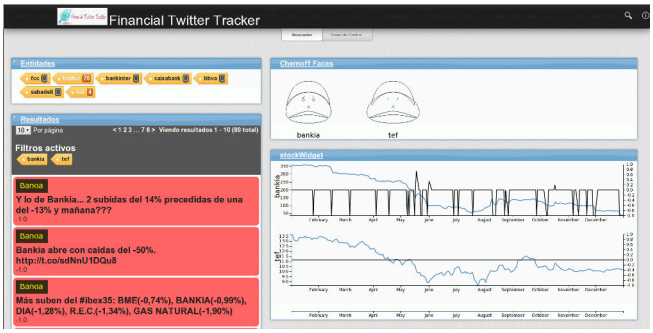


Fig. 3. Financial information displayed using Sefarad

system itself that directly infers patterns from the raw information. The current project uses supervised learning: a random set of tagged tweets has been trained by experts in finances and added to the established groups.

The model has been trained 4 times by modifying the range of information in order to measure and test the system. The first approach makes use of 90% of values for the training and 10% for the assessment, such proportions vary in the second training to 80%-20%, 70%-30% for the third, and 60%-40% for the last one [35]. Each of these cases has been tested five times in order to achieve the harmonic mean of the model accuracy with values chosen randomly for either experiment. The results of these experiments are summarised in Table 1.

Model training-Test	90%-10%	80%-20%	70%-30%	60%-40%
Average precision	0,940	0,9393	0,9369	0,9290
Supported Vectors	886,4548	825,4787	757,501	674,8602

Table 1. Results using different training options

From these results we observe that the bigger the quantity of information used to train the model, the more precise is the outcome, and that value decreases as the volume of data saved for the assessment grows. However, it is remarkable that the more data is used to train the system, the greater is the number of supporting vectors, and, consequently, the classifier loses computational performance.

The Sentiment Analysis phase has been tested against a set of manually annotated corpus. The evaluation was carried out in order to measure the effectiveness. Such accuracy conforms the ability the system owns to satisfy the feature it was developed for [36]. We have classified the results according to the parameters in Table 2. We used these values to define a set of quality metrics as shown in Table 2. The obtained results can be seen in Table 3.

System \ Expert	Identified	Not identified
Retrieved	a	b
Not retrieved	c	d

$$Recall := \frac{a}{a+c} \quad (1)$$

$$Precision := \frac{a}{a+b} \quad (2)$$

$$P_{omissions} := \frac{c}{a+c} \quad (3)$$

$$P_{falsepositive} := \frac{b}{b+d} \quad (4)$$

$$F_1 := 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (5)$$

Table 2. Parameters used in the quality metrics.

Precision	Recall	Probability omissions	Probability false positive	F-Score
84'4%	63'87%	36'12%	77'04%	0.7271

Table 3. Resulting metrics.

5 Conclusions and Future Work

In this article we have presented a vocabulary for modelling sentiments and emotions. This vocabulary can be used to query opinions and emotions about financial institutions and stock values across different web sites and information sources. The main advantage of this approach is that heterogeneous sentiment indexes can be easily integrated and used together with other vocabularies such as FIBO. We have evaluated these vocabularies in a sentiment analysis service based on Twitter for tracking financial institutions.

As a future work, we are working on improving the visualisation and query capabilities of the interface so that non technical users, such as business analysts can take advantage of the possibilities that the Web of Data brings for exploring and consulting, sentiment about financial institutions in large amounts of complex and heterogeneous data.

Another current line of research is the standardisation of these vocabularies for sentiment and emotion. With this aim, we are participating in the Linked Data Models for Emotion and Sentiment Analysis W3C Community Group, which takes as a baseline the vocabularies Marl and Onyx.

6 Acknowledgement

This research has been partially funded by the Spanish Ministry of Industry, Tourism and Trade through the project Financial Twitter Tracker (TSI-090100-2011-114) and the EUROSENTIMENT FP7 Project (Grant Agreement no: 296277)

References

1. Chatfield, A.T., Brajawidagda, U.: Twitter early tsunami warning system: A case study in indonesia's natural disaster management. In: System Sciences (HICSS), 2013 46th Hawaii International Conference on. (Jan 2013) 2050–2060
2. Yardeni, E.: Stock market indicators: Fundamental, sentiment & technical. Technical report, Yardeni Research (2014) Available at `http://www.yardeni.com/pub/peacockbullbear.pdf`.
3. Vu, T.T., Chang, S., Ha, Q.T., Collier, N.: An experiment in integrating sentiment features for tech stock prediction in twitter. In: 24th International Conference on Computational Linguistics. (2012) 23
4. Bollen, J., Mao, H., Zeng, X.: Twitter mood predicts the stock market. Journal of Computational Science **2**(1) (2011) 1–8
5. O'Riain, S., Curry, E., Harth, A.: XBRL and open data for global financial ecosystems: a linked data approach. International Journal of Accounting Information Systems **13**(2) (June 2012) 141–162
6. O'Riain, S., Harth, A., Curry, E.: Linked Data Driven Information Systems as an Enabler for Integrating Financial Data. In: Information Systems for Global Financial Markets: Emerging Developments and Effects. IGI Global (2012)
7. Krieger, H., Declerck, T., Nedunchezhian, A.: MFO - the federated financial ontology for the monnet project. In: Proceedings of the 4th International Conference on Knowledge Engineering and Ontology Development, Barcelona, Spain (2012)
8. O'Riain, S., Coughlan, B., Buitelaar, P., Declerk, T., Krieger, U., Marie-Thomas, S.: Cross-lingual querying and comparison of linked financial and business data. In: Proceedings of 10th Extended Semantic Web Conference (ESWC), Montpellier, France (2013)
9. Grcar, M., Häusser, T., Ressel, D.: D3.1 semantic resources and data acquisition. Technical report, First project (2011)
10. Klein, A., Altuntas, O., Häusser, T., Kessler, W.: Extracting investor sentiment from weblog texts: A knowledge based approach. In: Proc. of the 2011 IEEE Conference on Commerce and Enterprise Computing. (2011) 1–9
11. Goto, M., Hu, B., Naseer, A., Vandenbusshe, P.I.: Linked data for financial reporting. In: 4th International Worshop on Consuming Linked Data (COLD2013), CEUR Workshop proceedings (2013)
12. Council, E.: FIBO. Financial Industry Business Ontology. Available at http://www.edmcouncil.org/financialbusiness (June 2013)
13. IFIP: FEF. financial exchange framework ontology. Available at http://www.financial-format.com/index.html (June 2003)
14. Westerski, A., Iglesias, C.A., Tapia, F.: Linked Opinions: Describing Sentiments on the Structured Web of Data. In: Proceedings of the 4th International Workshop Social Data on the Web. (2011)
15. Sánchez-Rada, J.F., Iglesias, C.A.: Onyx: Describing Emotions on the Web of Data. In: Proceedings of the 1st International Workshop on Emotion and Sentiment in Social and Expressive Media: approaches and perspectives from AI, Torino, Italy, AI*IA, Italian Association for Artificial Intelligence (December 2013)
16. Shinavier, J.: Real-time# semanticweb in $\leq$ 140 chars. In: Proceedings of the Third Workshop on Linked Data on the Web (LDOW2010) at WWW2010. (2010)
17. Golbeck, J., Rothstein, M.: Linking social networks on the web with foaf: A semantic web case study. In: AAAI. Volume 8. (2008) 1138–1143

18. Breslin, J.G., Decker, S., Harth, A., Bojars, U.: Sioc: an approach to connect web-based communities. International Journal of Web Based Communities **2**(2) (2006) 133–142
19. Kurz, T., Schaffert, S., Burger, T.: Lmf: A framework for linked media. In: Multimedia on the Web (MMWeb), 2011 Workshop on, IEEE (2011) 16–20
20. Smiley, D., Pugh, D.E.: Apache Solr 3 Enterprise Search Serve. Packt Publishing (2011)
21. Dilrukshi, I., De Zoysa, K., Caldera, A.: Twitter news classification using svm. In: Computer Science & Education (ICCSE), 2013 8th International Conference on, IEEE (2013) 287–291
22. Jakkula, V.: Tutorial on support vector machine (svm). School of EECS, Washington State University (2006)
23. Pak, A., Paroubek, P.: Twitter as a corpus for sentiment analysis and opinion mining. In: LREC. (2010)
24. Balahur, A., Steinberger, R., Goot, E.v.d., Pouliquen, B., Kabadjov, M.: Opinion mining on newspaper quotations. In: Web Intelligence and Intelligent Agent Technologies, 2009. WI-IAT'09. IEEE/WIC/ACM International Joint Conferences on. Volume 3., IET (2009) 523–526
25. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up?: sentiment classification using machine learning techniques. In: Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10, Association for Computational Linguistics (2002) 79–86
26. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: Machine learning in Python. Journal of Machine Learning Research **12** (2011) 2825–2830
27. Oliphant, T.E.: Guide to NumPy, Provo, UT. (March 2006)
28. Schmid, H.: Probabilistic part-of-speech tagging using decision trees. In: Proceedings of international conference on new methods in language processing. Volume 12., Manchester, UK (1994) 44–49
29. Fellbaum, C.: WordNet. Wiley Online Library (1999)
30. Pianta, E., Bentivogli, L., Girardi, C.: Multiwordnet. developing an aligned multilingual database. In: Proc. 1st International Conference on Global WordNet. (2002)
31. Strapparava, C., Valitutti, A.: Wordnet affect: an affective extension of wordnet. In: LREC. Volume 4. (2004) 1083–1086
32. Magnini, B., Strapparava, C., Pezzulo, G., Gliozzo, A.: The role of domain information in word sense disambiguation. Natural Language Engineering **8**(4) (2002) 359–373
33. Baccianella, S., Esuli, A., Sebastiani, F.: Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In: LREC. Volume 10. (2010) 2200–2204
34. Chernoff, H.: The use of faces to represent points in k-dimensional space graphically. Journal of the American Statistical Association **68**(342) (1973) 361–368
35. Kohavi, R., et al.: A study of cross-validation and bootstrap for accuracy estimation and model selection. In: IJCAI. Volume 14. (1995) 1137–1145
36. Manning, C.D., Raghavan, P., Schütze, H.: Introduction to information retrieval. Volume 1. Cambridge university press Cambridge (2008)

3.1.8 Generating Linked-Data based Domain-Specific Sentiment Lexicons from Legacy Language and Semantic Resources

Title	Generating Linked-Data based Domain-Specific Sentiment Lexicons from Legacy Language and Semantic Resources
Authors	Vulcu, Gabriela and Buitelaar, Paul and Negi, Sapna and Pereira, Bianca and Arcan, Mihael and Coughlan, Barry and Sánchez-Rada, J. Fernando and Iglesias, Carlos A.
Proceedings	International Workshop on Emotion, Social Signals, Sentiment & Linked Open Data, co-located with LREC 2014,
ISBN	
Year	2014
Keywords	domain lexicon, domain lexion, sentiment analysis, Sentiment analysis
Pages	6–9
Abstract	We present a methodology for legacy language resource adaptation that generates domain-specific sentiment lexicons organized around domain entities described with lexical information and sentiment words described in the context of these entities. We explain the steps of the methodology and we give a working example of our initial results. The resulting lexicons are modelled as Linked Data resources by use of established formats for Linguistic Linked Data (lemon, NIF) and for linked sentiment expressions (Marl), thereby contributing and linking to existing Language Resources in the Linguistic Linked Open Data cloud.

Generating Linked-Data based Domain-Specific Sentiment Lexicons from Legacy Language and Semantic Resources

Gabriela Vulcu, Paul Buitelaar, Sapna Negi, Bianca Pereira, Mihael Arcan, Barry Coughlan, J. Fernando Sanchez, Carlos A. Iglesias
Insight, Centre for Data Analytics, Galway, Ireland
gabriela.vulcu@insight-center.org, paul.buitelaar@insight-center.org, sapna.negi@insight-center.org,
bianca.pereira@insight-center.org, mihael.arcan@insight-center.org, b.coughlan2@gmail.com,
Universidad Politecnica de Madrid, Spain
jfernando@gsi.dit.upm.es, cif@dit.upm.es

Abstract

We present a methodology for legacy language resource adaptation that generates domain-specific sentiment lexicons organized around domain entities described with lexical information and sentiment words described in the context of these entities. We explain the steps of the methodology and we give a working example of our initial results. The resulting lexicons are modelled as Linked Data resources by use of established formats for Linguistic Linked Data (lemon, NIF) and for linked sentiment expressions (Marl), thereby contributing and linking to existing Language Resources in the Linguistic Linked Open Data cloud.

Keywords: domain specific lexicon, entity extraction and linking, sentiment analysis

1. Introduction

In recent years, there has been a high increase in the use of commercial websites, social networks and blogs which permitted users to create a lot of content that can be reused for the sentiment analysis task. However the development of systems for sentiment analysis which exploit these valuable resources is hampered by difficulties to access the necessary language resources for several reasons: (i) language resource owners fear for losing competitiveness; (ii) lack of agreed language resource schemas for sentiment analysis and not normalised magnitudes for measuring sentiment strength; (iii) high costs for adapting existing language resources for sentiment analysis; (iv) reduced visibility, accessibility and interoperability of the language resources with other language or semantic resources like the Linguistic Linked Open Data cloud (i.e. LLOD). In this paper we are focusing on the second and the forth challenges by describing a methodology for the conversion, enhancement and integration of a wide range of legacy language and semantic resources into a common format based on the lemon[1](McCrae et al., 2012) and Marl [2] (Westerski et al., 2011) Linked Data formats.

1.1. Legacy Language Resources

We identified several categories of legacy language resources with respect to our methodology: domain-specific English review corpora, non-English review corpora, sentiment annotated dictionaries and Wordnets. The existing legacy language resources (gathered in the EUROSENTIMENT project [3]) are available in many formats and they contain several types of annotations that are relevant for the sentiment analysis task. The language resources formats range from plain text with or without custom made annotations, HTML, XML, EXCEL, TSV, CSV to RDF/XML. The language resources annotations are all or a subset of: *domain* - the broad context of the review corpus (i.e. 'hotel' is the domain for the TripAdvisor corpus); *language* - the language of the language resource; *context entities* - relevant entities in the corpus; *lemma* - lemma annotations of the relevant entities; *POS* - part-of-speach annotations of the relevant entities; *WordNet synset* - annotations with existing synsets from Wordnet of the relevant entities; *sentiment* - positive or negative sentiment annotation both at sentence level and or at entity level; *emotion* - more fine grained polarity values both expressed as numbers or as concepts from well defined ontologies; *inflections* - morphosyntactic annotations of the relevant entities.

1.2. Methodology for LR Adaptation and Sentiment Lexicon Generation

Our method generates domain-specific sentiment lexicons from legacy language resources and enriching them with semantics and additional linguistic information from resources like DBpedia and BabelNet. The language resources adaptation pipeline consists of four main steps highlighted by dashed rectangles in Figure 1: (i) the Corpus Conversion step normalizes the different language resources to a common schema based on Marl and NIF[4]; (ii) the Semantic Analysis step extracts the domain-specific entity classes and named entities and identifies links between these entities and concepts from the LLOD Cloud; (iii) the Sentiment Analysis step extracts contextual sentiments and identifies SentiWordNet synsets corresponding to these contextual sentiment words; (iv) the Lexicon Generator step uses the results of the previous steps, enhances them with multilingual and morphosyntactic information and converts the results into a lexicon based on the lemon and Marl formats. Different language resources are processed with variations of the given adaptation pipeline. For example the domain-specific English review corpora are

[1] http://lemon-model.net/lexica/pwn/
[2] http://www.gi2mo.org/marl/0.1/ns.html
[3] http://eurosentiment.eu/
[4] http://persistence.uni-leipzig.org/nlp2rdf/

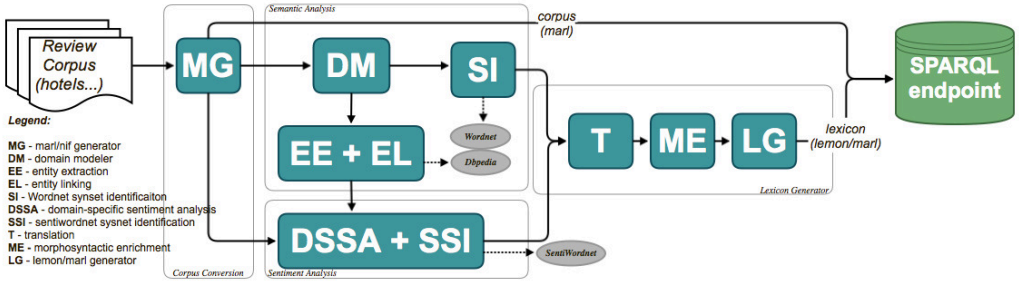


Figure 1: Methodology for Legacy Language Resources Adaptation for Sentiment Analysis.

processed using the pipeline described in Figure 1 while the sentiment annotated dictionaries are converted to the lemon/Marl format using the Lexicon Generator step. We detail these steps in the subsequent sections.

2. Corpus conversion

Due to the formats heterogeneity of the legacy language resources we need a common model that captures all the existing annotations in a structural way. The Corpus Conversion step adapts corpus resources to a common schema. We defined a schema based on the NIF and Marl formats that structures the annotations from the corpora reviews. For example each review in the corpus is an entry that can have overall sentiment annotations or annotations at the substring level. The Corpus Generator has been designed to be extensible and to separate the technical aspects from the content and formats being translated.

3. Semantic analysis

The Semantic Analysis step consists of: Domain Modeller (DM), Entity Extraction (EE), Entity Linking (EL) and Synset Identification (SI) components. The DM extracts a set of entity class using a pattern-based term extraction algorithm with a generic domain model (Bordea, 2013) on each document, aggregates the lemmatized terms and computes their ranking in the corpus(Bordea et al., 2013). The EE and EL components are based on AELA (Pereira et al., 2013) framework for Entity Linking that uses a Linked Data dataset as reference for entity mentioning identification, extraction and disambiguation. By default, DBPedia and DBPedia Lexicalization (Mendes et al., 2011) are used as reference sources but domain-specific datasets could be used as well. The SI identifies and disambiguates WordNet synsets that match with the extracted entity classes. It extends each candidate synset with their direct hyponym and hypernym synsets. Then we compute the occurrence of a given entity class in each of these bag of words. We choose the synset with the highest occurrence score for an entity class.

4. Sentiment analysis

The Sentiment Analysis step consists of: Domain-Specific Sentiment Polarity Analysis (DSSA) and Sentiment Synset Identification (SSI) components. The DSSA component identifies a set of sentiment words and their polarities in the context of the entities identified in the Semantic Analysis step. The clause in which a entity mention occurs is considered the span for a sentiment word/phrase in the context of that entity. The DSSA is based on earlier research on sentiment analysis for identifying adjectives or adjective phrases (Hu and Liu, 2004), adverbs (Benamara et al., 2007), two-word phrases (Turney and Littman, 2005) and verbs (Subrahmanian and Reforgiato, 2008). Particular attention is given to the sentiment phrases which can represent an opposite sentiment than what they represent if separated into individual words. For example, 'ridiculous bargain' represents a positive sentiment while 'ridiculous' could represent a negative sentiment. Sentiment words/phrases in individual reviews are assigned polarity scores based on the available user ratings. In case of language resources with no ratings we use a bootstrapping process based on Sentiwordnet that will rate the domain aspects in the review. We select the most frequent scores as the final sentiment score for a sentiment word/phrase candidate based on its occurrences in all the reviews. The SSI component identifies SentiWordNet synsets for the extracted contextual sentiment words. The sentiment phrases however, are not assigned any synset. Linking the sentiment words with those of SentiWordNet further enhances their semantic information. We identify the nearest SentiWordNet sense for a sentiment candidate using Concept-Based Disambiguation (Raviv and Markovitch, 2012) which utilizes the semantic similarity measure 'Explicit Semantic Analysis' (Gabrilovich and Markovitch, 2006) to represent senses in a high-dimensional space of natural concepts. Concepts are obtained from large knowledge resources such as Wikipedia, which also covers domain specific knowledge. We compare the semantic similarity scores obtained by computing semantic similarity of a bag of words containing domain name, entity and sentiment word with bags of words which contain members of the synset and the gloss for each synset of that SentiWordNet entry. The synset with the highest similarity score above a threshold is considered.

5. Lexicon generator

The Lexicon Generator step consists of: MorphoSyntactic Enrichment (ME), Machine Translation(T) and lemon/Marl Generator(LG) components. As WordNet does not provide

Sentiment	PolarityValue	Context
”good”@en	”1.0”	”alarm”@en
”damaged”@en	”-2.0”	”apple”@en
”amazed”@en	”2.0”	”flash”@en
”expensive”@en	”-1.0”	”flash”@en
”annoying”@en	”-1.5”	”player”@en

Table 1: Sentiment words the ’electronics’ domain.

Sentiment	PolarityValue	Context
”warm”@en	”2.0”	”pastries”@en
”warm”@en	”2.0”	”comfort”@en
”warm”@en	”1.80”	”restaurant”@en
”warm”@en	”1.73”	”service”@en
”warm”@en	”0.98”	”hotel”@en

Table 2: Sentiment word ’warm’ in the ’hotel’ domain.

any morphosyntactic information (besides part of speech), such as inflection and morphological or syntactic decomposition, the ME provides a further process for the conversion and integration of lexical information for selected synsets from other legacy language resources like CELEX [5]. Next, the T component translates extracted entity classes and sentiment words in other languages using a domain-adaptive machine translation approach (Arcan et al., 2013). This way we can build sentiment lexicons in other languages. It uses the SMT toolkit Moses (Koehn et al., 2007). Word alignments are built with the GIZA++ toolkit (Och and Ney, 2003), where a 5-gram language model was built by SRILM with Kneser-Ney smoothing (Stolcke, 2002). We use two different parallel resources: the JRC-Acquis (Steinberger et al., 2006) available in almost every EU official language (except Irish) and the OpenSubtitles2013 (Tiedemann, 2012) which contains fan-subtitled text for the most popular language pairs. The LG component converts the results of the previous components (named entities and entity classes linked to LOD and sentiment words with polarity values) to a domain-specific sentiment lexicon represented as RDF in the lemon/Marl format. The lemon model was developed in the Monnet project to be a standard for sharing lexical information on the semantic web. The model draws heavily from earlier work, in particular from LexInfo (Cimiano et al., 2011), LIR (Montiel-Ponsoda et al., 2008) and LMF (Francopoulo et al., 2006). The Marl model is a standardised data schema designed to annotate and describe subjective opinions.

6. Working Example

Figure 2 shows an example of a generated lexicon for the domain ’hotel’ in English. It shows 3 *lemon:LexicalEntries*: ’room ’ (entity class), ’Paris’ (named entity) and ’small’ (sentiment word) which in the context of the lexical entry ’room’ has negative polarity. Each of them consists of senses, which are linked to DBpedia and/or Wordnet concepts.

We applied our methodology on an annotated corpus of 10.000 reviews for the hotel domain and an annotated corpus of 600 reviews for the electronics domain. Table 1 shows an example of sentiment words from the ’electronics’ domain, while Table 2 shows an example of different contexts of the sentiment word ’warm’ with their corresponding polarities in the ’hotel’ domain.

7. Future Work

We are currently working on evaluating the Semantic Analysis and Sentiment Analysis components by participating in the SemEval challenge [6] on aspect-based sentiment analysis. We also plan to investigate ways of linking the extracted named entities with other Linked Data datasets like Yago or Freebase. A next step for the use of our results is to aggregate sentiment lexicons obtained from Language Resources on the same domain.

8. Conclusions

In this paper we presented a methodology for creating domain-specific sentiment lexicons from legacy Language Resources, described the components of our methodology and provided example results.

9. Acknowledgements

This work has been funded by the European project EUROSENTIMENT under grant no. 296277.

10. References

Arcan, M., Thomas, S. M., Brandt, D. D., and Buitelaar, P. (2013). Translating the FINREP taxonomy using a domain-specific corpus. Poster presented at the Machine Translation Summit XIV, Nice, France.

Benamara, F., Cesarano, C., Picariello, A., Reforgiato, D., and Subrahmanian, V. S. (2007). Sentiment analysis: Adjectives and adverbs are better than adjectives alone. In *Proceedings of the International Conference on Weblogs and Social Media*, ICWSM’07.

Bordea, G., Buitelaar, P., and Polajnar, T. (2013). Domain-independent term extraction through domain modelling. In *Proceedings of the 10th International Conference on Terminology and Artificial Intelligence*, TIA’13, Paris, France.

Bordea, G. (2013). *Domain Adaptive Extraction of Topical Hierarchies for Expertise Mining*. Ph.D. thesis, National University of Ireland, Galway.

Cimiano, P., Buitelaar, P., McCrae, J., and Sintek, M. (2011). Lexinfo: A declarative model for the lexicon-ontology interface. *Web Semantics: Science, Services and Agents on the World Wide Web*.

Francopoulo, G., Bel, N., George, M., Calzolari, N., Monachini, M., Pet, M., and Soria, C. (2006). Lexical markup framework (LMF) for NLP multilingual resources. In *Proceedings of the Workshop on Multilingual Language Resources and Interoperability*, Sydney, Australia. ACL.

Gabrilovich, E. and Markovitch, S. (2006). Overcoming the brittleness bottleneck using wikipedia: Enhancing text categorization with encyclopedic knowledge. In *Proceedings of the 21st National Conference on Artificial Intelligence*, AAAI’06. AAAI Press.

[5] http://celex.mpi.nl/

[6] http://alt.qcri.org/semeval2014/

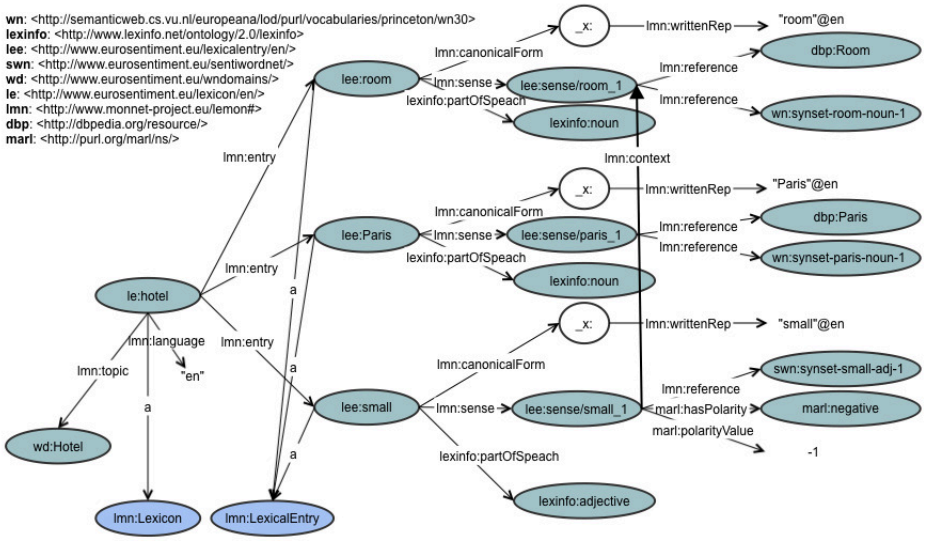


Figure 2: Example lexicon for the domain 'hotel' in English.

Hu, M. and Liu, B. (2004). Mining and summarizing customer reviews. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, New York, NY, USA. ACM.

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A., and Herbst, E. (2007). Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ACL '07, Stroudsburg, PA, USA. ACL.

McCrae, J., de Cea, G. A., Buitelaar, P., Cimiano, P., Declerck, T., Gómez-Pérez, A., Gracia, J., Hollink, L., Montiel-Ponsoda, E., Spohr, D., and Wunner, T. (2012). Interchanging lexical resources on the semantic web. *Language Resources and Evaluation*.

Mendes, P. N., Jakob, M., García-Silva, A., and Bizer, C. (2011). Dbpedia spotlight: Shedding light on the web of documents. In *Proceedings of the 7th International Conference on Semantic Systems*, I-Semantics '11, New York, NY, USA. ACM.

Montiel-Ponsoda, E., de Cea, G. A., Gómez-Pérez, A., and Peters, W. (2008). Modelling multilinguality in ontologies. In *Poster at COLING'10*.

Och, F. J. and Ney, H. (2003). A systematic comparison of various statistical alignment models. *Computational Linguistics*, March.

Pereira, B., Aggarwal, N., and Buitelaar, P. (2013). Aela: An adaptive entity linking approach. In *Proceedings of the 22nd International Conference on World Wide Web Companion*, WWW'13, Republic and Canton of Geneva, Switzerland.

Raviv, A. and Markovitch, S. (2012). Concept-based approach to word-sense disambiguation. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence*.

Steinberger, R., Pouliquen, B., Widiger, A., Ignat, C., Erjavec, T., and Tufis, D. (2006). The jrc-acquis: A multilingual aligned parallel corpus with 20+ languages. In *Proceedings of the 5th International Conference on Language Resources and Evaluation*, LREC'06.

Stolcke, A. (2002). Srilm - an extensible language modeling toolkit. In *Proceedings International Conference on Spoken Language Processing (ICSLP 2002*.

Subrahmanian, V. and Reforgiato, D. (2008). Ava: Adjective-verb-adverb combinations for sentiment analysis. *Intelligent Systems*.

Tiedemann, J. (2012). Parallel data, tools and interfaces in opus. In *Proceedings of the 8th International Conference on Language Resources and Evaluation*, LREC'12, Istanbul, Turkey. ELRA.

Turney, P. D. and Littman, M. L. (2005). Corpus-based learning of analogies and semantic relations. *Machine Learning*.

Westerski, A., Iglesias, C. A., and Tapia, F. (2011). Linked Opinions: Describing Sentiments on the Structured Web of Data. In *Proceedings of the 4th International Workshop Social Data on the Web*.

3.2 Linked Data tools and services for sentiment analysis

3.2.1 Senpy: A framework for semantic sentiment and emotion analysis services

Title	Senpy: A framework for semantic sentiment and emotion analysis services
Authors	Sánchez-Rada, J. Fernando and Araque, Oscar and Iglesias, Carlos A.
Journal	Knowledge-Based Systems
Impact factor	JCR 2018 Q1 (5.101)
ISSN	
Publisher	
Year	2019
Keywords	emotion analysis, evaluation, linked data, Sentiment analysis
Pages	
Online	https://www.sciencedirect.com/science/article/pii/S0950705119305313
Abstract	Senpy is a framework to develop, evaluate and publish web services for sentiment and emotion analysis in text. The framework is aimed towards both developers and users. For developers, it is a means to evaluate their classifiers and easily publish them as web services. For users, it is a way to consume sentiment analysis from different providers through the same interface. This is achieved through a combination of an API aligned with the NLP Interchange Format (NIF) service specification, the use of semantic formats and a series of well established vocabularies. The framework is Open Source, and has been used extensively in several projects. As a result, several Senpy Open Source services are available for use and download.

Senpy: a Framework for Semantic Sentiment and Emotion Analysis Services

J. Fernando Sánchez-Rada*, Oscar Araque, Carlos A. Iglesias

Grupo de Sistemas Inteligentes, Universidad Politécnica de Madrid, Spain

Abstract

Senpy is a framework to develop, evaluate and publish web services for sentiment and emotion analysis in text. The framework is aimed towards both developers and users. For developers, it is a means to evaluate their classifiers and easily publish them as web services. For users, it is a way to consume sentiment analysis from different providers through the same interface. This is achieved through a combination of an API aligned with the NLP Interchange Format (NIF) service specification, the use of semantic formats and a series of well established vocabularies. The framework is Open Source, and has been used extensively in several projects. As a result, several Senpy Open Source services are available for use and download.

Keywords: Sentiment Analysis, Emotion Analysis, Linked Data, Classification, Evaluation

1. Introduction

Sentiment analysis is a field of research and application in full expansion, driven by the popularity of social media and the need to give meaning to the concept of collective opinions [1]. A large number of sentiment analysis tools and services have been developed in recent years. Unfortunately, the lack of standard tools and APIs makes development costly, and it makes it harder to use or evaluate several services.

*I am the corresponding author

Email addresses: jf.sanchez@upm.es (J. Fernando Sánchez-Rada), o.araque@upm.es (Oscar Araque), carlosangel.iglesias@upm.es (Carlos A. Iglesias)

Senpy is an opinionated framework to publish sentiment and emotion analysis algorithms as web services. Its ultimate aim is to facilitate the growth of sentiment analysis by providing services that can be interchanged and easily evaluated. This will contribute to the quality and quantity of sentiment analysis services available to the research community.

2. Problems and Background

Senpy addresses three challenges of sentiment analysis services: model heterogeneity, interoperability of APIs and formats, and service evaluation.

There is a wide range of models for sentiment and emotion. A sentiment model may include several classes or levels (e.g. positive, negative, and neutral), and a polarity value within a given range (e.g., from -1 to +1). Emotion models are more complicated, and they are usually grouped into categorical models (e.g., Ekman's model uses 6 categories of emotion), and dimensional models (e.g., the VAD model represents emotions as vectors in a 3-dimensional space). Model heterogeneity may cause ambiguity and confusion, unless the model used is explicit and available to every user. For that reason, the Emotion Markup Language (EmotionML) [2] provides the means to link annotations to a model that can be defined and re-used.

A second issue is that most analysis tools and services use ad-hoc APIs, representation formats (e.g., JSON and XML) and schemas. This is a burden for consumers, who need to study and adapt to each service. It also hinders research, as it makes it harder to compare different approaches and solutions.

Lastly, evaluating services is crucial for two main reasons: 1) to compare different approaches in the state of the art; and 2) to estimate the performance of a given service in a specific domain (e.g., evaluating over a dataset of posts from a niche social network in the electronics domain). The lack of evaluation APIs or options to classify a dataset in bulk means that researchers need to learn to manually consume a service before they know it is appropriate for their use case.

Earlier works propose a linked data approach to solve these aspects [3, 4], but the concepts and tooling behind ontologies and linked data publishing are unfamiliar to both the linguistic community and developers. Senpy aims to bridge this gap by presenting a tool that creates full-fledged services that benefit from the power of semantic technologies without requiring any prior knowledge of web or semantic technologies.

3. Senpy Framework

3.1. Architecture

At a high level, Senpy consists of three main modules: 1) a core that provides the HTTP service layer to consume and evaluate services; 2) a module to develop plug-ins that provide specific features (e.g., a sentiment analysis algorithm),; and 3) a Web User Interface (Web UI) that can be used to consume and evaluate services in a user-friendly way, instead of directly interacting with the HTTP layer.

A plugin may be defined in a python module or a YAML file. When the Senpy tool is launched, it detects all plugins and plugin code available to it (i.e., working directory and configurable folders), and launches the Web UI and the HTTP server.

3.2. Functionalities

The main features of Senpy are:

- Development of sentiment and emotion classifiers that can be exposed as semantic multi-format HTTP services.
- Sentiment and emotion analysis from different providers (i.e., a service that uses Vader or one that uses SenticNet [5]) using the same interface (including a NIF-based [6] API and vocabularies [7]). In this way, applications do not depend on the API offered for these services. The API and vocabularies follow the draft guidelines for developing semantic services by the The Linked Data Models for Emotion and Sentiment Analysis W3 Community Group [1].
- Combination of services even if they use different models for sentiment (e.g., converting polarity levels from a "-1 to 1" interval to a "0 to 10" interval) or emotion (e.g., Ekman or VAD). Within a server, users can create processing pipelines that combine several analyses within a single request. For instance, asking for sentiment and emotion analysis in the same request, or for two complementary sentiment analyses.
- Evaluation of sentiment algorithms with well-known datasets. The included evaluation API allows users to compare specific combinations of services and datasets.

[1] `https://www.w3.org/community/sentiment/`

- A Web User-Interface for users of any skill level to consume and evaluate services. The interface visualizes the results in a user-friendly way, including tabular and multi-format visualization.

4. Implementation

The framework is implemented with Python 3.5 using open source libraries. It can be installed manually or from pip/PyPI. The HTTP layer is built on Flask, and its Web UI uses bootstrap and other Javascript libraries. RDFLib is used to convert to and from semantic formats (XML-RDF, JSON-LD, and Turtle). Scikit-Learn provides machine learning primitives.

A docker image is also provided, which contains all necessary dependencies and can be used as a standalone server or to develop new services. Moreover, the main repository contains the configuration necessary to deploy services to a Kubernetes cluster.

5. Illustrative Examples

Users may explore the main features of Senpy on the online demo[2]. It includes several Open Source services[3] for sentiment and emotion analysis. The project's documentation[4] contains a list of features as well as instructions for users and developers. It explains different ways in which these and services can be consumed, and how to evaluate their performance in a series of public datasets (Figure 1). It also covers how to develop new sentiment analysis services.

6. Conclusions

The framework presented in this paper has the potential to ease adoption, development, integration and evaluation of sentiment and emotion analysis services.

Acknowledgements

This work has been partially funded by the SEMOLA project, funded by the Ministry of Economy and Competitiveness of Spain (TEC2015-68284-R).

[2]http://senpy.gsi.upm.es/

[3]http://github.com/gsi-upm/senpy-plugins-community

[4]https://senpy.readthedocs.io

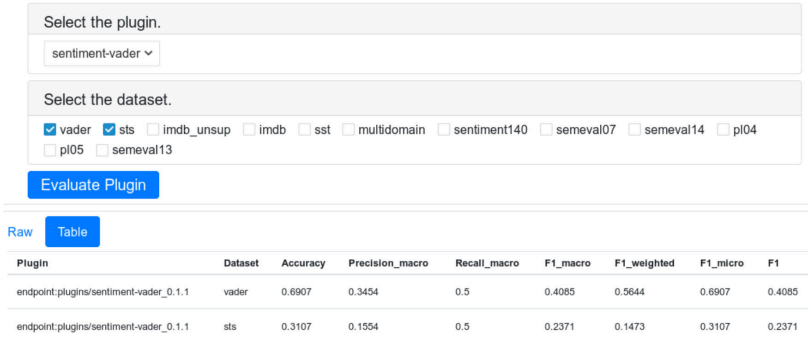


Figure 1: Evaluating a service in two datasets through the Web UI.

References

[1] B. Liu, Sentiment analysis and opinion mining, Synthesis Lectures on Human Language Technologies 5 (1) (2012) 1–167.

[2] M. Schröder, et al., EmotionML – an upcoming standard for representing emotions and related states, in: Affective Computing and Intelligent Interaction, Vol. 6974, Springer Berlin Heidelberg, 2011, pp. 316–325.

[3] S. Hellmann, J. Lehmann, S. Auer, M. Brümmer, Integrating NLP using linked data, in: The Semantic Web–ISWC, Springer, 2013, pp. 98–113.

[4] J. F. Sánchez-Rada, et al., Towards a Common Linked Data Model for Sentiment and Emotion Analysis, in: Proceedings of the LREC 2016 Workshop Emotion and Sentiment Analysis, 2016, pp. 48–54.

[5] E. Cambria, R. Speer, C. Havasi, A. Hussain, Senticnet: A publicly available semantic resource for opinion mining, in: AAAI Fall Symposium Series, 2010.

[6] S. Hellmann, J. Lehmann, S. Auer, M. Nitzschke, NIF combinator: Combining nlp tool output, in: Knowledge Engineering and Knowledge Management, Springer, 2012, pp. 446–449.

[7] J. F. Sánchez-Rada, C. A. Iglesias, Onyx: A linked data approach to emotion representation, Inf. Proc. & Manag. 52 (1) (2016) 99–114.

Required Metadata

Current executable software version

Nr.	Software metadata description	Please fill in this column
S1	Current software version	1.0.1
S2	Permanent link to executables of this version	https://github.com/gsi-upm/senpy/tree/1.0.1
S3	Legal Software License	Apache License, Version 2.0
S4	Computing platform/Operating System	Linux, OS X, Microsoft Windows, Unix-like
S5	Installation requirements & dependencies	Python 3.5+,NetworkX, NLTK, RDFLib, Scipy, Scikit-learn, Numpy
S6	Link to user manual	https://senpy.readthedocs.io/
S7	Support email for questions	jf.sanchez@upm.es

Table 1: Software metadata

Current code version

Nr.	Code metadata description	Please fill in this column
C1	Current code version	v1.0.1
C2	Permanent link to code/repository used of this code version	https://github.com/gsi-upm/senpy/tree/1.0.1
C3	Legal Code License	Apache License, Version 2.0
C4	Code versioning system used	git
C5	Software code languages, tools, and services used	Python
C6	Compilation requirements, operating environments & dependencies	Python 3.5+, NetworkX, NLTK, RDFLib, Scipy, Scikit-learn, Numpy
C7	Link to developer documentation/-manual	https://senpy.readthedocs.io/
C8	Support email for questions	jf.sanchez@upm.es

Table 2: Code metadata

3.2.2 Multimodal Multimodel Emotion Analysis as Linked Data

Title	Multimodal Multimodel Emotion Analysis as Linked Data
Authors	Sánchez-Rada, J. Fernando and Iglesias, Carlos A. and Sagha, Hesam and Schuller, Björn and Wood, Ian and Buitelaar, Paul
Proceedings	Proceedings of ACII 2017
ISBN	
Year	2017
Keywords	emotion analysis, linked data, social networks
Pages	
Abstract	The lack of a standard emotion representation model hinders emotion analysis due to the incompatibility of annota-tion formats and models from different sources, tools and an- notation services. This is also a limiting factor for multimodal analysis, since recognition services from different modalities (audio, video, text) tend to have different representation models (e. g., continuous vs. discrete emotions). This work presents a multi-disciplinary effort to alleviate this problem by formalizing conversion between emotion models. The specific contributions are: i) a semantic representation of emotion conversion; ii) an API proposal for services that perform automatic conversion; iii) a reference implementation of such a service; and iv) validation of the proposal through use cases that integrate different emotion models and service providers.

2017 Seventh International Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW)

Multimodal Multimodel Emotion Analysis as Linked Data

J. Fernando Sánchez-Rada, Carlos A. Iglesias
GSI Universidad Politécnica de Madrid
Hesam Sagha, Björn Schuller
University of Passau
Ian Wood, Paul Buitelaar
National University of Ireland Galway

***Abstract*—The lack of a standard emotion representation model hinders emotion analysis due to the incompatibility of annotation formats and models from different sources, tools and annotation services. This is also a limiting factor for multimodal analysis, since recognition services from different modalities (audio, video, text) tend to have different representation models (e. g., continuous vs. discrete emotions).**

This work presents a multi-disciplinary effort to alleviate this problem by formalizing conversion between emotion models. The specific contributions are: i) a semantic representation of emotion conversion; ii) an API proposal for services that perform automatic conversion; iii) a reference implementation of such a service; and iv) validation of the proposal through use cases that integrate different emotion models and service providers.

1. Introduction

Emotions permeate every aspect of our lives, from our facial expressions to our comments on social media. However, there is no consensus on the representation of human emotions. So far there has been little attention to unifying measurements, categories, and emotion codes. This is partly due to the field of 'affective computing' being relatively young. As a result, there is a plethora of rivaling emotion representation models with varying degrees of popularity, from categorical models such as Ekman's to Scherer's process model [1].

The lack of a standard is a hindrance when working with different sources, such as datasets annotated by different experts, due to additional effort that has to be spent in understanding the definitions of emotion in every source. It also limits the amount of annotated data for training. In some cases, for the sake of interoperability, a single representation model is chosen on a per-project basis. Then the use of other models is restricted -limiting the resources and quality-, or an ad-hoc conversion mechanism is used, which is costly and inaccurate. However, this compromise is not possible in all cases.

Initiatives such as Emotion Markup Language (EmotionML) [2] and the Onyx Emotion Ontology [3] account for the heterogeneity of models and provide vocabularies or meta-models that enable interoperability. When using these meta-models, annotations do not refer to ambiguous terms (such as *anger*) but to specific definitions (e. g., *Ekman's definition of anger*). This has two consequences: the choice of models is explicit in the annotation itself, and different models may be used in the same data set. As a result, annotations unambiguously refer to the model being used.

However, to the best of our knowledge, none of the meta-models addresses the combination of annotations using different models in a meaningful way. Hence, these annotations are still independent. This work aims to remedy this by formalizing the conversion between emotion models.

The rest of the paper is structured as follows. Section 2 introduces enabling technologies and related research in multimodal and multimodel representation; Section 3 details the proposal for semantic representation of emotion conversion; Section 4 presents our evaluation by means of a reference implementation and a use case; lastly, Section 5 summarizes our conclusions and future work.

2. Background

This section focuses on two aspects: the definition and quantification of emotions (emotion models) and how this information is encoded (representation formats). Previous works have discussed the difference between emotions and related terms (e. g., 'feelings', 'affects', 'sentiment') in detail [4], [5], [6], [7].

2.1. Models for emotions and emotion analysis

There are several models for emotions, ranging from the most simplistic and ancient that come from Chinese philosophers to the most modern theories that refine and expand older models [8], [9]. The literature on the topic is vast, and it is out of the scope of this paper to reproduce it. For the purpose of this paper, it is important to know that emotion models vary in the characteristics of the emotion they represent, and the way in which these characteristics are represented. The main two groups would be: discrete and dimensional models. In discrete models, emotions belong to one of a predefined set of categories, which varies from

978-1-5386-0680-3/17/$31.00 ©2017 IEEE

IEEE computer society

model to model. In dimensional models, an emotion is represented by the value in different axes or dimensions. A third category, mixed models, merges both views. The recent work by Cambria et al. [10] contains a comprehensive state of the art on the topic, as well as an introduction to a novel model, *the Hourglass of emotions*, inspired by Plutchik's studies [11]. Plutchik's model is a model of categories that has been extensively used [12], [13] in the area of emotion analysis and affective computing, relating all the different emotions to each other in what is called the wheel of emotions.

A more recent development in emotion representation designed as a principled annotation scheme is the Geneva Emotion Wheel [14]. This scheme combines 20 emotion labels arranged as a circle in a 2-dimensional Valence/Power space with four levels of intensity represented by distance from the centre.

Other models cover affects in general, which include emotions as part of them. One of them is the work done by Strapparava and Valitutti in WordNet-Affect [15]. It comprises more than 300 affects linked by concept-superconcept relationships, many of which are considered emotions. What makes this categorization interesting is that it effectively provides a taxonomy of emotions. It provides information about relationships between emotions and makes it possible to make choices on the level of granularity of the emotion model.

Despite all efforts, there is no universally accepted model for emotions [7], [16]. This complicates the task of representing emotions. In a discussion regarding EmotionML, Schröder et al. pose that *given the fact that even emotion theorists have very diverse definitions of what an emotion is, and that very different representations have been proposed in different research strand, any attempt to propose a standard way of representing emotions for technological contexts seems doomed to fail* [17]. Instead they claim that the markup should offer users a choice of representation, including the option to specify the affective state that is being labeled, different emotional dimensions and appraisal scales.

EmotionML [2] is one of the most notable general-purpose emotion annotation and representation languages. It was born from the efforts made for Emotion Annotation and Representation Language (EARL) [16], [18]. EARL originally included 48 emotions divided into 10 different categories. EmotionML offers twelve vocabularies for categories, appraisals, dimensions and action tendencies. A vocabulary is a set of possible values for any given attribute of the emotion. A complete description of those vocabularies and its computer-readable form is available in [19].

2.2. Multimodal Linked Data approaches

Recent work has expanded traditional annotation, such as that of EmotionML, by adding semantics and following Linked Data principles [20]. This shift has several important implications. First and foremost, it fosters the integration of different data sources. Whereas traditional annotations are usually tied to a document, this new type of annotation is meant to be queried, consumed and integrated with other sources. As a consequence, it also entails the formal definition of vocabularies and ontologies, which serve as a common representation for all sources.

The most common linked data model for emotion representation is a combination of several existing vocabularies: Onyx [3], a vocabulary to annotate and describe emotions which provides interoperability with EmotionML [21]; and NLP Interchange Format (NIF) 2.0 [22], which defines a semantic format and API for improving interoperability among natural language processing services.

Another important contribution laid the foundation to multimodal annotation [23] by adding a multimedia extension to the NIF model. The bulk of this extension (MESA) is the addition of a URI scheme for multimedia contexts, which complement the original NIF string context. This scheme follows the media fragments recommendation [24] to provide URIs for multimedia segments.

3. Proposal

Our proposal for representation of multimodal multimodel emotion analysis consists of two parts. The main one is the definition of a semantic vocabulary for annotation and conversion of annotations in different models and modalities. The second part is an API for emotion analysis services and tools that leverages the vocabulary.

We selected six basic aspects that a potential vocabulary needs to cover in order to be complete: 1) the definition of emotion annotations; 2) the definition of each emotion model (e. g., Ekman's categories); 3) multimodality; 4) the definition of the process of annotation with emotion (e. g., manual or automatic annotation) and the link between this process and the emotion annotations it generates (provenance); 5) the definition of conversion process between annotation models in a way that is compatible with (4); and 6) integration with RDF and linked data.

We reviewed existing publicly available vocabularies, looking for these criteria (Table 1). None of the vocabularies reviewed included the concept of emotion conversion. However, all the candidates can be extended, either via an XML schema or a semantic extension. Consequently, conversion could be integrated as a small extension of already existing vocabularies. To cover the rest of the criteria, there were two clear alternatives. The first one was to combine several XML schemas: EmotionML (emotions and models), EMMA [25] (multimodality) and Provenance Ontology (PROV-O) [26](provenance). The main advantage of this option is adopting EmotionML, which is well known and already integrated in several tools. Additionally, EMMA is both a W3C recommendation and the encouraged way to integrate multimodality into EmotionML annotations. Unfortunately, although EmotionML and EMMA are semantic in nature, this approach does not meet the linked data requirements, with the exception of Prov-O. However, a subset of EmotionML has already been included as Onyx sub-vocabularies, and Onyx has been successfully used as an

Table 1. COMPARISON OF DIFFERENT VOCABULARIES AND THE PROPOSAL IN THIS PAPER.

	emotions	multimodel	multimodal	provenance	conversion	semantic
Emotion-ML	✓	✓				
EMMA			✓			
PROV-O				✓		✓
Onyx	✓	✓		✓		✓
Onyx+MESA	✓	✓	✓	✓		✓
Emotion-ML + EMMA + PROV-O	✓	✓	✓	✓		
Proposed vocabulary	✓	✓	✓	✓	✓	✓

alternative to EmotionML in several projects where linked data was a strong constraint. This reason led us to the second alternative: to extend the Onyx and MESA vocabularies to include the notion of emotion conversion. We will briefly cover how NIF, Onyx, MESA and Prov-O can be used together to cover our first criteria before introducing the extension.

First of all, emotion models are represented with Onyx *EmotionModel*. All public vocabularies in the EmotionML vocabularies specification have their counterpart in the Onyx EmotionML vocabularies extension. If none of those models cover a specific case, it is also possible to define custom models and to publish them as linked data. The emotion analysis task is encoded by Onyx's *EmotionAnalysis*, a subclass of Prov-O *Activity*. As such, it can provide provenance information. It should also specify the specific model it uses for annotation. Each piece of text to be analyzed is represented as a NIF context. One of the main advantages of NIF is that it defines URI schemes for contexts which only depend on the content itself and its source. Using these unique identifiers, it is possible to aggregate annotations added by independent analysis to the same source. In order to preserve this property, provenance is stored at the annotation level. Hence, contexts are annotated with one or several Onyx *EmotionSet* entities. These *EmotionSet* entities contain all the emotion information, as well as a link to the analysis activity that generated the annotation.

To achieve multimodality in a NIF-compatible manner, MESA includes a specific NIF URI scheme to annotate string contexts within multimedia, based on the media fragments recommendation. An example is shown in Listing 1.

Listing 1. ANNOTATING STRINGS IN MULTIMEDIA.

```
<http://video.com/example#t=0,11>
   a mesa:MediaFragmentsString ;
```

To cover all the criteria, the existing models need to be extended to include the concept of emotion conversion. Figure 1 illustrates how this extension integrates with the existing entities in Onyx and Prov-O. First of all, the extension provides a new class, *Conversion*, which subclasses

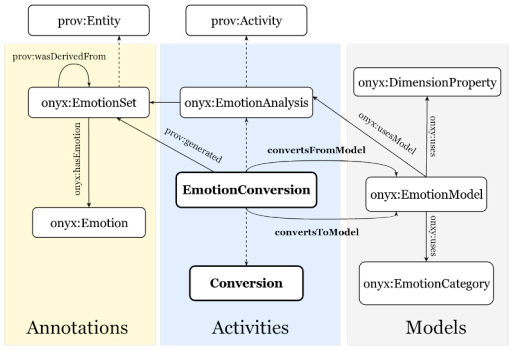


Figure 1. Extension of the Onyx ontology for emotion conversion. The extension itself shown in bold and without a prefix.

Activity. By making this class independent of emotions, it can be reused for other types of affect conversion. For instance, it could be used to represent the conversion of star-based opinion models to polarity based (thumbs up/down) models. The vocabulary also provides a more specific *EmotionConversion* activity, which subclasses both *Conversion* and *EmotionAnalysis*. The main specific properties of this class are *convertsFrom* and *convertsTo*, both of which point to an *EmotionModel* instance.

Using this extension, we can encode the conversion of the previous example from using Ekman's categorical model to PAD dimensions. An excerpt of this representation is shown in Listing 2.

Listing 2. ONYX EXTENSION TO COVER CONVERSION.

```
:Big6_to_PAD rdf:type onyx:EmotionConversion ;
             onyx:convertsFrom emoml:big6 ;
             onyx:convertsTo emoml:pad .

<http://microblog.com/User1/Post1#char=0,11>
  onyx:hasEmotionSet :EmotionSet1 ,
                     :EmotionSet2 .

:EmotionSet2
  rdf:type onyx:EmotionSet;
  onyx:hasEmotion [
    rdf:type onyx:Emotion ;
    emoml:pad_potency 0.8 ;
    emoml:pad_arousal 0.5 ;
    emoml:pad_dominance 0.3 ;
  ] ;
  prov:wasGeneratedBy :Big6_to_PAD .
```

The second part of the proposal is the web API that allows emotion analysis services to integrate emotion conversion. In particular, a service has to: advertise what models they use to annotate; advertise the conversions available; allow users to request a specific model; in case of not being able to convert from the original model to the one requested by the user, raise an error. The complete API for services is included in Table 2.

The proposed model and API have been integrated in

Senpy, a semantic framework to build sentiment and emotion analysis services [27]. This allows Senpy emotion analysis services to offer automatic emotion model conversion. The framework currently includes a very generic implementation of a centroid-based conversion, inspired by Kim et al [28]. In this centroid-based conversion, each category or label is mapped to a centroid, a point in an N-dimensional space (e.g. VAD). This provides bi-directional conversion. Given an emotion with one or more categories and an optional intensity for each category, the algorithm returns a new emotion whose VAD values are the weighted average of the values of the centroids corresponding to the categories. Given dimensional value, the conversion consists in calculating the distance of the value to each centroid, and either returning a new Emotion with a single category (the closest centroid), or one emotion per centroid and an intensity proportional to the normalized distance between the centroid and the value. This algorithm can be applied with different sets of centroids. We provide centroids for a conversion from Ekman categories to VAD values. To calculate them, we averaged the VAD values of the words in ANEW [29] that were also present in WordNet-Affect [15] under a label that can be mapped to one of Ekman's categories. All the code is Open Source and available on the framework website.

4. Evaluation

This section presents a real scenario where the proposed conversion model and service API have been used. It serves as a starting point to assess the usefulness and adequacy of the proposal. It is also a way to identify possible deficiencies and to encourage further discussion on the topic.

Table 2. THE EXTENDED EMOTION ANALYSIS SERVICE API INCLUDES PARAMETERS TO CONTROL EMOTION CONVERSION.

parameter	description
input(i)	serialized data (i.e. the text or other formats, depends on informat)
informat (f)	format in which the input is provided: turtle, text (default) or json-ld
outformat (o)	format in which the output is serialized: turtle (default), text or json-ld
prefix (p)	prefix used to create and parse URIs
minpolarity (min)	minimum polarity value of the sentiment analysis
maxpolarity (max)	maximum polarity value of the sentiment analysis
language (l)	language of the sentiment or emotion analysis
domain (d)	domain of the sentiment or emotion analysis
algorithm (a)	plugin that should be used for this analysis
emotionModel (emodel, e)	emotion model in which the output is serialized (e.g. WordNet-Affect, PAD, etc.)
conversionType	type of emotion conversion. Currently accepted values: 1) full, results contain both the converted emotions and the original emotions, alongside; 2) nested, converted emotions should appear at the top level, and link to the original ones; 3) filtered, results should only contain the converted emotions.

The use case is as follows. A given video is analyzed by three different services: a video analysis that detects emotions in faces; an emotion analysis on speech; and a text emotion analysis service that annotates the transcription of the speech. Annotations are converted to a continuous space (if necessary) and the results are fused to yield the final outcome. Figure 2 shows an overview of the analysis and their relationships.

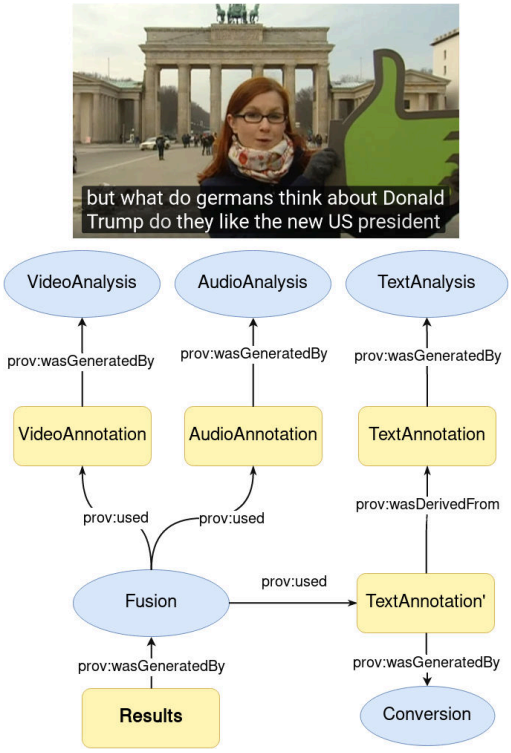


Figure 2. Generation of results combining emotion analysis in three modalities. Ellipses are provenance activities, rectangles are provenance entities.

The video emotion analyzer consists of a face detector (based on on a discriminatively trained deformable part model [30]), face tracking, and facial expression detector (based on convolutional neural networks), which recognises emotions in the continuous arousal/valence space. The speech emotion analyzer is based on openSMILE acoustic feature extractor [31], and Bag-of-Audio-Words [32], a similar concept to the Bag-of-Words of text analysis. Emotions are predicted in a continuous arousal/valence space. The text emotion analysis is performed through the *emotion-wnaffect* senpy plugin, which uses a lexicon-based approach based on WordNet-Affect [15]. It maps every affect label in the WordNet-Affect taxonomy to five of Ekman's categories:

anger, fear, disgust, joy and sadness.

The first two services, audio and video, use the PAD (Pleasure, Arousal, Dominance) model, whereas the analysis on text uses a simpler categorical analysis based on Ekman's model. In order to fuse the three annotations, the categories used in the annotation of text are converted to PAD values, using the centroid-based conversion activity which we developed as a senpy plugin. Among other things, the definition of the activity includes the algorithm being used (*senpy.plugins.conversion.centroids* at version 0.1), the values for each of the centroids, and the corresponding emotion for each of the centroids. An excerpt the definition of the conversion from Ekman dimensions to VAD dimensions is included as Listing 3.

Listing 3. DEFINITION OF THE ACTIVITY THAT CONVERTS ANNOTATIONS FROM EKMAN'S CATEGORIES TO PAD VALUES.

```
<http://servicehost/api/plugins/Ekman2PAD_0.1> a :
     emotionConversionPlugin ;
   onyx:centroids [
        onyx:hasEmotionCategory emoml:big6anger ;
        emoml:arousal 6.95e+00 ;
        emoml:dominance 5.1e+00 ;
        emoml:valence 2.7e+00 ] ;
       [
       onyx:hasEmotionCategory emoml:big6disgust ;
        emoml:arousal 5.3e+00 ;
        emoml:dominance 8.05e+00 ;
        emoml:valence 2.7e+00 ] ;
       [
       onyx:hasEmotionCategory emoml:big6fear ;
        emoml:arousal 6.5e+00 ;
        emoml:dominance 3.6e+00 ;
        emoml:valence 3.2e+00 ] ;
       [
       onyx:hasEmotionCategory emoml:big6happiness ;
        emoml:arousal 7.22e+00 ;
        emoml:dominance 6.28e+00 ;
        emoml:valence 8.6e+00 ] ;
       [
       onyx:hasEmotionCategory emoml:big6sadness
        emoml:arousal 5.21e+00 ;
        emoml:dominance 2.82e+00 ;
        emoml:valence 2.21e+00 ] ] ;
   onyx:convertsFrom emoml:big6 ;
   onyx:convertsTo emoml:pad ;
   senpy:description "Plugin to convert emotion sets from
         Ekman to VAD" ;
   senpy:module "senpy.plugins.conversion.emotion.
        centroids" ;
   senpy:name "Ekman2PAD" ;
   senpy:version 1e-01 .
```

Once all the dimensions are mapped into the PAD model, the *fusion service* combines the results of the different modalities and compute the final results. We have chosen the weighted average classifier fusion technique for this task, since within this schema (i) different analyzers are considered independent of each other (comparing with feature fusion techniques), and (ii) each modality may contribute differently for each emotion dimension (e. g., it is well-known that speech has higher impact on arousal detection, while facial movements have higher impact on valence detection). Weights can be trained offline, by heuristics, or the uniform weights can can be used if no information is provided.

Listing 4 shows an edited fragment of the annotations at the end of the whole process. The actual results were collected in JSON-LD format and stored in an elasticsearch database, and Kibana was used for visualization [1].

Listing 4. RESULTS FROM THE FUSION PHASE

```
ex:video.mp4:time=0,2 a nif:Context ;
   onyx:hasEmotionSet [
     prov:wasGeneratedBy ex:videoAnalysis ;
     onyx:hasEmotion [
       emoml:pad_arousal 0.5 ;
       emoml:pad_valence 0.1
       ] .
     ] ,
     [
     prov:wasGeneratedBy ex:audioAnalysis ;
     onyx:hasEmotion [
       emoml:pad_pleasure -0.2 ;
       emoml:pad_arousal -0.4
       ] .
     ] ;
     _:textResults ;
     [
     prov:wasGeneratedBy <http://servicehost/api/plugins/
          Ekman2PAD_0.1> ;
     prov:wasDerivedFrom _:testResults ;
     onyx:hasEmotion [
       emoml:pad_pleasure 0 ;
       emoml:pad_arousal 0 ;
       emoml:pad_dominance 0
       ] .
     ] ;
     prov:wasGeneratedBy ex:fusion ;
     onyx:hasEmotion [
       emoml:pad_arousal 0.049 ;
       emoml:pad_valence -0.05 ;
       ] .
     ] .

_:textResults prov:wasGeneratedBy ex:textAnalysis ;
     onyx:hasEmotion [] ,

ex:fusion
   a onyx:EmotionConversion ;
   onyx:convertsFrom emoml:pad-dimensions ;
   onyx:convertsTo emoml:pad-dimensions .
```

5. Conclusions

In this paper, we proposed an approach for the integration of emotion analysis in different modalities (multimodal) and using different emotion representation models (multimodel). The proposed linked data vocabulary unifies and extends existing vocabularies to provide a complete coverage of multimodal multimodel emotion annotations, including the unambiguous definition of conversion to different emotion models. The vocabulary is compatible with existing specifications and recommendations, such as EmotionML. Additionally, it integrates with the provenance ontology, which means annotations are modeled as entities whose provenance (origin) can be traced to either an annotation or a conversion activity. These activities can in turn be precisely modeled, including the resources being used, the emotion models adopted, and other entities that were transformed by them. In addition to the model, a reference implementation

1. Elasticsearch and Kibana: https://www.elastic.co/

of automatic emotion conversion has been integrated into senpy (a framework for sentiment and emotion analysis).

Lastly, the applicability and completeness of this approach and the reference implementation has been assessed through a use case that integrates multimodel multimodal annotations.

Acknowledgments

The research leading to these results has received funding from the European Union's Horizon 2020 Programme research and innovation programme under grant agreement No. 644632 (MixedEmotions). The work of J. Fernando Sánchez has been supported by the Spanish Ministry of Economy and Competitiveness under the R&D project SEMOLA (TEC2015-68284-R).

References

[1] K. R. Scherer, "Psychological models of emotion," *The neuropsychology of emotion*, vol. 137, no. 3, pp. 137–162, 2000.

[2] P. Baggia, C. Pelachaud, C. Peter, and E. Zovato, "Emotion Markup Language (EmotionML) 1.0 W3C Recommendation," W3C, Tech. Rep., May 2014. [Online]. Available: http://www.w3.org/TR/emotionml/

[3] J. F. Sánchez-Rada and C. A. Iglesias, "Onyx: A Linked Data Approach to Emotion Representation," *Information Processing & Management*, vol. 52, pp. 99–114, January 2016.

[4] E. Cambria, B. Schuller, Y. Xia, and C. Havasi, "New avenues in opinion mining and sentiment analysis," *Intelligent Systems, IEEE*, vol. 28, no. 2, pp. 15–21, March 2013.

[5] M. Munezero, C. Montero, E. Sutinen, and J. Pajunen, "Are they different? affect, feeling, emotion, sentiment, and opinion detection in text," *Affective Computing, IEEE Transactions on*, vol. 5, no. 2, pp. 101–111, April 2014.

[6] B. Pang and L. Lee, "Opinion mining and sentiment analysis," *Foundations and trends in information retrieval*, vol. 2, no. 1-2, pp. 1–135, 2008.

[7] S. M. Mohammad, "Sentiment analysis: Detecting valence, emotions, and other affectual states from text," in *Emotion Measurement*, H. Meiselman, Ed. Woodhead Publishing, 2016.

[8] P. Ekman, "Basic emotions," *Handbook of cognition and emotion*, vol. 98, pp. 45–60, 1999.

[9] J. J. Prinz, *Gut reactions: A perceptual theory of emotion*. Oxford University Press, 2004.

[10] E. Cambria, A. Livingstone, and A. Hussain, "The hourglass of emotions," in *Cognitive Behavioural Systems*. Springer, 2012, pp. 144–157.

[11] R. Plutchik, *Emotion: A psychoevolutionary synthesis*. Harper & Row New York, 1980.

[12] D. Borth, T. Chen, R. Ji, and S.-F. Chang, "Sentibank: large-scale ontology and classifiers for detecting sentiment and emotions in visual content," in *Proc. 21st ACM international conference on Multimedia*. New York, NY, USA: ACM, 2013, pp. 459–460.

[13] E. Cambria, C. Havasi, and A. Hussain, "Senticnet 2: A semantic and affective resource for opinion mining and sentiment analysis." in *FLAIRS Conference*, 2012, pp. 202–207.

[14] K. R. Scherer, "What are emotions? And how can they be measured?" *Social Science Information*, vol. 44, no. 4, pp. 695–729, 2005.

[15] C. Strapparava and A. Valitutti, "Wordnet-affect: An affective extension of wordnet," in *Proc. LREC*, vol. 4, 2004, pp. 1083–1086.

[16] M. Schröder, H. Pirker, and M. Lamolle, "First suggestions for an emotion annotation and representation language," in *Proc. LREC*, vol. 6, 2006, pp. 88–92.

[17] M. Schröder, L. Devillers, K. Karpouzis, J.-C. Martin, C. Pelachaud, C. Peter, H. Pirker, B. Schuller, J. Tao, and I. Wilson, "What should a generic emotion markup language be able to represent?" in *Affective Computing and Intelligent Interaction*. Springer, 2007, pp. 440–451.

[18] H. N. of Excellence, "Humaine emotion annotation and representation language (earl): Proposal." HUMAINE Network of Excellence, Tech. Rep., Jun. 2006. [Online]. Available: http://emotion-research.net/projects/humaine/earl/proposal#Dialects

[19] K. Ashimura, P. Baggia, F. Burkhardt, A. Oltramari, C. Peter, and E. Zovato, "Emotionml vocabularies," W3C, Tech. Rep., May 2012. [Online]. Available: http://www.w3.org/TR/2012/NOTE-emotion-voc-20120510/

[20] C. A. Iglesias, J. F. Sánchez-Rada, G. Vulcu, and P. Buitelaar, "Linked Data Models for Sentiment and Emotion Analysis in Social Networks," in *Sentiment Analysis in Social Networks*. Morgan Kauffman, October 2016, ch. Linked Dat, pp. 46–66.

[21] M. Schröder, P. Baggia, F. Burkhardt, C. Pelachaud, C. Peter, and E. Zovato, "Emotionml – an upcoming standard for representing emotions and related states," in *Affective Computing and Intelligent Interaction*, ser. Lecture Notes in Computer Science, S. D'Mello, A. Graesser, B. Schuller, and J.-C. Martin, Eds. Springer Berlin Heidelberg, 2011, vol. 6974, pp. 316–325.

[22] S. Hellmann, J. Lehmann, S. Auer, and M. Brümmer, "Integrating nlp using linked data," in *The Semantic Web–ISWC 2013*. Springer, 2013, pp. 98–113.

[23] J. F. Sánchez-Rada, C. A. Iglesias, and R. Gil, "A Linked Data Model for Multimodal Sentiment and Emotion Analysis," Beijing, China, July 2015, pp. 11–19.

[24] Media fragments URI 1.0 (basic). 00005.

[25] W. Chou, D. A. Dahl, G. Mccobb, and D. Raggett, "EMMA: Extensible Multi-Modal Annotation Markup language," 2005.

[26] P. Groth and L. Moreau, "Prov-o w3c recommendation," W3C, Tech. Rep., 2013. [Online]. Available: http://www.w3.org/TR/prov-o/

[27] J. F. Sánchez-Rada, C. A. Iglesias, I. Corcuera-Platas, and O. Araque, "Senpy: A pragmatic linked sentiment analysis framework," in *Proc. DSAA Special Track on Emotion and Sentiment in Intelligent Systems and Big Social Data Analysis (SentISData)*, 2016.

[28] S. M. Kim, A. Valitutti, and R. A. Calvo, "Evaluation of unsupervised emotion models to textual affect recognition," in *Proc. NAACL HLT Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*. Association for Computational Linguistics, 2010, pp. 62–70.

[29] M. M. Bradley and P. J. Lang, "Affective norms for English words (ANEW): Instruction manual and affective ratings," University of Florida, Tech. Rep., 2010.

[30] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 9, pp. 1627–45, sep 2010.

[31] F. Eyben, F. Weninger, F. Gross, and B. Schuller, "Recent developments in opensmile, the munich open-source multimedia feature extractor," in *Proc. 21st ACM International Conference on Multimedia*, ser. MM '13. New York, NY, USA: ACM, 2013, pp. 835–838.

[32] M. Schmitt, F. Ringeval, and B. Schuller, "At the border of acoustics and linguistics: Bag-of-audio-words for the recognition of emotions in speech," in *Proc. 17th International Speech Communication Association*. San Francisco, USA: ISCA, 2016, pp. 495–499.

3.2.3 MixedEmotions: An Open-Source Toolbox for Multi-Modal Emotion Analysis

Title	MixedEmotions: An Open-Source Toolbox for Multi-Modal Emotion Analysis
Authors	Buitelaar, Paul and Wood, Ian D. and Arcan, Mihael and McCrae, John P. and Abele, Andrejs and Robin, Cécile and Andryushechkin, Vladimir and Ziad, Housam and Sagha, Hesam and Schmitt, Maximilian and Schuller, Björn W. and Sánchez-Rada, J. Fernando and Iglesias, Carlos A. and Navarro, Carlos and Giefer, Andreas and Heise, Nicolaus and Masucci, Vincenzo and Danza, Francesco A. and Caterino, Ciro and Smrž, Pavel and Hradiš, Michal and Povolný, Filip and Klimeš, Marek and Matějka, Pavel and Tummarello, Giovanni
Journal	IEEE Transactions on Multimedia
Impact factor	JCR 2018 Q1 (4.292)
ISSN	1520-9210
Publisher	
Year	2018
Keywords	affective computing, audio processing, emotion analysi, linked data, open source toolbox, text processing, video processing
Pages	
Online	http://ieeexplore.ieee.org/document/8269329/
Abstract	Recently, there is an increasing tendency to embed the functionality of recognizing emotions from the user generated contents, to infer richer profile about the users or contents, that can be used for various automated systems such as call-center operations, recommendations, and assistive technologies. However, to date, adding this functionality was a tedious, costly, and time consuming effort, and one should look for different tools that suits one's needs, and should provide different interfaces to use those tools. The MixedEmotions toolbox leverages the need for such functionalities by providing tools for text, audio, video, and linked data processing within an easily integrable plug-and-play platform. These functionalities include: (i) for text processing: emotion and sentiment recognition, (ii) for audio processing: emotion, age, and gender recognition, (iii) for video processing: face detection and tracking, emotion recognition, facial landmark localization, head pose estimation, face alignment, and body pose estimation, and (iv) for linked data: knowledge graph. Moreover, the MixedEmotions Toolbox is open-source and free. In this article, we present this toolbox in the context of the existing landscape, and provide a range of detailed benchmarks on standardized test-beds showing its state-of-the-art performance. Furthermore, three real-world use-cases show its effectiveness, namely emotion-driven smart TV, call center monitoring, and brand reputation analysis.

MixedEmotions: An Open-Source Toolbox for Multi-Modal Emotion Analysis

{Paul Buitelaar, Ian D. Wood, Sapna Negi, Mihael Arcan, John P. McCrae, Andrejs Abele, Cécile Robin, Vladimir Andryushechkin, Housam Ziad}@*National University of Ireland Galway, Ireland,* {Hesam Sagha, Maximilian Schmitt, Björn W. Schuller}@*Chair of Complex & Intelligent Systems, University of Passau, Germany,* {J. Fernando Sánchez-Rada, Carlos A. Iglesias}@*GSI Universidad Politécnica de Madrid, Spain,* {Carlos Navarro}@*Paradigma Digital, Spain,* {Andreas Giefer, Nicolaus Heise}@*Deutsche Welle, Germany,* {Vincenzo Masucci, Francesco A. Danza, Ciro Caterino}@*Expert Systems, Italy,* {Pavel Smrž, Michal Hradiš}@*Brno University of Technology, Czech Republic,* {Filip Povolný, Marek Klimeš, Pavel Matějka}@*Phonexia, Czech Republic,* {Giovanni Tummarello}@*Siren solutions, Ireland.*

***Abstract*—Recently, there is an increasing tendency to embed functionalities for recognizing emotions from user generated media content in automated systems such as call-centre operations, recommendations and assistive technologies, providing richer and more informative user and content profiles. However, to date, adding these functionalities was a tedious, costly, and time consuming effort, requiring identification and integration of diverse tools with diverse interfaces as required by the use case at hand. The MixedEmotions Toolbox leverages the need for such functionalities by providing tools for text, audio, video, and linked data processing within an easily integrable plug-and-play platform. These functionalities include: (i) for text processing: emotion and sentiment recognition, (ii) for audio processing: emotion, age, and gender recognition, (iii) for video processing: face detection and tracking, emotion recognition, facial landmark localization, head pose estimation, face alignment, and body pose estimation, and (iv) for linked data: knowledge graph integration. Moreover, the MixedEmotions Toolbox is open-source and free. In this article, we present this toolbox in the context of the existing landscape, and provide a range of detailed benchmarks on standard test-beds showing its state-of-the-art performance. Furthermore, three real-world use-cases show its effectiveness, namely emotion-driven smart TV, call center monitoring, and brand reputation analysis.**

***Index Terms*—emotion analysis, open source toolbox, affective computing, linked data, audio processing, text processing, video processing**

I. Motivation & Introduction

ANY Media content (e. g., social media, TV/Radio program) contains a vast amount of information which can be harvested for various analysis from a content perspective (e. g., reputation analysis [1], content emotion analysis [2]) and a content-authors perspective (e. g., user profiling and recommendation [3], [4], user community analysis [5], [6]). Nevertheless, as part of this information, the emotional aspects of the media content has not received its well-deserved attention and its utility of those aspects have not yet been well-exploited in real-world or commercial scenarios. Emotions are important part of human life as they enhance communication and understanding between people. Similarly, incorporating emotion-related information into multimedia content and multimedia analysis could enhance usability and user-adaptability. Although some research advances have been made in this direction (such as: emotion analysis of users' audio or video for enriching users' profiles for media recommendation [3], [7], [4], affect prediction from movies [8], or speech [9]), they have not gone further than research, and reproduction of such algorithms is time consuming and fault-prone.

The MixedEmotions Toolbox[1] introduced herein fills this gap by providing a plug-and-play and ready-to-use set of emotion recognition modules that can be used in isolation or in combination through predefined or configurable workflows. It provides a unified solution for large-scale emotion analysis on heterogeneous, multilingual, text, speech, video, and social media data streams, leveraging open access and proprietary data sources **including modules for collection of social media data**, and exploiting social context by leveraging social network graphs. It also includes entity linking and knowledge graph technologies for semantic-level emotion information aggregation and integration. **Available free tools have been adapted and included in the platform alongside tools developed by the authors of this paper.**

This paper describes the current version of the MixedEmotions Toolbox, including its underlying architecture, the modules it comprises and their capabilities, and applications of the platform in three representative multimedia-related use cases: Social TV, Brand Reputation Management, and Call Center Operations.

Before describing the toolbox in detail, we describe what *emotion* actually is and how it is represented and provide a quick review of existing emotion analysis platforms as well as an overview of requirements for emotion analysis on big-data.

Hesam Sagha is now working at audEERING GmbH.

Björn W. Schuller is also with the Department of Computing, Imperial College London, United Kingdom.

[1]MixedEmotions Toolbox is the outcome of the European Project MixedEmotions (https://mixedemotions-project.eu/). Note that it is not about the 'co-occurrence of different emotions' (as the psychological term 'mixed emotions'), but about the 'emotions from mixed modalities'.

A. What is Emotion?

One of the most complete and accepted definitions of emotion is proposed by Scherer [10] through a component process model, in which an emotion is a synchronization of different cognitive and physiological components in response to a stimulus event. The expression of emotions through facial and vocal changes is originated from the 'somatic nervous system' component. Moreover, emotions and preferences (as stable emotions with low behavioural impacts) can be conveyed through verbal or written *content*, such as product reviews, opinions, and suggestions. Therefore, analysing the facial and vocal changes as well as verbal and written content provides clues for automatic emotion recognition.

B. Quick overview of emotion representations used

Various representation schemes for emotions have been proposed, each based on particular criteria. Ekman's *six basic emotions* (Anger, Fear, Surprise, Happiness, Disgust, Sadness) are based on the universality of those emotions [11]; Plutchik's *wheel of emotion* is further based on contrast and closeness of emotions [12]; Russel's *Circumplex model* is constructed to capture the core affect in a two dimensional (Arousal and Valence) model [13], [14]; Osgood identified three primary dimensions of emotion expression (Pleasure, Arousal, Dominance) [15]; and more recently, Fontaine et al. identified a fourth dimension (unpredictability) [16]. Arousal reflects the level of energy in the emotion (e. g., pleased vs. ecstatic); valence reflects the hedonic tone (e. g., pleasant vs. unpleasant); dominance represents the sense of control or dominant nature of the emotion (e. g., fear vs. anger); and unpredictability refers to the appraisal of expectedness or familiarity.

In the MixedEmotions Toolbox, the preferred emotion representation model is the four dimensional model, combined with emotion intensity as a fifth dimension and a level of confidence in the measurement. However, due to limitations in available gold standard data and error-prone human ability to map perceived emotions into these dimensions, some modules in the MixedEmotions Toolbox represent emotions as a subset of these dimensions. For emotion representation in audio and video processing, we chose a two (arousal and valence) or three (+ dominance) dimensional emotion model. The choice of the dimensional model is due (among others) to: (i) it can be mapped not only to the six basic emotions but to a myriad of emotion categories, (ii) emotions which resemble each other are located in the vicinity of each other, (iii) it is easier to define continuous values as the output of machine learning systems (such as neural networks), and (iv) it is easier to handle the decision fusion of different subsystems in the continuous domain. In the analysis of text, there were previously no substantial resources annotated with a dimensional emotion model; however, resources and tools that utilize Ekmans' six basic emotions were available. In addition, there are many resources available for "sentiment analysis", which is essentially just the Valence dimension. For this reason, several toolbox modules for text analysis utilize these representation schemes, and functionality is provided for translating to and from a dimensional representation. New data annotated with a four dimensional model is provided alongside models for detecting emotion with this scheme utilising the new data.

C. Existing emotion analysis platforms

Some web services for emotion analysis from textual contents, facial expressions, and speech already exist. Table I summarizes some known services along with their characteristics. As can be seen in the table, all the services are for the analysis of only one modality such as facial, textual, or speech. Moreover, most of the services are not free and not open-source. The MixedEmotions Toolbox overcomes these limitations by providing multi-modal, open-source, free, and user-friendly emotion analyzers.

D. Emotion Analysis in Big Data and Pre-requisites

To deploy a multifaceted emotion analyzer for big-data, the seven "Vs" of big data (Volume, Velocity, Variety, Variability, Veracity, Visualization, and Value) should be addressed. Among them, Variety encompasses multimodality (audio, video, text) and multilinguality/multiculturalism, and Veracity emerges from subjectivity of assessments (annotations). These aspects have been addressed for: (i) the textual modality by: automatic translation [17], defining multilingual WordNet Grid [18], and (ii) for the audio modality by: analyzing within or between language family emotion recognition [19], feature transfer learning between languages [20], model transfer learning [21], language identification [22], audio denoising [23], and decision aggregation through cooperative speaker models [24]. Regarding the Volume and Velocity, there is a need for fast computation. This has been investigated using End-to-End approaches for speech emotion analysis [25], fast GPU processing of audio and video processing [26], and crowdsourcing and a semi-supervised active learning approach for automatically labeling large amounts of data [27], [28]. Some of these aspects have been deployed within the MixedEmotions Toolbox. Further, the MixedEmotions Toolbox can be easily deployed on one or more machines for distributed analysis and fast processing of large amount of data. A Visualization module is also included in the toolbox (Section III-D). Moreover, to investigate the Value of this MixedEmotions Toolbox, we designed three case studies on multimedia emotion processing which will be discussed in Section IV.

II. Architecture overview

The MixedEmotions Toolbox follows a microservice architecture in which the modules in the toolbox are independent of each other, so users need only the modules required for his/her analysis and can skip the others. The modules are containerized using Docker[2], and therefore can be deployed without dependency restrictions, with the only requirement being a Docker server. Docker servers exist for all major operating systems, can be installed on small computers as well as in extensible cloud environments. As well as individual modules, users can also benefit from an orchestrator in the toolbox to enable big data operations sustained on horizontal

[2]www.docker.com

TABLE I
A SHORT LIST OF AVAILABLE EMOTION ANALYZER SERVICES FOR T(EXTUAL), F(ACIAL), AND S(PEECH) CONTENTS.

Service	Modality	Open Source	Free
IBM Watson AlchemyLanguage (www.ibm.com/watson),Bitext (www.bitext.com)	T	No	No
MoodPatrol (market.mashape.com/soulhackerslabs/)	T	No	No
Synesketch (krcadinac.com/synesketch)	T	Yes	Yes
Microsoft Cognitive Services (www.microsoft.com/cognitive-services)	F	No	No
IMOTIONS (wwww.imotions.com)	F	No	No
Affectiva Emotion API (www.affectiva.com)	F	No	Free/Enterprise Editions
EmoVu (www.emovu.com), CrowdEmotions (www.crowdemotion.co.uk)	F	No	?
Nviso (www.nviso.ch/technology.html), SkyBiometry (www.skybiometry.com)	F	No	Limited/Non-Free Editions
audEERING SensAI (www.audeering.com/technology/sensai/)	S	Yes	Free Research Edition (openSMILE)
Good Vibrations (www.good-vibrations.nl)	S	No	No
Vokaturi (www.vokaturi.com/)	S	No	Limited/Enterprise Editions

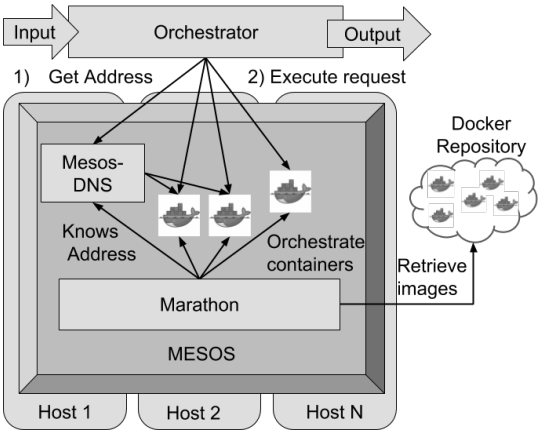


Fig. 1. Orchestrator within the MixedEmotions Toolbox.

scalability (using more machines). This orchestrator provides users an easy starting point to build applications as needed. In a nutshell, the orchestrator is an ETL[3] pipeline [29] adapted to the structure of the MixedEmotions, thus, it is suited to work with Docker containers deployed in Mesos[4](Fig. 1), as well as external services as long as they have a REST API[5]. It is fully configurable with plain text configuration files, so a user does not need to have programming skills.

Note that Docker Servers and Mesos Services can be deployed on multiple platforms, including, Linux, OS X, Windows and Windows Server, and making the MixedEmotions Toolbox platform independent.

[3]Extract, Transform, Load

[4]mesos.apache.org: The Mesos kernel runs on every machine and provides applications with API's for resource management and scheduling across entire datacenter and cloud environments.

[5]REST = Representational State Transfer, API=Application Programming Interface. This is a simple and widely used standard for providing services over the internet.

Where to find the MixedEmotions Toolbox

The MixedEmotions platform is available online for demonstration and testing[6]. Open source and free for research purposes modules are located on GitHub[7] (source code and documentation), and ready-to-use modules can be found in the MixedEmotions docker repository[8].

III. OVERVIEW OF MAIN FUNCTIONALITIES

In this section, we describe the modules in the MixedEmotions Toolbox for text, audio, and video processing with the focus of emotion recognition.

A. Text Processing

The toolbox includes the following modules for text processing: (1) several modules that implement recognition of affect expressed in text, (2) a module for the recognition of suggestions expressed in text, and (3) modules for semantic processing of text. While sentiment analysis (the recognition of positive/negative sentiment often directed at a particular entity) is an established field with many standard data sets and well developed methodologies (e. g., [30]), the recognition of more nuanced affect has received less attention, and in particular, there are very few gold standard annotated resources. This is also true for analysis of sentiment and emotion from many languages. To address this lack, two new resources for emotion detection from text were developed: (4a) a collection of tweets annotated with four emotion dimensions, and (4b) translations of WordNet into all official European languages, enabling the application of WordNet-based affective lexical resources (e. g., WordNet-Affect [31] and Senti-WordNet [32]) in those languages. Details of these modules and resources are as follows.

1) Sentiment and Emotion Recognition: Models for sentiment and emotion recognition from text across several languages and for general text and social media domains are included in the platform (see Tables II and III).

Several free and/or open source sentiment analysis tools are included in the toolbox. In addition, two Long-Short Term Memory (LSTM) [33] deep learning models trained on

[6]http://mixedemotions.insight-centre.org/

[7]https://github.com/MixedEmotions

[8]https://hub.docker.com/r/mixedemotions/

movie reviews [34] and tweets [35] are provided (see Table II). Evaluation of the English language sentiment models was performed on test tweets from the SemEval2015 task 10B [36] for tweet models, and movie reviews from [37] for general text models. F1 scores from the cross-validation analysis of the training data are also provided where appropriate.

The toolbox includes models for emotion detection from text for two emotion representation schemes: Ekman's six emotion categories [11] and the 4-dimensional Valence/Arousal/Dominance/Surprise representation scheme [16] (see Table III). These models fall into two broad categories: unsupervised lexicon based models, provided primarily as baseline systems, and supervised models trained on publicly available annotated data sets. The lexicon based models count word occurrences, summing associated emotions. Models built with WordNet-Affect [31] for Ekman emotions and Affective Norms for English Words (ANEW) [43], [44] for VAD are also provided. Two supervised Ekman models are included: one trained on tweet data utilising emotion hash tags as noisy emotion labels [45] and another from the recent WASSA shared task on emotion recognition [46], [47]. A final model trained on new VADS annotated data (see Section III-A4a) is also provided. F_1 and R^2 scores from the cross-validation analysis of the training data are provided where appropriate.

2) Suggestion Mining: Alongside requirements for the detection of sentiment and emotions in an opinionated text, another useful service which has been developed in the MixedEmotions Toolbox is the identification of suggestions and advice that may have been made in those texts. This will allow users and service providers to make more valuable decisions based on richer inferences on data (e. g., a brand reputation can be affected by positive and negative suggestions from the users alongside with their expressed sentiments).

Suggestion mining refers to the task of detection of such suggestions (advice, tips, recommendations, etc.) in the text obtained from social media. An example of suggestion in tweets can be: *"Dear Microsoft, release a new zune with your wp7 launch on the 11th. It would be smart"*. Since suggestion mining is a very recent area of research, our contribution also covers the creation of benchmark datasets to facilitate the development and evaluation of suggestion mining methods [49]. Currently, this module is only available for the English language.

The module utilises a Long Short Term Memory (LSTM) Neural Network to classify texts as suggestion or not suggestion. It is trained on suggestion mining datasets developed in-house, using crowdsourced annotations of hotel and electronics reviews [49]. This classifier yields a F_1 score of 0.64 and 0.67 over 10-fold cross-validation for hotel and electronics datasets respectively.

3) Semantic Analysis of Text: The toolbox includes modules for entity recognition in Spanish and English, both built on DBpedia[9] [50]. The English module issues queries to a Lucene[10] database containing all matching WikiPedia URIs, and entities are selected according to the score from the Lucene index. The DBpedia URI, the entity and its type are returned by the module.

The Spanish Entity recognition module is created using entities from DBpedia and their inlink count, which is the number of other entities related to it. Then, an *entities dictionary* is created using all the entities above a certain threshold. Given a text, the module will then extract all the phrases that can be found in the entities dictionary.

4) New Resources for Affective Analysis of Text:

a) Emotion annotated text data (standard and new): There exists a limited number of publicly available emotion annotated text resources; these include: two thousand news headlines annotated with Ekman's six emotions [51], and several dimensionally annotated corpora: Affective Norms for English Texts [52] (a collection of 120 generic texts with VAD annotations), a collection of 2895 Facebook posts annotated by two annotators with Valence and Arousal dimensions [53], and the recent EMOBANK [54] (a collection of ten thousand texts from diverse sources but not including tweets). Moreover, Yu et al. [55] presented a collection of 2009 Chinese sentences from various online texts annotated with Valence and Arousal.

As a step towards addressing this limitation, we collected two new annotated tweet corpora: one containing 2019 generic tweets annotated with *Valence*, *Arousal*, *Dominance*, and *Surprise* (with annotator agreement of Krippendorffs' Alpha .42) [56], and another containing 360 tweets containing expressive emoji annotated with Ekman's six emotions [57] (with annotator agreement of Krippendorffs' Alpha .33).

b) Polylingual WordNet: The Princeton WordNet [58] is one of the most important resources for natural language processing, but is only available for English. Although it has been translated using the *expand* approach to many other languages [59], [60], [61], most of the WordNet resources resulting from these efforts have fewer synsets than the Princeton WordNet. Since manual translation and evaluation of WordNets is a very time consuming and expensive process, we apply Statistical Machine Translation (SMT)[11] to automatically translate WordNet entries. The biggest challenge in translating WordNets with an SMT system lies in the need to translate all senses of a word including low frequency senses. While an SMT system can only return the most frequent translation when given a term by itself, it has been observed that it provides strong word sense disambiguation when the word is given in a disambiguated context [17]. Therefore, we leverage existing translations of WordNet in other languages to identify contextual information for WordNet senses from a large set of generic parallel corpora. We used an approach to select the most relevant sentences from a parallel corpus based on the overlap with existing translations of WordNet in as many pivot languages as possible. The goal is to identify sentences that share the same semantic information with respect to the synset of the WordNet entry that we want to translate. This approach allows us to provide a large multilingual WordNet in 23 different European languages, which we call Polylingual

[9]DBpedia is a crowd-sourced community effort to extract structured information from Wikipedia and make this information available on the Web.

[10]Apache Lucene is an open-source search software: https://lucene.apache.org/

[11]The SMT models also exist as a MixedEmotions' module.

TABLE II
MODELS FOR SENTIMENT DETECTION FROM TEXT.
EVALUATION DATA: SEMEVAL2015 TASK 10B [36] (TWEET SENTIMENT), MOVIE REVIEWS FROM [37] (TEXT SENTIMENT)

Affect Representation	Lang	Domain	Algorithm	Train CV F_1	Test F_1	Reference
Sentiment (+,n,-)	EN	Text	LSTM	—	.76	[33] trained on [34]
Sentiment (++,+,n,-,–)	EN	Text	CoreNLP (NN)	—	.62	[38]
Sentiment (+,n,-)	EN	Text	LingPipe (SVM/NB)	—	.76	[39]
Sentiment (+,n,- / continuous)	EN	Text	VADER (Lexical + Rules)	—	.76	[40]
Sentiment (+,n,-)	EN	Tweets	LSTM	.48	.67	[33] trained on [35]
Sentiment (+,n,-)	EN,ES	Tweets	Sentiment140	.76	.79	[41]
Sentiment (+,n,-)	ES	Tweets	SVM (TASS2015)	.74	—	[42]
Sentiment (+,n,-)	CZ	Text	LingPipe (CZ reviews)	.86	—	[39]

TABLE III
MODELS FOR EMOTION DETECTION FROM TEXT.

Affect Representation	Lang	Domain	Algorithm	Train Eval.	Reference
Emotion (Ekman)	Multiple	Text	WordNet-Affect	—	[31]
Emotion (Ekman)	EN	Tweets	SVM (hashtags)	F_1: .37	[45]
Emotion (4 Ekman Intensities)	EN	Tweets	BLSTM+SVM	R^2: .45	[46] trained on [47]
Emotion (VAD)	EN,ES	Text	ANEW	—	[43], [44]
Emotion (VADS)	EN	Tweets	BLSTM	R^2: .24	[48] trained on new data (See 4a below)

WordNet[12]. As a result, the WordNet-Affect based emotion detection module is also applicable to those languages.

B. Audio Processing

This module recognizes emotions in terms of arousal and valence from speech signals[13]. It is based on the Bag-of-Audio-Words (BoAW) approach [62], trained on continuous emotionally labeled data (the *RECOLA* database [63]). RECOLA is an audio-visual database of 46 subjects during dyadic conversation in French. For each subject, a recording of 5 minutes length has been annotated time-continuously for Arousal and Valence dimensions by six different annotators (3 female, 3 male). From the 6 annotations, a single *gold standard* sequence has been computed for each dimension, using an *evaluator weighted estimator* [64].

BoAW originates from the bag-of-words approach in natural language processing. In this approach, word histogram vectors are used as a feature to classify text documents, e. g., in terms of *sentiment* or the author's *gender* [65]. For BoAW, the first step is the extraction of acoustic low-level descriptors (LLDs) from the raw waveform of the speech signal. audEERING's open-source toolkit openSMILE[14] [66] is used to extract *Mel-frequency cepstral coefficients (MFCCs)* and logarithmic energy over a short audio frame of 25 ms, with a step size of 10 ms. Each 13-dimensional LLD vector is then assigned to a so-called *audio word*, i. e., a template of an LLD vector. This is accomplished through a vector quantization step using a codebook which has been learned beforehand. A random sampling [67] of 200 LLDs from the training data has proven to be suitable for the task. In the vector quantization step, Euclidean distance is taken into account.

[12]http://polylingwn.linguistic-lod.org/

[13]**Although *audio* includes speech, music, and other acoustics, the module that we built within the MixedEmotions Toolbox is for speech. Other modules, such as music emotion may be added later to the toolbox.**

[14]opensmile.audeering.com

TABLE IV
PERFORMANCE (CCC) OF THE AUDIO EMOTION RECOGNITION MODULE ON THE RECOLA DATABASE.

Database	Partition	Arousal	Valence
RECOLA	Development	.797	.529
	Test	.722	.452
SEWA	Development	.359	.157

To make the power of the histogram independent from the duration of the input segment, a histogram normalization is performed. The whole BoAW-processing is accomplished by the open-source toolkit openXBOW[15] [68].

For decoding, a *support vector regressor* (SVR) with a linear kernel was trained [69]. All hyperparameters have been optimized systematically using a speaker-independent split of the database into training, validation, and test partitions [62]. The performance of arousal and valence recognition in terms of *Concordance Correlation Coefficients (CCC)*[16][70] for the RECOLA and SEWA [71] datasets are summarized in Table IV.

C. Video Processing

This module is responsible for emotion recognition (arousal/valence and Ekman's emotions) from facial gestures. The emotion recognition runs on top of face detection and tracking, facial landmark localization, head pose estimation, and face alignment. Face detection is based on a discriminatively trained deformable part model [72] which runs at approximately 8-16 fps on 720p video. Faces are tracked in a video according to standard tracking by detection. To maintain identities across these partial tacks, visual fingerprints are extracted from individual frames, and clustered by hierarchical

[15]https://github.com/openXBOW/openXBOW

[16]**CCC is similar to the Correlation Coefficient, but it also considers the mean and variance of the two random variables.**

TABLE V
FACE VERIFICATION ACCURACY OF CNN [73] FINETUNED WITH AFFINE AND SIMILARITY GEOMETRIC ALIGNMENTS ON YOUTUBE FACES DATASET.

Original	Affine alignment	Similarity alignment
.973	.974	.977

clustering using complete linkage and cosine distance. The features are then used as activations of a convolutional neural network (CNN) [73] which is fine-tuned on the Megaface dataset [74] for similarity transform facial alignment (See Table V for the effect of the fine-tuning).

Facial landmarks are localized by an ensemble of regression trees [75] which provides decent facial point localization at real-time speed even on a single core CPU. The faces are aligned using similarity transformation. Head orientation is estimated by Random Regression Forests [76] trained on AFLW dataset [77]. Body pose is tracked using Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields [78] which can run at 10fps on an Nvidia GeForce GTX 1080 graphics card and can handle arbitrary poses, occlusion, and motion blur.

Facial expressions (sadness, happiness, surprise, disgust, anger) are estimated from aligned face regions using a CNN consisting of four convolution layers, two pooling layers, and three fully connected layers. The network achieves classification accuracy of 0.705 among five expression classes on the Facial Expression Recognition Challenge dataset [79]. More detailed facial information is extracted using the OpenFace toolkit [80] which implements a Constrained Local Neural Field (CLNF) deformable model for gaze tracking [81] and additional Support Vector Machine and Support Vector Regression models trained on the merged SEMAINE [82], DISFA [83], and BP4D [84] data for facial action unit detection.

Visual valence and arousal models were trained on the *RECOLA* database [63]. These models reuse activation features from the fully connected layers of the facial expression network (CNN-fc5 for the first fully connected layer and CNN-fc6 for the second). The per-frame features are compressed using PCA[17], basic statistics are computed from a temporal window (mean, variance, minimum, maximum), and statistics from several neighboring frames are compressed again using PCA. The models built on these features are linear regressors trained with *Concordance Correlation Coefficients (CCC)*[18] [70] objective function and weight decay. The results on the training and validation parts of the RECOLA database from AV+EC 2016 Challenge [85] are shown in Table VI.

TABLE VI
PERFORMANCE (CCC) OF THE VIDEO EMOTION RECOGNITION USING CNN ON AV+EC 2016 CHALLENGE DEVELOPMENT SET. FEATURES VIDEO-APPEARANCE AND VIDEO-GEOMETRIC WERE PROVIDED AS BASELINES BY THE CHALLENGE ORGANIZERS.

	Valence	Arousal
Video Appearance	.474	.483
Video Geometric	.612	.379
CNN-fc5	.512	.532
CNN-fc6	.498	.585

D. Linked Data and Knowledge Graph

MixedEmotions Toolbox intends to exploit (emotion-related) information across different sources (i. e., emotion analysers for text, audio, video). To enable this capability, Linked Data principles have been investigated to define protocols and approaches to link the information of these sources to each other [86]. In the MixedEmotions Toolbox, the JSON Linked-Data (JSON-LD) format has been used for this task (Section III-D1). The use of linked data formats allows us to easily connect resources to common sense knowledge captured in knowledge graphs such as DBpedia [50] (Section III-D2).

1) Linked Data Representation: The MixedEmotions Toolbox follows a linked data approach in its services. The pillars of this approach are: (i) a representation model for all types of annotations covered by the toolbox (sentiments, emotions, suggestions), (ii) a means to uniquely identify annotations, (iii) a representation format to capture those annotations, (iv) a common interface for services within the toolkit to allow communication between them, and (v) a set of tools that unites all these aspects and enables the creation of new services. This section briefly covers these aspects, focusing on the representation.

The representation model includes all the concepts in the domain (*social post*, *entity*, *emotion*) and their properties or relationships (e. g., post *has* emotion, emotion *is of category* happy). Rather than creating an ad-hoc model for each domain, linked data principles encourage reusing already existing models. These models are also referred to as ontologies, vocabularies, or specifications. There are three vocabularies that are very relevant for sentiment and emotion annotation: Marl [87] (to annotate and describe subjective opinion), Onyx [88] (to annotate and describe emotions) with interoperability with Emotion Markup Language (EmotionML) [89] and NLP Interchange Format (NIF) 2.0 [90] (a semantic format and API for Natural Language Processing services). **Moreover, the Onyx vocabulary provides a meta-model of emotions, i. e., instead of defining a set of categories or dimensions for emotions, it provides a meta-model so that different models can be defined and uniquely identified. It also contains definitions for the emotion models (vocabularies) in Emotion-ML and WordNet-Affect. Hence, annotators and service developers can be specific about what emotion models they are using (e. g., Ekman's big-6 categorical model,Russel's Circumplex model, etc).**

Nevertheless, these models alone may not cover all the possible needs of possible use-cases for the the MixedEmotions Toolbox. Therefore, additional concepts (e. g., suggestions, multi-results that include several entries and multimedia results) are defined, and the final proposed model (named the "MixedEmotions model") contains existing models (Marl, Onyx, NIF) and their extensions. This model uses NIF as the

[17] Principal Component Analysis

[18] **CCC is similar to the Correlation Coefficient, but it also considers the mean and variance of the two random variables.**

foundation for annotation of NLP results. NIF also provides different URI Schemes to identify text fragments inside a string, e. g., a scheme based on RFC5147 [91], and a custom scheme based on context. To this end, texts are converted to RDF[19] literals and a URI[20] is generated so that linked data annotations can be defined for that text. The same idea can also be applied to annotate multimedia [92]. **The combination of Onyx's meta-models of emotion with the homogeneous multimedia annotation can be leveraged for automatic conversion and fusion of multimodal results [93].**

To serialize these annotations, the MixedEmotions toolbox uses a common JSON-LD (JSON for Linked Data) schema. JSON-LD is a way of encoding Linked Data as JSON which provides a balance between semantic expression and ease of use for developers [94]. Moreover, this format is a good fit with the REST API that NIF defines for Natural Language Processing (NLP) services with standardized parameters. The MixedEmotions API adds several new concepts and parameters to those originally included in NIF, to cover the broad scope of the toolbox. It also establishes JSON-LD as its standard serialization format.

Lastly, these concepts are tightly integrated in the development kits and libraries provided by the MixedEmotions Toolkit. **A notable example is Senpy[21], a linked data framework for NLP services [95]. The aim of Senpy is to allow researchers to effortlessly turn their NLP analysis (e. g., sentiment and emotion analysis) into semantic web services. It also provides a series of common features that complement the services by leveraging their inherent semantics, such as automatic emotion model and format conversion, normalization of results and pipelining of several analysis. Senpy has been extensively used in the development of several modules of the MixedEmotions Toolbox.**

Listing 1 (in the Supplementary Materials) illustrates the semantic representation in a comprehensive example that includes multimodality (audio, video and text), fusion, and conversion of annotation. In particular, this example covers the analysis of the first two seconds of a video (located at `http://example.com/video.mp4`), and fusion of the three modalities. Since fusion requires all modalities to use the same dimensional emotion model, a conversion service exploits the semantic representation of emotion models in each annotation to find the appropriate conversion mechanism. As a result, the text results are converted form a categorical model to a dimensional one.

2) Knowledge Graph (KG): Knowledge graph theory uses graphs for the representation of concepts such as in medical and sociological texts [96]. The cumulation of such graphs can work as decision support system that can document the consequences of actions. The combination of knowledge graphs with concept models [97] led to the development of ontologies, which focused on the logical relations of concepts instead of words.

[19]Resource Description Framework
[20]Uniform Resource Identifier
[21]https://github.com/MixedEmotions/senpy

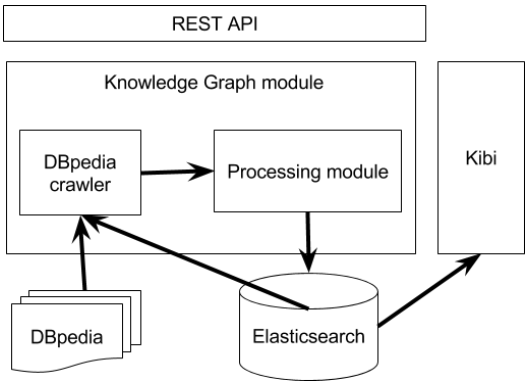


Fig. 2. Knowledge Graph module architecture.

For a long time, knowledge graph theory was used for specific tasks: modeling of ecosystems or in linguistics for analyzing content of books [98]. Recently, the increasing popularity of linked data and the emergence of knowledge bases made large and general purpose knowledge graphs possible, such as Google's Knowledge Graph, which is a compilation of facts and figures that provides contextual meaning to its searches [99]. In the MixedEmotions Toolbox, we provide a KG module that can be used to provide insights into relations between recognized entities using semantic knowledge from DBpedia [50]. **The Entity Extraction and Linking module identifies entities mentioned in the analyzed resources, and then the KG module links them to the entities in DBpedia, so more specific information can be obtained about them (see Supplementary Materials: entities).** Once the relations are extracted and filtered to keep the relevant ones only, they are stored in an Elasticsearch[22] database alongside other content metadata such as emotion annotations, where they can be readily visualized (e. g., using the Kibi graph browser, see below). **The resulting KG contains the extracted entities, their specifications and related information, and the relations among them.** The KG module is managed by a REST API, and needs an index in the Elasticsearch database that contains both the source text and the entities extracted. **It can be queried by exploiting the REST API, so other modules can retrieve parts of the graph; in addition, it can be navigated through the Kibi graph browser (see Supplementary materials: Kibi browser).**

The architecture of the KG module consists of five main parts: the Database, the DBpedia crawler, the Processing module, the Web server that exposes a REST interface, and the Kibi graph browser:

- **Database**: Elasticsearch repository stores information processed by other modules as well as KG module.
- **DBpedia crawler** is responsible for crawling information from DBpedia, that is, related entities in the Database

[22]Elasticsearch provides a distributed, multitenant-capable full-text search engine with an HTTP web interface and schema-free JSON documents.

that are identified by the Entity Extraction and Linking module.

- **Processing module** filters the extracted information and splits it by type. The Entity Extraction and Linking module assigns one of the three types to the recognized entity: Person, Organization, and Location. Each type is processed separately so they can be stored in separate indexes. As the extracted information is not always 'clean' (it can wrongly be classified as a certain type of entity), the module applies customized filters for each type of entity to reduce the number of wrongly classified entities. Apart from writing the extracted information to the Database, the KG module automatically defines links between entities, adds the mapping of relations to Elasticsearch, and creates Kibi dashboards for each type of entity as well as a dashboard for the graph browser.
- **Web Server** allows monitoring and control of the KG module externally through a REST API.
- **Kibi graph browser** is a very powerful platform for interactive, exploratory big/streaming data discovery and alerting, with specific focus on exploration/leveraging of relationships across datasets. It performs 'on the fly' analytics on the collected entities and processed data stored in Elasticsearch. The Kibi graph browser provides the capability to visualize connections between entities and explore existing connections based on relations in DBpedia.

E. Social network analysis

In general, sentiment and subjectivity are quite context-sensitive [100]; The meaning of a particular piece of content (e. g., a tweet, a Facebook status, or a blog post) may only be fully understood when its social context is taken into consideration. In fact, social context has an effect on the behaviour of users in social networks [101]. Recent work has demonstrated the existence of certain patterns in relationships in social media, which is explained by several social theories [102]. One notable example is social influence [103], which pertains to behavioural changes due to perceived relationships with other people, organizations and society in general.

Detecting and characterising social contexts and the emotions that are expressed therein has multiple applications. First, the detection of the most relevant shared content (e. g., tweets or posts), users (e. g., influencers), and groups of users (communities) provides a path for micro-analysis of opinions in brand monitoring [104] and content recommendation scenarios. Second, emotion propagation patterns can be used for both analysis and prediction of expected social influence of a message [105], [106], [107], [108]. Those same patterns may also indicate false information or rumours [109]. Finally, social features can improve sentiment analysis and emotion detection [110], [111]. This can be specially relevant in microblogging based social networks such as Twitter, where the short length of the content makes the task very complex.

'Scanner' as a module in the MixedEmotions Toolbox that provides a standalone framework for crawling and analysing Twitter contents to perform social network and emotion analysis. It is capable of calculating different social metrics (e. g., content metrics, group metrics, temporal metrics, influence metrics). The architecture of this module is depicted in Fig.3.

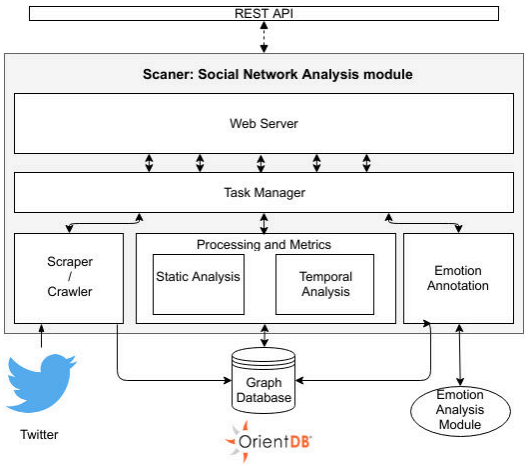


Fig. 3. Architecture of the Social Context Analysis module 'Scanner'.

F. Decision fusion

Since within the MixedEmotions Toolbox, emotions can be extracted from diverse modalities (video, audio, text) and sources, there is a need to combine extracted results and yield a final (more reliable) estimate. For this, the decision fusion module accepts the outputs (in the MixedEmotions JSON-LD format) of modules that represent emotions in terms of continuous arousal and valence (irrespective of modality), and combines them by a weighted average of the values. The choice of classifier fusion (vs. feature fusion) is to keep modules independent of each other, and the choice of weighted average is because each modality may contribute differently to recognizing emotions (for example, it is known that valence can be recognized better via facial monitoring, while arousal can be recognized better via speech monitoring). Weights can be learned offline, set manually, or have the same values.

IV. Use Cases of the MixedEmotions Toolbox

The MixedEmotions Toolbox has been tested in the context of three concrete use cases (Emotion-driven Smart TV, Brand Reputation Analysis, Call Center Monitoring) to verify its usefulness.

A. Emotion-driven Smart TV

In this use case, an emotion-driven recommendation engine is developed. The purpose of this engine is to use emotion signals to enhance traditional content- and user-based recommendations for TV programs. More specifically, the Apache Mahout open-source recommender in conjunction with video material published by the broadcaster Deutsche Welle is fed with emotion predictions of the MixedEmotions Toolbox. For

each of Deutsche Welle's videos, the following contents are used for the emotion analyzer:

- the video's title and description text
- the transcription of the video's soundtrack[23]
- twitter messages relating to the video's topics
- the video's soundtrack itself

For each of these contents, the distribution of emotions was calculated using MixedEmotions emotion detection modules for the appropriate modality and fed into the recommendation engine. This was done alongside classical features such as keywords and the percentage of the video duration that the viewers actually watched.

The resulting recommendations were used to present viewers of Deutsche Welle's Apple TV application with suggestions of video contents to watch from two categories:

1) Eudaimonic content — intriguing/challenging videos
2) Hedonic content — joyful/entertaining videos.

These categories are based on recent research into media consumption [112], [113]. The idea is to give viewers the possibility to choose from these two distinct categories depending on their current mood, where they either prefer purely joyful content (e. g., travel and lifestyle) or more intriguing content (e. g., documentaries about conflicts or confrontational interviews). In the Supplementary Materials (dw-snapshot), a snapshot of the emotion analysis of videos of Deutsche Welle's programs after fusing transcription, audio, and tweet analysis is presented. **In this case, the fusion is based on the collective histograms from different modalities. If the histogram is skewed toward positive valence, the content is Hedonic, and if it is skewed toward positive arousal, it is considered as Eudaimonic.**

An A/B test is conducted to verify whether the addition of emotion signals helps to identify videos that a viewer is more likely to prefer — and therefore watch to the end. 1227 users registered and 79 videos were selected as part of the experiment. After a user watches a video, a user has the option to classify it as Eudaimonic or Hedonic and the recommendation engine prepares two sets of videos based on their Eudaimonic and Hedonic contents. A 'hit' is counted when the user selects a video from the same emotional content as the previously-shown video. Overall, 9060 videos were watched. The results are presented in Table VII, and shows that, users tend to watch videos that were proposed by the emotion-driven recommendation engine to a fuller extent (99 % of Hedonic and 92 % of the Eudaimonic video's total duration was actually watched) compared to videos where the recommendation engine did not make use of emotion predictions (88 % and 86 % respectively). Moreover, the performance of classification (Eudaimonic or Hedonic) is 86 %.

In another study, also we investigated if acoustic-based emotional features of a video can help to predict the popularity of that video [114]. We have used the 'Audio processing' module to extract acoustic features. We could achieve 70 % accuracy on recognizing popular vs. non-popular content only using seven features. For more information please refer to [114].

[23]**using https://github.com/MixedEmotions/MixedEmotions/wiki/m17.-Speech-to-text-by-Phonexia**

TABLE VII
THE AVERAGE PERCENTAGE OF THE SUGGESTED VIDEOS WATCHED BY USERS

Mood Category	Without Emotions	With Emotions
Hedonic	88 %	**99** %
Eudaimonic	86 %	**92** %

TABLE VIII
PERFORMANCE OF DIFFERENT APPROACHES FOR SENTIMENT RECOGNITION (3-CLASS TASK) IN TERMS OF UNWEIGHTED AVERAGE RECALL (UAR), EVALUATED ON TWO CZECH CALL CENTERS.

Method	language	Call Center 1	Call Center 2
(i) acoustic	Multi-lingual	.344	.431
(i) acoustic	Czech	.370	.449
(ii) keywords	Czech	.381	.359
(iii) sentiment	English	**.438**	**.496**

B. Call Center Monitoring

Call center Monitoring is the second use-case, which mostly relies on emotion analysis from speech. Call centers offer a promising natural space for emotion mining and analysis. On a daily basis, each agent in a call center encounters customers with different emotions and moods. Recognition of these emotions will help to write better scripts for call center agents that can soothe negative emotions and lead to higher customer satisfaction.

To embed the emotion analysis functionality into this use case, three approaches were considered: (i) acoustic-based valence recognition **with multilingual and Czech models**[24] (ii) analysis of the automatic transcription of the audio, based on the list of pre-defined positive and negative keywords and phrases, and (iii) sentiment recognition on the **translation of the** transcriptions **using the statistical machine translation (SMT) module (Section III-A4b)**. This later approach is an extension of approach (ii) using methods of natural language processing that consider also the context of the utterance. **In this case, we used the Phonexia sentiment analyzer, which is a Long Short-Term Memory Recurrent Neural Network (LSTM-RNN) fed with word2vec word embeddings**[25]**. The system produces the posterior probability of positive sentiment for each sentence, which is then mapped to one of the sentiment classes (positive, negative or neutral)**. In the Supplementary Materials (call-center), we provided a snap shot of this tool.

Acoustic-, keyword- and sentiment-based systems were evaluated on Czech call center data. Transcriptions were automatically translated to English so that the above-mentioned English sentiment analyzer (which is trained on English corpora) can be applied. The results for 3-class sentiment recognition (positive, neutral, negative) are provided in Table VIII. As the results suggest, sentiment analysis on the translated transcriptions outperforms the acoustic- and keyword-based systems.

[24]https://github.com/MixedEmotions/MixedEmotions/wiki/m23.-Audio-Emotion-extraction-by-Phonexia

[25]http://www.fit.vutbr.cz/∼imikolov/rnnlm/

C. Brand Reputation Analysis

Brand Reputation analysis is the third use-case that uses the MixedEmotions Toolbox to implement an application for the assessment of the perceived reputation of a brand or product on the web. Its main objective is to mine selected sources of information and provide human interpretable results that can be investigated by the person in charge of the brand.

This use-case monitors Twitter and YouTube, and processes textual and audio contents to evaluate sentiments and emotions. Entities and the distribution of languages are also extracted. Human-readable results are visualized at real-time using Kibi to compare between different brands and to study emotions and sentiments regarding different dimensions such as hashtags, YouTube channels, or locations. A snapshot of the Kibi for emotion distribution for a Brand is provided in the Supplementary materials (brand reputation).

V. Conclusions

In this paper, we introduced a free, open-source, and multimodal toolbox for emotion analysis: the 'MixedEmotions Toolbox'. The toolbox includes functionalities for text, audio, and video processing with the aim of emotion recognition. Three use cases were described: Emotion-driven Smart TV (emotion-based recommendation), Brand Reputation Analysis (monitoring reputation of a brand from tweets and YouTube videos), and Call Centre Monitoring (monitoring emotion of customers in a help-desk setting). In the future, we hope to see contributions to the release and will ourselves update further functionality aiming beyond improved robustness and increases in efficiency — multimedia data is often 'big', but it is always emotional!

Acknowledgement

The research leading to these results has received funding from the European Union's Horizon 2020 Programme research and innovation programme under grant agreements No. 644632 (MixedEmotions).

References

[1] K. Zhang, D. Downey, Z. Chen, Y. Xie, Y. Cheng, A. Agrawal, W. k. Liao, and A. Choudhary, "A probabilistic graphical model for brand reputation assessment in social networks," in *IEEE/ACM Int. Conf. on Advances in Social Networks Analysis and Mining*, Aug 2013, pp. 223–230.

[2] L. Pang, S. Zhu, and C. W. Ngo, "Deep multimodal learning for affective analysis and retrieval," *IEEE Trans. on Multimedia*, vol. 17, no. 11, pp. 2008–2020, Nov 2015.

[3] S. E. Shepstone, Z. H. Tan, and S. H. Jensen, "Using audio-derived affective offset to enhance tv recommendation," *IEEE Trans. on Multimedia*, vol. 16, no. 7, pp. 1999–2010, Nov 2014.

[4] I. Arapakis, Y. Moshfeghi, H. Joho, R. Ren, D. Hannah, and J. M. Jose, "Integrating facial expressions into user profiling for the improvement of a multimodal recommender system," in *IEEE Int. Conf. on Multimedia and Expo*, Jun 2009, pp. 1440–1443.

[5] F. Huang, X. Li, S. Zhang, J. Zhang, J. Chen, and Z. Zhai, "Overlapping community detection for multimedia social networks," *IEEE Trans. on Multimedia*, vol. 19, no. 8, pp. 1881–1893, Aug 2017.

[6] R. A. Negoescu and D. Gatica-Perez, "Modeling Flickr communities through probabilistic topic-based analysis," *IEEE Trans. on Multimedia*, vol. 12, no. 5, pp. 399–416, Aug 2010.

[7] M. Tkalčič, A. Odić, A. Košir, and J. Tasič, "Affective labeling in a content-based recommender system for images," *IEEE Trans. on Multimedia*, vol. 15, no. 2, pp. 391–400, Feb 2013.

[8] J. Tarvainen, M. Sjöberg, S. Westman, J. Laaksonen, and P. Oittinen, "Content-based prediction of movie style, aesthetics, and affect: Data set and baseline experiments," *IEEE Trans. on Multimedia*, vol. 16, no. 8, pp. 2085–2098, Dec 2014.

[9] Q. Mao, M. Dong, Z. Huang, and Y. Zhan, "Learning salient features for speech emotion recognition using convolutional neural networks," *IEEE Trans. on Multimedia*, vol. 16, no. 8, pp. 2203–2213, Dec 2014.

[10] K. R. Scherer, "What are emotions? and how can they be measured?" *Social Science Information*, vol. 44, no. 4, pp. 695–729, Dec 2005.

[11] P. Ekman and W. V. Friesen, "Constants across cultures in the face and emotion." *J. of personality and social psychology*, vol. 17, no. 2, p. 124, 1971.

[12] R. Plutchik, "A general psychoevolutionary theory of emotion," *Theories of emotion*, vol. 1, no. 3-31, p. 4, 1980.

[13] J. A. Russell, "Core affect and the psychological construction of emotion." *Psychological review*, vol. 110, no. 1, p. 145, 2003.

[14] J. Posner, J. A. Russell, and B. S. Peterson, "The circumplex model of affect: An integrative approach to affective neuroscience, cognitive development, and psychopathology," *Development and psychopathology*, vol. 17, no. 03, pp. 715–734, 2005.

[15] C. E. Osgood, G. J. Suci, and P. H. Tannenbaum, *The Measurement of Meaning*. Urbana, Illinois, USA: University of Illinois Press, 1957.

[16] J. R. J. Fontaine, K. R. Scherer, E. B. Roesch, and P. C. Ellsworth, "The world of emotions is not two-dimensional," *Psychological Science*, vol. 18, no. 12, pp. 1050–1057, 2007.

[17] M. Arcan, J. P. McCrae, and P. Buitelaar, "Expanding WordNets to new languages with multilingual sense disambiguation," in *Proc. 26th Int. Conf. on Computational Linguistics*, Osaka, Japan, 2016, pp. 97–108.

[18] P. Vossen, F. Bond, and J. P. McCrae, "Toward a truly multilingual global wordnet grid," in *Proc. Global WordNet Conference*, Bucharest, Romania, 2016, pp. 419–427.

[19] S. Feraru, D. Schuller, and B. Schuller, "Cross-language acoustic emotion recognition: An overview and some tendencies," in *Proc. 6th biannual Conf. on Affective Computing and Intelligent Interaction*, AAAC. Xi'an, P. R. China: IEEE, Sep 2015, pp. 125–131.

[20] H. Sagha, J. Deng, M. Gavryukova, J. Han, and B. Schuller, "Cross lingual speech emotion recognition using canonical correlation analysis on principal component subspace," in *Proc. 41st Int. Conf. on Acoustics, Speech, and Signal Processing*. Shanghai, P. R. China: IEEE, Mar 2016, pp. 5800–5804.

[21] A. Popková, F. Povolný, P. Matějka, O. Glembek, F. Grézl, and J. H. Černocký, "Investigation of bottle-neck features for emotion recognition," in *Proc. 19th Int. Conf. on Text, Speech, and Dialogue*, P. Sojka, A. Horák, I. Kopeček, and K. Pala, Eds. Brno , Czech Republic: Springer International Publishing, Sep. 2016, pp. 426–434.

[22] H. Sagha, P. Matejka, M. Gavryukova, F. Povolny, E. Marchi, and B. Schuller, "Enhancing multilingual recognition of emotion in speech by language identification," in *Proc. 17th Annual Conf. of the Int. Speech Communication Association*. San Francisco, CA: ISCA, Sep 2016, pp. 2949–2953.

[23] Z. Zhang, F. Ringeval, J. Han, J. Deng, E. Marchi, and B. Schuller, "Facing realism in spontaneous emotion recognition from speech: Feature enhancement by autoencoder with lstm neural networks," in *Proc. 17th Annual Conf. of the Int. Speech Communication Association*. San Francisco, CA: ISCA, Sep 2016, pp. 3593–3597.

[24] A. Mencattini, E. Martinelli, F. Ringeval, B. Schuller, and C. Di Natale, "Continuous estimation of emotions in speech by dynamic cooperative speaker models," *IEEE Trans. on Affective Computing*, vol. 7, 2016.

[25] G. Trigeorgis, F. Ringeval, R. Brückner, E. Marchi, M. Nicolaou, B. Schuller, and S. Zafeiriou, "Adieu features? end-to-end speech emotion recognition using a deep convolutional recurrent network," in *Proc. 41st Int. Conf. on Acoustics, Speech, and Signal Processing*. Shanghai, P. R. China: IEEE, Mar 2016, pp. 5200–5204.

[26] F. Weninger, J. Bergmann, and B. Schuller, "Introducing CURRENNT: the Munich Open-Source CUDA RecurREnt Neural Network Toolkit," *J. of Machine Learning Research*, vol. 16, pp. 547–551, 2015.

[27] S. Hantke, E. Marchi, and B. Schuller, "Introducing the weighted trustability evaluator for crowdsourcing exemplified by speaker likability classification," in *Proc. 10th Language Resources and Evaluation Conference*. Portoroz, Slovenia: ELRA, May 2016, pp. 2156–2161.

[28] Z. Zhang, F. Ringeval, B. Dong, E. Coutinho, E. Marchi, and B. Schuller, "Enhanced semi-supervised learning for multimodal emotion recognition," in *Proc. 41st Int. Conf. on Acoustics, Speech, and Signal Processing*. Shanghai, P. R. China: IEEE, Mar 2016, pp. 5185–5189.

[29] P. Vassiliadis, "A survey of extract–transform–load technology," *Int. J. of Data Warehousing and Mining*, vol. 5, no. 3, pp. 1–27, 2009.

[30] H. Sagha, N. Cummins, and B. Schuller, "Stacked denoising autoencoders for sentiment analysis: a review," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 7, no. 5, pp. e1212–n/a, 2017.

[31] C. Strapparava and A. Valitutti, "WordNet-Affect: an affective extension of WordNet," in *Proc. 4th Int. conf. on Language Resources and Evaluation*, vol. 4, 2004, pp. 1083–1086.

[32] A. Esuli and F. Sebastiani, "Sentiwordnet: A publicly available lexical resource for opinion mining," in *Proc. 6th Int. conf. on Language Resources and Evaluation*, Genoa, Italy, 2006, pp. 417–422.

[33] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. pp. 1735–1780, 1997.

[34] R. Socher, A. Perelygin, J. Y. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," in *Proc. conf. on empirical methods in natural language processing*, vol. 1631. Citeseer, 2013, p. 1642.

[35] P. Nakov, Z. Kozareva, A. Ritter, S. Rosenthal, V. Stoyanov, T. Wilson, and P. Nakov, "Semeval-2013 task 2: Sentiment analysis in twitter," *Atlanta, Georgia, USA*, vol. 312, 2013.

[36] S. Rosenthal, P. Nakov, S. Kiritchenko, S. M. Mohammad, A. Ritter, and V. Stoyanov, "Semeval-2015 task 10: Sentiment analysis in twitter," in *Proc. 9th Int. Workshop on Semantic Evaluation, SemEval*, Denver, Colorado, USA, 2015.

[37] M. Hu and B. Liu, "Mining and summarizing customer reviews," in *Proc. 10th Int. Conf. on Knowledge Discovery and Data Mining*. New York, NY, USA: ACM, 2004, pp. 168–177.

[38] C. D. Manning, M. Surdeanu, J. Bauer, J. R. Finkel, S. Bethard, and D. McClosky, "The Stanford CoreNLP natural language processing toolkit," in *ACL (System Demonstrations)*, 2014, pp. 55–60.

[39] Alias-i, "Lingpipe 4.1.0." Tech. Rep. [Online]. Available: http://alias-i.com/lingpipe

[40] C. J. Hutto and E. Gilbert, "VADER: A parsimonious rule-based model for sentiment analysis of social media text," in *Proc. 8th Int. AAAI Conf. on Weblogs and Social Media*, Oxford, UK, 2014.

[41] A. Go, R. Bhayani, and L. Huang, "Twitter sentiment classification using distant supervision," *CS224N Project Report, Stanford*, vol. 1, p. 12, 2009.

[42] J. V. Román, J. G. Morera, M. Ángel García Cumbreras, E. M. Cámara, M. T. M. Valdivia, and L. A. U. López, "Overview of TASS 2015," in *TASS 2015: Workshop on Sentiment Analysis at SEPLN*, vol. 1397. Alicante: CEUR Workshop Proc., Aachen., 2015.

[43] M. M. Bradley and P. J. Lang, "Affective norms for English words (ANEW): Instruction manual and affective ratings," Technical Report C-1, The Center for Research in Psychophysiology, University of Florida, Tech. Rep., 1999.

[44] J. Redondo, I. Fraga, I. Padrón, and M. Comesaña, "The Spanish adaptation of ANEW (Affective Norms for English Words)," *Behavior research methods*, vol. 39, no. 3, pp. 600–605, 2007.

[45] S. M. Mohammad, "# Emotional tweets," in *Proc. 1st Joint conf. on Lexical and Computational Semantics*. Montréal, Canada: Association for Computational Linguistics, 2012, pp. 246–255.

[46] V. Andryushechkin, I. D. Wood, , and J. O'Niel, "BLSTM and SVR ensemble for WASSA-2017 Shared Task on Emotion Intensity (EmoInt)," in *Proc. Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, Copenhagen, Denmark, 2017.

[47] S. M. Mohammad and F. Bravo-Marquez, "WASSA-2017 shared task on emotion intensity," in *Proc. Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, Copenhagen, Denmark, 2017.

[48] A. Graves, S. Fernández, and J. Schmidhuber, "Bidirectional LSTM networks for improved phoneme classification and recognition," in *Artificial Neural Networks: Formal Models and Their Applications*. Springer, Berlin, Heidelberg, 2005, pp. 799–804.

[49] S. Negi, K. Asooja, S. Mehrotra, and P. Buitelaar, "A study of suggestions in opinionated texts and their automatic detection," in *Proc. 5th Joint Conf. on Lexical and Computational Semantics*. Berlin, Germany: Association for Computational Linguistics, Aug 2016, pp. 170–178.

[50] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives, "DBpedia: A nucleus for a web of open data," in *The Semantic Web*. Springer, 2007, pp. 722–735.

[51] C. Strapparava and R. Mihalcea, "SemEval-2007 task 14: Affective text," in *Proc. 4th Int. Workshop on Semantic Evaluations*. Association for Computational Linguistics, 2007, pp. 70–74.

[52] M. M. Bradley and P. J. Lang, "Affective norms for English text (ANET): affective ratings of texts and instruction manual," University of Florida, Gainesville, Fl., USA, Tech. Rep., 2007.

[53] D. Preotiuc-Pietro, H. A. Schwartz, G. Park, J. C. Eichstaedt, M. Kern, L. Ungar, and E. P. Shulman, "Modelling valence and arousal in Facebook posts," in *Proc. 15th Annual Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, San Diego, CA, USA, 2016, pp. 9–15.

[54] S. Buechel and U. Hahn, "EMOBANK: Studying the impact of annotation perspective and representation format on dimensional emotion analysis," *European Chapter of the Association for Computational Linguistics*, p. 578, 2017.

[55] L.-C. Yu, L.-H. Lee, S. Hao, J. Wang, Y. He, J. Hu, K. R. Lai, and X. Zhang, "Building Chinese affective resources in valence-arousal dimensions," in *Proc. 15th Annual Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, San Diego, CA, USA, 2016, pp. 540–545.

[56] I. D. Wood, J. P. McCrae, V. Andryushechkin, and P. Buitelaar, "A comparison of emotion annotation schemes and a new annotated data set," in *11th edition of the Language Resources and Evaluation Conf.*, Miyazaki, Japan, 2018.

[57] I. Wood and S. Ruder, "Emoji as emotion tags for tweets," in *Emotion and Sentiment Analysis Workshop, at 10th edition of the Language Resources and Evaluation Conf.*, Portorož, Slovenia, 2016.

[58] C. Fellbaum, *WordNet: An Electronic Lexical Database*. Bradford Books, 1998.

[59] P. Vossen, Ed., *EuroWordNet: A Multilingual Database with Lexical Semantic Networks*. Norwell, MA, USA: Kluwer Academic Publishers, 1998.

[60] D. Tufiş, D. Cristea, and S. Stamou, "Balkanet: Aims, methods, results and perspectives. a general overview," *Romanian J. of Information science and technology*, vol. 7, no. 1-2, pp. 9–43, 2004.

[61] E. Pianta, L. Bentivogli, and C. Girardi, "MultiWordNet: developing an aligned multilingual database," in *Proc. 1st Int. Conf. on Global WordNet*, Mysore, India, Jan 2002.

[62] M. Schmitt, F. Ringeval, and B. Schuller, "At the border of acoustics and linguistics: Bag-of-Audio-Words for the recognition of emotions in speech," in *Proc. 17th Annual Conf. of the International Speech Communication Association*. San Francisco, CA: ISCA, Sep 2016, pp. 495–499.

[63] F. Ringeval, A. Sonderegger, J. Sauer, and D. Lalanne, "Introducing the RECOLA multimodal corpus of remote collaborative and affective interactions," in *Proc. Int. Workshop on Emotion Representation, Analysis and Synthesis in Continuous Time and Space*, Shanghai, China, 2013, 8 pages.

[64] B. Schuller, *Intelligent Audio Analysis*, ser. Signals and Communication Technology. Springer, 2013.

[65] B. Schuller, A. E.-D. Mousa, and V. Vasileios, "Sentiment analysis and opinion mining: On optimal parameters and performances," *WIREs Data Mining and Knowledge Discovery*, vol. 5, pp. 255–263, Sep/Oct 2015.

[66] F. Eyben, F. Weninger, F. Groß, and B. Schuller, "Recent developments in openSMILE, the munich open-source multimedia feature extractor," in *Proc. 21st Int. Conf. on Multimedia*. Barcelona, Spain: ACM, Oct 2013, pp. 835–838.

[67] S. Rawat, P. F. Schulam, S. Burger, D. Ding, Y. Wang, and F. Metze, "Robust audio-codebooks for large-scale event detection in consumer videos," in *Proc. 14th Int. Speech Communication Association*. Lyon, France: ISCA, 2013, pp. 2929–2933.

[68] M. Schmitt and B. W. Schuller, "openXBOW - Introducing the Passau Open-Source Crossmodal Bag-of-Words Toolkit," *CoRR*, vol. abs/1605.06778, 2016.

[69] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "Liblinear: A library for large linear classification," *The J. of Machine Learning Research*, vol. 9, pp. 1871–1874, 2008.

[70] L. I.-K. Lin, "A concordance correlation coefficient to evaluate reproducibility," *Biometrics*, vol. 45, no. 1, pp. 255–268, 1989.

[71] F. Ringeval, B. Schuller, M. Valstar, J. Gratch, R. Cowie, S. Scherer, S. Mozgai, N. Cummins, M. Schmitt, and M. Pantic, "Avec 2017: Real-life depression, and affect recognition workshop and challenge," in *Proc. 7th Annual Workshop on Audio/Visual Emotion Challenge*. ACM, 2017, pp. 3–9.

[72] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE trans. on pattern analysis and machine intelligence*, vol. 32, no. 9, pp. 1627–45, Sep 2010.

[73] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in *Proc. British Machine Vision Conference*, 2015, p. 6.

[74] I. Kemelmacher-Shlizerman, S. M. Seitz, D. Miller, and E. Brossard, "The MegaFace benchmark: 1 million faces for recognition at scale," in *Proc. 59th Conf. on Computer Vision and Pattern Recognition*. Las Vegas, NV, USA: IEEE, 2016.

[75] V. Kazemi and J. Sullivan, "One millisecond face alignment with an ensemble of regression trees," in *Proc. 57th Conf. on Computer Vision and Pattern Recognition*. Columbus, OH, USA: IEEE, 2014.

[76] A. Pavelková, A. Herout, and K. Behún, "Usability of pilot's gaze in aeronautic cockpit for safer aircraft," in *Proc. 18th Int. Conf. on Intelligent Transportation Systems*. IEEE, Sep 2015, pp. 1545–1550.

[77] M. Kostinger, P. Wohlhart, P. M. Roth, and H. Bischof, "Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization," in *Proc. Int. Conf. on Computer Vision Workshops*, Barcelona, Spain, 2011, pp. 2144–2151.

[78] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," *arXiv preprint arXiv:1611.08050*, Nov 2016.

[79] I. J. Goodfellow, D. Erhan, P. L. Carrier, A. Courville, M. Mirza, B. Hamner, W. Cukierski, Y. Tang, D. Thaler, D.-H. Lee, and et al., "Challenges in representation learning: A report on three machine learning contests," *Neural Networks*, vol. 64, pp. 59–63, 2015.

[80] T. Baltrušaitis, P. Robinson, and L. P. Morency, "OpenFace: An open source facial behavior analysis toolkit," in *Proc. Winter Conf. on Applications of Computer Vision*. Lake Placid, NY, USA: IEEE, Mar 2016, pp. 1–10.

[81] E. Wood, T. Baltrušaitis, X. Zhang, Y. Sugano, P. Robinson, and A. Bulling, "Rendering of eyes for eye-shape registration and gaze estimation," in *Proc. Int. Conf. on Computer Vision*. Santiago, Chile: IEEE, 2015.

[82] G. McKeown, M. F. Valstar, R. Cowie, and M. Pantic, "The SEMAINE corpus of emotionally coloured character interactions," in *Proc. Int. Conf. on Multimedia and Expo*. Singapore: IEEE, Jul 2010, pp. 1079–1084.

[83] S. M. Mavadati, M. H. Mahoor, K. Bartlett, P. Trinh, and J. F. Cohn, "DISFA: A spontaneous facial action intensity database," *IEEE Trans. on Affective Computing*, vol. 4, no. 2, pp. 151–160, Apr 2013.

[84] X. Zhang, L. Yin, J. F. Cohn, S. Canavan, M. Reale, A. Horowitz, P. Liu, and J. M. Girard, "BP4D-Spontaneous: a high-resolution spontaneous 3d dynamic facial expression database," *Image and Vision Computing*, vol. 32, no. 10, pp. 692–706, 2014.

[85] M. Valstar, J. Gratch, B. Schuller, F. Ringeval, D. Lalanne, M. Torres Torres, S. Scherer, G. Stratou, R. Cowie, and M. Pantic, "AVEC 2016: Depression, mood, and emotion recognition workshop and challenge," in *Proc. 6th Int. Workshop on Audio/Visual Emotion Challenge*. Amsterdam, The Netherlands: ACM, 2016, pp. 3–10.

[86] C. Bizer, T. Heath, and T. Berners-Lee, "Linked data-the story so far," *Semantic Services, Interoperability and Web Applications: Emerging Concepts*, pp. 205–227, 2009.

[87] A. Westerski, C. A. Iglesias, and F. Tapia, "Linked opinions: Describing sentiments on the structured web of data," in *Proc. 4th Int. Workshop on Social Data on the Web*. CEUR, Oct 2011, pp. 21–32.

[88] J. F. Sánchez-Rada and C. A. Iglesias, "Onyx: A linked data approach to emotion representation," *Information Processing & Management*, vol. 52, no. 1, pp. 99–114, 2016.

[89] M. Schröder, P. Baggia, F. Burkhardt, C. Pelachaud, C. Peter, and E. Zovato, "EmotionML – an upcoming standard for representing emotions and related states," in *Affective Computing and Intelligent Interaction*, ser. Lecture Notes in Computer Science, S. D'Mello, A. Graesser, B. Schuller, and J.-C. Martin, Eds. Springer Berlin Heidelberg, 2011, vol. 6974, pp. 316–325.

[90] S. Hellmann, J. Lehmann, S. Auer, and M. Brümmer, "Integrating NLP using linked data," in *The Semantic Web*. Springer, 2013, pp. 98–113.

[91] E. Wilde and M. Duerst, "URI fragment identifiers for the text/plain media type," Internet Engineering Task Force, Apr. 2008.

[92] J. F. Sánchez-Rada, C. A. Iglesias, and R. Gil, "A Linked Data Model for Multimodal Sentiment and Emotion Analysis," in *Proc. 4th Workshop on Linked Data in Linguistics*, Beijing, China, Jul 2015, pp. 11–19.

[93] J. F. Sánchez-Rada, C. A. Iglesias, H. Sagha, B. Schuller, I. Wood, and P. Buitelaar, "Multimodal Multimodel Emotion Analysis as Linked Data," in *Proceedings of ACII 2017*, San Antonio, Texas, USA, October 2017.

[94] M. Lanthaler and C. Gütl, "On using JSON-LD to create evolvable RESTful services," in *Proc. 3rd Int. Workshop on RESTful Design*. Lyon, France: ACM, 2012, pp. 25–32.

[95] J. F. Sánchez-Rada and C. A. Iglesias, "Senpy: A pragmatic linked sentiment analysis framework," in *Proc. Special Track on Emotion and Sentiment in Intelligent Systems and Big Social Data Analysis*, Oct 2016.

[96] R. Bakker, "Knowledge graphs : Representation and structuring of scientific knowledge," Ph.D. dissertation, University of Twente, Enschede, The Netherlands, 1987.

[97] J. Sowa, *Conceptual structures: Information processing in mind and machine*. Addison-Wesley Pub.,Reading, MA, Jan 1983.

[98] C. Hoede, "Modelling knowledge in electronic study books," *J. of Computer Assisted Learning*, vol. 10, no. 2, pp. 104–112, 1994.

[99] A. Singhal, "Introducing the knowledge graph: things, not strings," *Official google blog*, 2012.

[100] B. Pang and L. Lee, "Opinion mining and sentiment analysis," *Foundations and trends in information retrieval*, vol. 2, no. 1-2, pp. 1–135, 2008.

[101] A. D. Kramer, J. E. Guillory, and J. T. Hancock, "Experimental evidence of massive-scale emotional contagion through social networks," *Proc. the National Academy of Sciences*, pp. 8788–8790, 2014.

[102] J. Tang, Y. Chang, and H. Liu, "Mining social media with social theories: A survey," *SIGKDD Explorations Newsletter*, vol. 15, no. 2, pp. 20–29, Jun 2014.

[103] C. Tan, L. Lee, J. Tang, L. Jiang, M. Zhou, and P. Li, "User-level sentiment analysis incorporating social networks," in *Proc. 17th SIGKDD Int. conf. on Knowledge Discovery and Data Mining*. ACM, 2011, pp. 1397–1405.

[104] W. Deitrick and W. Hu, "Mutually enhancing community detection and sentiment analysis on twitter networks," *J. of Data Analysis and Information Processing*, vol. 1, no. 3, pp. 19–29, 2013.

[105] J. Yang and J. Leskovec, "Patterns of temporal variation in online media," in *Proc. 4th int. conf. on Web search and data mining*. Kowloon, Hong Kong: ACM, 2011, pp. 177–186.

[106] Y. Artzi, P. Pantel, and M. Gamon, "Predicting responses to microblog posts," in *Proc. conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2012, pp. 602–606.

[107] S. Alhabash and A. R. McAlister, "Redefining virality in less broad strokes: Predicting viral behavioral intentions from motivations and uses of Facebook and Twitter," *New Media & Society*, vol. 17, no. 8, pp. 1317–1339, 2015.

[108] J. Cheng, L. Adamic, P. A. Dow, J. M. Kleinberg, and J. Leskovec, "Can cascades be predicted?" in *Proc. 23rd Int. conf. on World Wide Web*. New York, NY, USA: ACM, 2014, pp. 925–936.

[109] M. De Domenico, A. Lima, P. Mougel, and M. Musolesi, "The anatomy of a scientific rumor," *Scientific Reports*, vol. 3, 2013.

[110] M. Speriosu, N. Sudan, S. Upadhyay, and J. Baldridge, "Twitter polarity classification with label propagation over lexical links and the follower graph," in *Proc. conf. on Empirical Methods in Natural Language Processing*. Edinburgh, UK: Association for Computational Linguistics, 2011, pp. 53–56.

[111] X. Hu, L. Tang, J. Tang, and H. Liu, "Exploiting social relations for sentiment analysis in microblogging," in *Proc. 6th Int. conf. on Web Search and Data Mining*. New York, NY, USA: ACM, 2013, pp. 537–546.

[112] M. B. Oliver and A. A. Raney, "Entertainment as pleasurable and meaningful: Identifying hedonic and eudaimonic motivations for entertainment consumption," *J. of Communication*, vol. 61, no. 5, pp. 984–1004, 2011.

[113] R. J. Lewis, R. Tamborini, and R. Weber, "Testing a dual-process model of media enjoyment and appreciation," *J. of Communication*, vol. 64, no. 3, pp. 397–416, 2014.

[114] H. Sagha, M. Schmitt, F. Povolny, A. Giefer, and B. Schuller, "Predicting the popularity of a talk-show based on the emotionality of its speech content," in *Proc. 3rd Int. Workshop on Affective Social Multimedia Computing at 18th Annual Conf. of the Int. Speech Communication Association*. Stockholm, Sweden: ISCA, Aug 2017, p. 5.

3.2.4 Senpy: A Pragmatic Linked Sentiment Analysis Framework

Title	Senpy: A Pragmatic Linked Sentiment Analysis Framework
Authors	Sánchez-Rada, J. Fernando and Iglesias, Carlos A. and Corcuera-Platas, Ignacio and Araque, Oscar
Proceedings	Proceedings DSAA 2016 Special Track on Emotion and Sentiment in Intelligent Systems and Big Social Data Analysis (SentISData)
ISBN	
Year	2016
Keywords	emotion analysis, framework, sentiment analysis
Pages	735–742
Online	http://ieeexplore.ieee.org/abstract/document/7796961/
Abstract	Sentiment and emotion analysis technologies have quickly gained momentum in industry and academia. This popularity has spawned a myriad of service and tools. Due to the lack of common interfaces and models, each of these services imposes specific interfaces and representation models. Heterogeneity makes it costly to integrate different services, evaluate them or switch between them. This work aims to remedy heterogeneity by providing an extensible framework and an API aligned with the NLP Interchange Format service specification. It also includes a reference implementation, a first step towards a successful and cost-effective adoption. The specific contributions in this paper are: (i) the Senpy framework; (ii) an architecture for the framework that follows a plug-in approach; (iii) a reference open source implementation of the architecture; (iv) the use and validation of the framework and architecture in a big data sentiment analysis European project. Our aim is to foster the development of a new generation of emotion aware services by isolating the development of new algorithms from the representation of results and the deployment of services.

2016 IEEE International Conference on Data Science and Advanced Analytics

Senpy: A Pragmatic Linked Sentiment Analysis Framework

J. Fernando Sánchez-Rada, Carlos A. Iglesias, Ignacio Corcuera and Óscar Araque

Intelligent Systems Group
Universidad Politécnica de Madrid
{jfernando,cif}@dit.upm.es, {ignacio.cplatas,oscar.aiborra}@alumnos.upm.es

***Abstract*—Sentiment and emotion analysis technologies have quickly gained momentum in industry and academia. This popularity has spawned a myriad of service and tools. Due to the lack of common interfaces and models, each of these services imposes specific interfaces and representation models. Heterogeneity makes it costly to integrate different services, evaluate them or switch between them. This work aims to remedy heterogeneity by providing an extensible framework and an API aligned with the NLP Interchange Format service specification. It also includes a reference implementation, a first step towards a successful and cost-effective adoption. The specific contributions in this paper are: (i) the Senpy framework; (ii) an architecture for the framework that follows a plug-in approach; (iii) a reference open source implementation of the architecture; (iv) the use and validation of the framework and architecture in a big data sentiment analysis European project. Our aim is to foster the development of a new generation of emotion aware services by isolating the development of new algorithms from the representation of results and the deployment of services.**

Keywords*—*sentiment analysis; emotion analysis; framework; linked data;

I. Introduction

Sentiment analysis is a blooming field of research and application, fueled by the popularity of social media and the need to make sense of collective opinions [1]. A vast number of sentiment analysis tools and services have emerged in recent years. Most of these tools and services use ad-hoc representation and schemas. This heterogeneity not only prevents reusing tools, but it also hinders the establishment of common terminology and models. Initiatives like NLP Interchange Format (NIF) [2] paved the way to standardization by publishing a semantic format and an API for NLP services. Thence, applications like the NIF combinator [3] appeared, demonstrating that a semantic format eases the integration of different services. Other works have applied this notion to multimodal sentiment analysis by extending NIF with existing and new ontologies [4]. The new ontologies for emotion representation enable a better and unambiguous annotation, as well as other interesting applications such as automatic mapping of emotions between different models (e.g. from Plutchik's categories to the Valence-Arousal-Dominance space). However, the concepts behind ontologies and linked data publishing are unfamiliar to both the linguistic community and developers. As a consequence, there are still few solutions in the field that use semantic technologies. This is a known problem that motivated the creation of JSON-LD [5].

The contributions of this paper are: (i) Senpy, a generic framework for NLP services based on the vocabularies NIF, Marl and Onyx; (ii) the architecture of a service in this framework; (iii) the reference implementation of the Senpy architecture, which follows a plug-in architecture and demonstrates the practical feasibility of the framework [6], as well as several plugins for custom algorithms and wrappers of popular services; (iv) the extensive use of the reference implementation in a big data sentiment in the context of a big data sentiment analysis platform and other research projects.

The ultimate goal of this work is to ease the adoption of the proposed linked data model for sentiment and emotion analysis services, so that services from different providers become interoperable. With this aim, the design of the reference implementation has focused on its extensibility and reusability. A modular approach allows organizations to replace individual components with custom ones developed in-house. Furthermore, organizations can benefit from reusing prepackaged modules that provide advanced functionalities, such as algorithms for sentiment and emotion analysis, linked data publication or emotion and sentiment mapping between different providers.

The rest of this paper is structured as follows. Section II introduces the main concepts behind Senpy and its linked data approach. Section III describes the architecture of the Senpy framework. Section IV describes the reference implementation of the framework. Section V illustrates how this architecture and existing tools can be used to develop and use an emotion analysis service; Lastly, Section VI presents our conclusions and future work.

II. Background

A key aspect of Senpy is its linked data approach. Its model is based on the following specifications:

- Marl [7], a vocabulary designed to annotate and describe subjective opinions expressed on the web or in information systems
- Onyx [8], which is built on the same principles as Marl to annotate and describe emotions, and provides interoperability with Emotion Markup Language (EmotionML) [9]
- NIF 2.0 [2], which defines a semantic format and API for improving interoperability among natural language

978-1-5090-5206-6/16 $31.00 © 2016 IEEE
DOI 10.1109/DSAA.2016.79

processing services NIF follows a linked data principled approach so that different tools or services can annotate a text. To this end, texts are converted to RDF literals and an URI is generated so that annotations can be defined for that text in a linked data way. NIF offers different URI Schemes to identify text fragments inside a string, e.g. a scheme based on RFC5147 [10], and a custom scheme based on context. In addition to the format itself, NIF 2.0 defines a REST API for Natural Language Processing (NLP) services with standardized parameters.

The integration of these ontologies has been covered in previous works [4]. For the sake of clarity, Listing 1 provides an example of annotation by a sentiment analysis service. In particular, it consists of the analysis of a microblog post with the text "The example they used was really good, I really enjoyed it" and whose URL is `http://microblog.com/User1/Post1`. The service response shown in Listing 1 indicates that an opinion (`:Opinion1`) has been detected. The properties of the entity are shown as well. Finally, it provides details of the analysis, such as the algorithm used, its confidence, polarity range and provenance (using the PROV-O ontology [11]). Note that a query to the service has the following format: `http://{endpoint}?i={text}&prefix={prefix}`.

```
<http://microblog.com/User1/Post1#char=0,49>
  rdf:type nif:RDF5147String, nif:Context;
  nif:beginIndex "0";
  nif:endIndex "75";
  nif:isString "The example they used in their last paper
      ↪ was very clear, I really liked it";
  marl:hasOpinion :Opinion1.
:Opinion1
  rdf:type marl:Opinion;
  marl:describesObject "paper";
  marl:describesObjectPart "example";
  marl:describesFeature  "clarity";
  marl:polarityValue "0.8";
  marl:hasPolarity: marl:Positive;
  prov:wasGeneratedBy :Analysis1.

:Analysis1
  rdf:type marl:SentimentAnalysis;
  marl:maxPolarityValue "1";
  marl:minPolarityValue "-1";
  marl:algorithm "dictionary-based";
  prov:wasAssociatedWith http://www.gsi.dit.upm.es/.
```

Listing 1: NIF + Marl output of a service call http://senpy.cluster.gsi.dit.upm.es/api?i=The example they used in their last paper was very clear, I really liked it&prefix=http://microblog.com/User1/Post1#

III. Framework

This section describes a framework for natural language processing services, with a special focus on sentiment and emotion analysis.

The main component of a sentiment analysis service is the algorithm itself. However, for the algorithm to work, it needs to get the appropriate parameters from the user, format the results according to the defined API, interact with the user when errors occur or more information is needed, etc. All this boilerplate of sorts, albeit essential for the service, is a burden on service developers. The situation is even worse when dealing with different algorithms at the same time, which usually requires developing and deploying them separately. For this reason, Senpy proposes a modular and dynamic architecture that allows: i) implementing different algorithms in an extensible way, yet offering a common interface, ii) offering common services that facilitate development, so developers can focus on implementing new and better algorithms. Furthermore, it fosters the creation of common tools such as service validators, evaluation suites and testing tools.

The framework covers all the aspects of developing, publishing and using a sentiment analysis service. These aspects are grouped into layers. In addition to giving a clearer view of the components of a service, separating the framework in aspects serves another purpose: it later helps with transferring this modularity to its implementations. Finally, modular implementation fosters the creation of new services and functionalities by reducing the cost of adding new features and algorithms.

As of this writing, we have identified five different layers: the Analysis Layer includes the core NLP process and the libraries to connect it to the rest of the layers; the Semantic Layer deals with conceptual models and their integration; the Syntactic Layer handles issues such as formatting, serialization and input/output validation; the User Interface (UI) is the way in which users interact with services; the Evaluation Layer allows users to benchmark different algorithms; and the Service Administration Layer offers tools and information to control running services. Figure 1 depicts these layers and the main components within each of them. The rest of this section describe each layer in detail.

The Analysis Layer includes those components that are directly involved in generating new annotations for a given input. More specifically, it comprises the implemented analysis algorithms and the libraries used in the implementation that are responsible for integrating one particular algorithm with the rest of the components. For instance, a specific service may include one or more sentiment analysis algorithms to choose from, a Named Entity Recognition algorithm, a gender detection algorithm, etc. Each of these algorithms should be developed independently from the rest, and should contain only the logic that concerns the generation of new annotations. The interface between every algorithm and the rest of the layers is well defined. The set of libraries that implement this interface are the Senpy SDK, which is also part of the framework.

The Semantic Layer provides semantic consistency to the service and adapts the results from the Analysis Layer to every request. To exemplify the role of this layer, let us consider the case of an emotion analysis service. The Analysis Layer of this service would consist of at least one implementation of an emotion analysis algorithm. This algorithm generates annotations using Ekman's six categories. In a traditional service, this would mean that the output of the service could only contain these categories. If an application requires a different representation, such as the VAD (valence, arousal, dominance) dimensional model, the conversion of the results is external to the analysis service. In a Senpy service, the Semantic Layer could include mappings to transparently adapt the annotations to the desired representation. In addition to mappings and conversion, the Semantic Layer could include other steps, such as validation and inference.

The lowest level of abstraction corresponds to the Syntactic Layer. Its role is to validate and adapt input from and output to the user. On the input side, it extracts all the necessary information from every request, and processess it so that it is understood by other layers. If there is an error in the request, such as missing parameters or wrong syntax, this layer communicates it to the user. When the output from other layers is ready to be sent back to the user, the Syntactic Layer formats it using the appropriate is to validate both the input and the output of the service, to process the input so that it can be understood by other layers, and to process the output so that it has the requested format and structure.

The User Interface (UI) Layer handles the interactions between users and the service. The way in which users make requests and receive the results back is different depending on the medium used. For instance, the same underlying analysis could be accessed through a Command Line Interface (CLI) and a web service (Web UI). This difference should be transparent to developers. Hence, the main task of the UI Layer is to gather requests from the user, forward them to the rest of the framework, and then adapt the output to the medium in use. Another element in this layer is the Playground aspect, which will be explained further in Section IV. The main idea behind it is that users want to experiment with new services before integrating them in their workflow or using them programmatically. The Playground is a simple UI that presents users with all available algorithms and options, and guides them through their use.

All previous layers cover functional aspects, i.e. developing a service and allowing users to make requests. The last two layers in this section cover aspects that do not concern users but developers and service administrators.

A key aspect of developing a new analysis algorithm is to evaluate it and compare it to others. The Evaluation Layer contains benchmarking and evaluation tools. Evaluation is facilitated by the fact that the framework imposes a common API. i.e., services of the same type will use the same annotation scheme and will be called in the same way. Using the common API and a set of gold standard corpora, it is possible to evaluate and compare different algorithms. The same concept applies to testing.

Lastly, the Service Administration Layer includes aspects useful to maintain a service and control its lifecycle. Some of its main functions would be: logging, which is used to control execution and find possible errors; resource manager, to control processing, memory and storage consumption; usage statistics, for an overview on how the service is being used; process monitoring, to control what tasks are running and when; and configuration manager, to view and change the parameters used in the service, such as activating or deactivating modules within the service.

IV. Reference Implementation

Providing a reference implementation of the conceptual framework serves three main purposes. Firstly, it allows us to assess the feasibility and completeness of the framework. Secondly, it acts as a showcase of the purpose and the concepts behind the framework. Thirdly, it can be used as a reference or gold standard for future implementations.

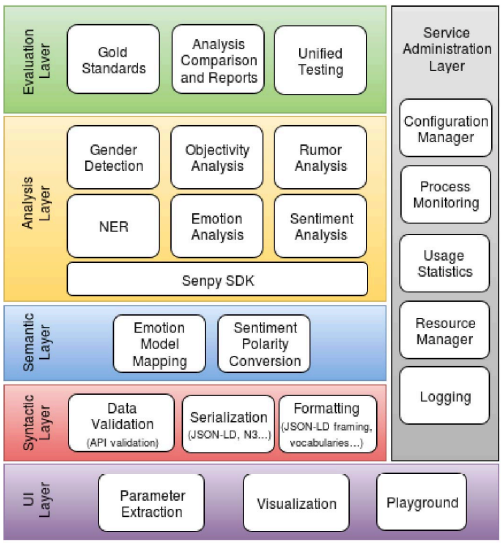


Fig. 1: Senpy framework. Each layer represents a functional block in a service.

The architecture of the reference implementation consists of two main modules: Senpy core, which is the building block of the service, and Senpy plugins, which contain the code for each analysis algorithm. The modularity of the architecture serves the overall goal of Senpy of providing seamless integration of different analysis algorithms while minimizing code duplication and development effort. Several plugins may coexist in the same service, accessing different resources and algorithms while benefiting from the nurturing environment of the common platform. Figure 2 depicts a simplified version of the processes involved in an analysis with Senpy. The following sections describe each component of the architecture in further detail.

The implementation is fully Open Source and published on GitHub[1], and a live demo is publicly available[2].

A. Core

As its name implies, the core of Senpy provides the main functionalities of the platform: an HTTP server/CLI interface, parameter extraction and validation, serialization of results using different formats and an abstraction and publication of results as Linked Data. It manages the lifecycle of plugins as well, orchestrating their execution and all interaction with the user. To better understand the features of the core, let us follow a typical analysis request from a user.

First of all, there are two ways in which a user may want to run their analysis: as a one-off local process or as a service. For

[1] http://www.github.com/gsi-upm/senpy
[2] http://senpy.cluster.gsi.dit.upm.es

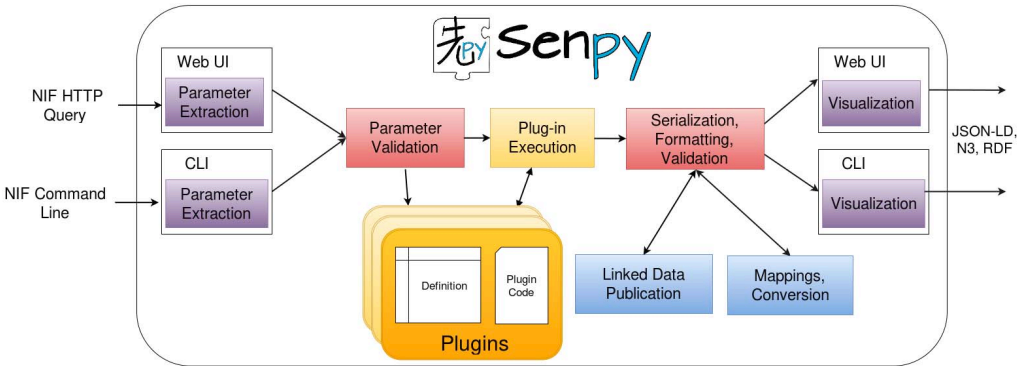


Fig. 2: Modules involved in an analysis with the reference implementation of Senpy

one-off commands, Senpy provides a command line interface (CLI), configurable via arguments. For long running processes or services, Senpy provides an HTTP server. In this case, users send their requests using HTTP queries to the server. Both the CLI and HTTP server use an API aligned with NIF, but using a JSON-LD representation and a JSON-schema by default. This difference makes it friendlier and more appealing to developers, as well as compatible with a wider range of tools. The API defines the parameters that are allowed (Table I), and is complemented by the extra parameters that each plugin declares in its definition (see Listing 2 for an example).

parameter	description
input(i)	serialized data (i.e. the text or other formats, depends on informat)
informat(f)	format in which the input is provided: turtle, text (default) or json-ld
outformat(o)	format in which the output is serialized: turtle (default), text or json-ld
prefix(f)	prefix used to create and parse URIs
emodel(e)	emotion model in which the output is serialized (e.g. WordNet-Affect, PAD, etc.)
minpolarity (min)	minimum polarity value of the sentiment analysis
maxpolarity (max)	maximum polarity value of the sentiment analysis
language (l)	language of the sentiment or emotion analysis
domain (d)	domain of the sentiment or emotion analysis
algorithm (a)	plugin that should be used for this analysis

TABLE I: Parameters of an Emotion or Sentiment analysis service using Senpy

Senpy uses these parameters in every request to extract all parameters from the request, and to warn the user whenever there are missing parameters.

If the basic arguments provided are correct, Senpy uses its selection algorithm to determine the plugin that will receive the request. Typically, users select the plugin manually using the `algorithm` parameter. Senpy will then check if the extra parameters defined in the selected plugin are met as well. If this validation succeeds, the plugin is asked to run an analysis, using the validated parameters.

Senpy leverages different ontologies (e.g. Marl, Onyx) to represent different types of information. For simplicity, the main types of results as well as their required and optional properties have been defined using JSON schema. This means that results are provided in a documented format that can also be validated before passing them to the user. Plugins use these models to return their results.

Once the analysis is done, its results are further modified before they are returned to the user. First of all, values are transformed to fit the parameters specified by the user. For instance, when a plugin uses a sentiment value in the interval (-1, 1), and the user requested a value in the (0,) interval. This phase is very useful when dealing with emotions. Senpy has several mappings from dimensional models to categories, and vice versa. An example of this can be seen in Section V.

Lastly, Senpy generates the final results in the appropriate format, including metadata and proper URI identifiers, so it can be published as Linked Data.

For convenience, Senpy includes a web interface to test all available plugins: the Senpy Playground (Figure 3). The Playground lists all available services, and dynamically adds fields for every parameter they accept, such as language.

B. Plugins

The components in the Analysis Framework from Figure 1 are plugins in the implementation. Hence, each plugin represents a different analysis process. For instance, we may have a plugin for emotion analysis using WordNet-Affect, and a plugin for sentiment analysis using SVM and the Sentiment140 corpus. In future versions of the implementation we plan to extend plugins to also cover components in other layers of the architecture.

A plugin is defined by two elements: a definition file and the plugin code. The definition file can be written in JSON or YAML (a JSON superset), and has the `.senpy` extension. It contains important information about the plugin such as: name, version, location of the plugin code, parameters needed and attributes of the plugin. Listing 2 shows the description file of an example plugin, which we will use in Section V. In this description we see that the plugin accepts an extra parameter

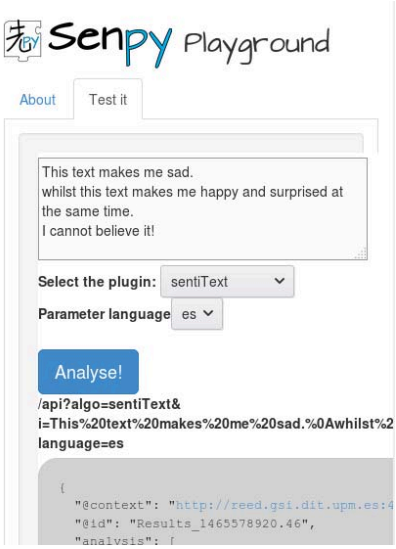


Fig. 3: Senpy Playground web interface.

in requests, `language`. When not provided, this parameter defaults to `en`. It also contains an attribute specific to this plugin, `default_value`, which determines the default value for words that are not found in ANEW.

Listing 2: Plugin definition using YAML

```
---
name: EmoTextANEW
module: emotextANEW
version: "0.1"
description: "Emotion classifier using rule-based
    ↪ classification."
extra_params:
language:
aliases:
- language
- l
default: en
options:
- es
- en
required: true
default_value: [0,0,0]
```

The `module` attribute indicates the module that will be loaded. In this case, that module corresponds to the code in Listing 3. A Senpy (Senpy) plugin has to implement three methods: `activate`, for allocation of resources; `deactivate` for their release; and `analyse`, which takes the user-supplied parameters and performs the analysis. Resource allocation may seem needlessly complicated, but it is an important process when dealing with models that take gigabytes of memory. Section V covers the creation of a this specific plugin in more detail.

There are three main states in the lifecycle of a plugin: unloaded, inactive and active. Only active plugins can be used in requests. For a plugin to be active, two things have to happen. First, the core has to load it. Once a plugin has been loaded, it gets in the inactive state. In this state, a plugin is listed by the core, but the variables necessary for analysis may not have been initialized. When a plugin is loaded, a special method in the plugin is called that initializes these variables. If the activation process is successful, the plugin enters the active state and can be used by users. If there are errors during the activation, the plugin remains inactive and all errors are logged. Active plugins can also be deactivated, which puts them in the inactive state again and should free up any variables that were initialized during activation.

To exemplify this process, let us consider the case of a sentiment analysis that uses a naive bayes classifier. This plugin requires a trained classifier to analyze text. However, when the plugin is loaded the classifier is not ready yet. The classifier is trained upon activation. Training may take a long time, depending on the size of our corpus and the features used. For this reason, changes of state are asynchronous operations for the core. When the activation finishes, our plugin will be automatically marked as active. Meanwhile, the core may handle other requests. Once our plugin is active, we can use it to analyze text. When our plugin is no longer needed, we may deactivate it. Deactivation will free up the memory used by the trained model.

Releasing resources when a plugin is not needed means that many resource hungry plugins can be loaded at the same time, and only activate them when they are needed. Resource initialization during activation also means that plugin variables will be consistent after it is activated. On the other hand, it also means that costly operations, such as training a model, have to be repeated several times. To avoid repetition and speed up start-up time considerably, Senpy ships with a special type of plugins that provide persistence. These persistent plugins have access to special variables that can be used to store the results of costly operations. When these variables are used, the plugin automatically checks the filesystem for a saved version of the variable. If if does, it loads the variable. If not, the plugin runs the appropriate operation and stores the value of the result in the filesystem.

The reference implementation Senpy has been validated by implementing wrappers to several available sentiment and emotion services, such as Sentiment 140 [12], MeaningCloud [13], Cogito [14], Vader [15] and Paradigma [16].

V. Use Case

In this section we briefly cover the process of using Senpy in a real scenario. Our use case is a Big Data platform that uses a series of NLP services on social media. In fact, the scenario is a simplification of one of the pilots in the MixedEmotions project. This platform is made up of several modules from different parties. Some of them are existing NLP and sentiment analysis services. The rest of the modules depend on one or more of these analysis services. Integrating the different modules and their interfaces would require a big effort from every parties involved. Senpy reduces the cost of integration with its common interface and tools.

Figure 4 depicts the main elements. There are two parts in the platform of this use case. Firstly, there is a live brand monitoring dashboard. The dashboard shows the opinions of

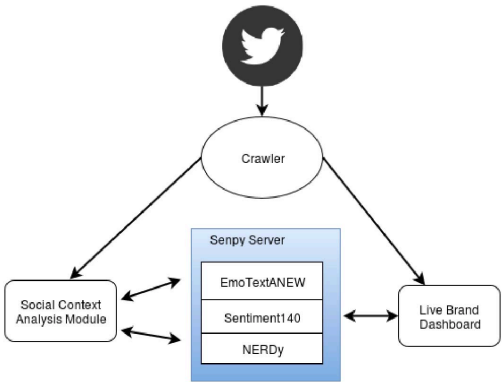


Fig. 4: Using Senpy in a Big Data Sentiment Analysis Platform

social media users about a brand. For this, it uses an external sentiment analysis service (sentiment140 [3]), to annotate social media content with opinions. Secondly, there is a social context analysis module that finds the most influential users and content, as well as the evoluation of emotion of all relevant users. The social context analysis module uses a NER (named entity recognition) module to gather only relevant content, and an emotion analysis module to annotate the emotions in the content. Social media content is provided by a separated module, labeled Crawler in Figure 4. A server running senpy will provide the NER, Sentiment and Emotion Analysis Analysis services.

Instead of accessing the external sentiment analysis service directly, we choose to use a custom sentiment analysis service in senpy that acts as a wrapper/proxy to the actual service. The main advantage of this approach is that it avoids having to deal with more than one API and schema for NLP service. Additionally, we gain access to all the extra capabilities of Senpy, such as the polarity conversion, benchmarking and service evaluation. Developing the wrapper is very straightforward and requires very little code [4].

The emotion analysis analysis relies on the ANEW [17] lexicon to analyze the emotion in text, using a simple bag-of-words approach. Turning this code into a Senpy plugin is trivial, and merely a matter of implementing the `analyse` method. We have already covered the description file in Listing 2. The accompanying code is shown in Listing 3.

Listing 3: Code for the EmoTextANEW plugin

```
from os.path import dirname, abspath
from senpy.plugins import SenpyPlugin, tokenize
from senpy.models import Results, EmotionSet,
                         Emotion, VAD
from emotextanew import Analyser

class EmoTextANEW(SenpyPlugin):

    def activate(self, *args, **kwargs):
        self._local_path = dirname(abspath(__file__))
        self._analyser = Analyser(self._local_path)

    def deactivate(self, *args, **kwargs):
        del self._analyser

    def analyse(self, params):
        r = Results.from_params(params)
        for i in r.entries:
            es = EmotionSet()
            e = Emotion()
            valence = 0
            arousal = 0
            dominance = 0
            for j in i.nif__isString:
                # Find V,A,D for each word
                v, a, d = self._analyser.get_vad(j)
                valence += v
                arousal += a
                dominance += d

            e[VAD.arousal] = a
            e[VAD.valence] = v
            e[VAD.dominance] = d
            es.onyx__hasEmotion.append(e)
            i.emotions = es
        return results
```

[3] http://www.sentiment140.com/

[4] https://github.com/gsi-upm/senpy/tree/master/senpy/plugins/sentiment140

Since ANEW uses the VAD emotion model, that is what our plugin will use as well. Normally, this would mean users would need to use the VAD model themselves. However, since we are using Senpy to publish our service, we can make use of its additional features, such as mapping of emotion models. Emotion mapping can be used by setting the `emodel` parameter in the request. Listing 4 shows the response to a request using the WordNet-Affect model. Notice the addition of the emotion category (joy) based on the VAD dimensions. In particular, the conversion from VAD to WordNet-Affect categories is based on centroids [18]. The information about the centroids is displayed in the results, which together with the use of the provenance ontology makes the process transparent and repeatable.

Listing 4: Requesting an emotion analysis with a different emotion than the one provided in the plugin.

```
{
  "@context": "http://senpy.cluster.gsi.dit.upm.es/api/
      ↪ contexts/context.jsonld",
  ]
  "analysis": [
  {
 "analysis": [
   {
     "@id": "EmoTextANEW_0.1",
     "@type": "onyx:EmotionAnalysis",
     "onyx:usesEmotionModel": "onyx-anew:ANEWModel"
   },
   {
     "@id": "ANEW_Mappings_0.1",
     "@type": "onyx:EmotionAnalysis",
     "onyx:usesEmotionModel": "wnaffect:WNAModel"
     "centroids": {
       "wnaffect:anger": {
         "A": 6.95,
         "D": 5.1,
         "V": 2.7
       },
       "wnaffect:disgust": {
         "A": 5.3,
         "D": 8.05,
         "V": 2.7
       },
       "wnaffect:fear": {
         "A": 6.5,
         "D": 3.6,
         "V": 3.2
       },
       "wnaffect:joy": {
         "A": 7.22,
```

```
          "D": 6.28,
          "V": 8.6
        },
        "wnaffect:sadness": {
          "A": 5.21,
          "D": 2.82,
          "V": 2.21
        }
      }
    }
  ],
  "entries": [
  {
    "emotions": [
    {
      "onyx:hasEmotion": [
      {
        "prov:wasGeneratedBy": "EmoTextANEW_0.1",
        "onyx-anew:arousal": 5.0,
        "onyx-anew:dominance": 5.62,
        "onyx-anew:valence": 6.23,
      },{
        "prov:wasGeneratedBy": "ANEW_Mappings_0.1",
        "onyx:hasEmotionCategory": "wn-affect:joy"
      }]
    }
    ],
    "language": "en",
    "nif:isString": "I am feeling excited"
  }
 ]
}
```


This section illustrates how easy it is to develop a new service from scratch and to integrate it in a bigger scenario. As shown in Listing 3, a plugin code is made up entirely of the `analyse` function, which almost perfectly matches the pseudocode of the algorithm. This succinct code provides a web service and a CLI tool that does parameter extraction and validation automatically. Furthermore, the platform provides additional features such as automatic linked data conversion and publication or mapping of emotions. Finally, the common interfaces and schemas provide loose coupling to the platform. This means that once a module is adapted for senpy to make use of a service, it can be made to use any equivalent service just by pointing it to a different endpoint.

VI. Conclusions and future work

The sentiment and emotion analysis community would highly benefit from a common framework for service and language resources. Such a framework would ease adoption, development, integration and evaluation of services. A linked data approach further adds to these benefits, but its use needs to be transparent to users and developers.

This paper proposes a generic framework that combines both worlds and a reference implementation of that framework that is currently being used in MixedEmotions[5], a European R&D project. The linked data model for sentiment and emotion services is based on the combination of NIF, Marl and Onyx vocabularies. Moreover, a number of parameters (e.g. min, max and e) have been defined following NIF Service specification so that sentiment and emotion service calls are interoperable.

We want to increase the adoption of the framework and to foster a community approach, where most plugins and features are provided by third parties. For this reason, we are currently working on easing the development of new plugins, and on making it possible to create plugins for any part of the framework. Other lines of research would be the connection between different plugins (e.g. pipelining), the integration of the framework with other distributed and big data systems, and the addition of authentication and rate limiting.

In conclusion, the framework proposed in this paper has already proven useful in a multilingual sentiment analysis scenario. It has enabled the integration of multiple services from different parties and eased the creation of novel algorithms. This new approach paves the way for new testing and validation tools, as well as advanced capabilities such as deployment in high availability and cluster environments.

[5] http://mixedemotions-project.eu/

VII. Acknowledgements

This work has been partially funded by the European Union - Horizon 2020 (Industrial Leadership) through the MixedEmotions project (number H2020 644632). Part of this work has been carried out in the context of the MOSI-AGIL-CM research programme (grant S2013/ICE-3019, supported by the Autonomous Region of Madrid and co-funded by EU Structural Funds FSE and FEDER, and the SEMOLA project, funded by the Ministry of Economy and Competitiveness of Spain (TEC2015-68284-R).

References

[1] B. Liu, "Sentiment analysis and opinion mining," *Synthesis Lectures on Human Language Technologies*, vol. 5, no. 1, pp. 1–167, 2012.

[2] S. Hellmann, J. Lehmann, S. Auer, and M. Brümmer, "Integrating nlp using linked data," in *The Semantic Web–ISWC 2013*. Springer, 2013, pp. 98–113.

[3] S. Hellmann, J. Lehmann, S. Auer, and M. Nitzschke, "Nif combinator: Combining nlp tool output," in *Knowledge Engineering and Knowledge Management*. Springer, 2012, pp. 446–449.

[4] J. F. Sánchez-Rada, C. A. Iglesias, and R. Gil, "A linked data model for multimodal sentiment and emotion analysis," *ACLIJCNLP 2015*, p. 11, 2015.

[5] M. Lanthaler and C. Gütl, "On using json-ld to create evolvable restful services," in *Proceedings of the Third International Workshop on RESTful Design*. ACM, 2012, pp. 25–32.

[6] E. Dalci, E. Fong, and A. Goldfine, "Requirements for gsc-is reference implementations. national institute of standards and technology," *Information Technology Laboratory*, 2003.

[7] A. Westerski, C. A. Iglesias, and F. Tapia, "Linked opinions: Describing sentiments on the structured web of data," in *Proceedings of the Fourth International Workshop on Social Data on the Web (SDoW2011)*. CEUR, Oct. 2011, pp. 21–32.

[8] J. F. Sánchez-Rada and C. A. Iglesias, "Onyx: A linked data approach to emotion representation," *Information Processing & Management*, vol. 52, no. 1, pp. 99–114, 2016.

[9] M. Schröder, P. Baggia, F. Burkhardt, C. Pelachaud, C. Peter, and E. Zovato, "Emotionml – an upcoming standard for representing emotions and related states," in *Affective Computing and Intelligent Interaction*, ser. Lecture Notes in Computer Science, S. D'Mello, A. Graesser, B. Schuller, and J.-C. Martin, Eds. Springer Berlin Heidelberg, 2011, vol. 6974, pp. 316–325.

[10] E. Wilde and M. Duerst, "URI Fragment Identifiers for the text/plain Media Type," Internet Engineering Task Force, Apr. 2008.

[11] P. Missier, K. Belhajjame, and J. Cheney, "The w3c prov family of specifications for modelling provenance metadata," in *Proceedings of the 16th International Conference on Extending Database Technology*. ACM, 2013, pp. 773–776.

[12] P. Chikersal, S. Poria, and E. Cambria, "Sentu: sentiment analysis of tweets by combining a rule-based classifier with supervised learning," in *Proceedings of the International Workshop on Semantic Evaluation, SemEval*, 2015, pp. 647–651.

[13] I. Segura-Bedmar, P. Martínez, R. Revert, and J. Moreno-Schneider, "Exploring spanish health social media for detecting drug effects," *BMC medical informatics and decision making*, vol. 15, no. Suppl 2, p. S6, 2015.

[14] Expert System, *COGITO Intelligence Platform*, 2015.

[15] C. J. Hutto and E. Gilbert, "Vader: A parsimonious rule-based model for sentiment analysis of social media text," in *Eighth International AAAI Conference on Weblogs and Social Media*, 2014.

[16] J. F. Sánchez-Rada, M. Torres, C. A. Iglesias, R. Maestre, and R. Peinado, "A Linked Data Approach to Sentiment and Emotion Analysis of Twitter in the Financial Domain," in *Second International Workshop on Finance and Economics on the Semantic Web (FEOSW 2014)*, vol. 1240, May 2014, pp. 51–62.

[17] M. M. Bradley and P. J. Lang, "Affective norms for english words (ANEW): Instruction manual and affective ratings," Technical Report C-1, The Center for Research in Psychophysiology, University of Florida, Tech. Rep., 1999.

[18] S. M. Kim, A. Valitutti, and R. A. Calvo, "Evaluation of unsupervised emotion models to textual affect recognition," in *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*. Association for Computational Linguistics, 2010, pp. 62–70.

3.3 Social context for Sentiment Analysis

3.3.1 Social context in sentiment analysis: Formal definition, overview of current trends and framework for comparison

Title	Social context in sentiment analysis: Formal definition, overview of current trends and framework for comparison
Authors	Sánchez-Rada, J. Fernando and Iglesias, Carlos A.
Journal	Information Fusion
Impact factor	JCR 2018 Q1 (10.716)
ISSN	1566-2535
Publisher	
Volume	52
Year	2019
Keywords	Sentiment analysis, social context
Pages	344–356
Online	https://www.sciencedirect.com/science/article/pii/S1566253518308704
Abstract	Sentiment analysis in social media is harder than in other types of text due to limitations such as abbreviations, jargon, and references to existing content or concepts. Nevertheless, social media provides more information beyond text, such as linked media, user reactions, and relations between users. We refer to this information as social context. Recent works have successfully leveraged the fusion of text with social context for sentiment analysis tasks. However, these works are usually limited to specific aspects of social context, and there have not been any attempts to analyze and apply social context systematically. This work aims to bridge this gap by providing three main contributions: 1) a formal definition of social context; 2) a framework for classifying and comparing approaches that use social context; 3) a review of existing works based on the defined framework.

Social Context in Sentiment Analysis: Formal Definition, Overview of Current Trends and Framework for Comparison

J. Fernando Sánchez-Rada and Carlos A. Iglesias

Intelligent Systems Group,
Universidad Politécnica de Madrid.
{*jf.sanchez,carlosangel.iglesias*}*@upm.es*

Abstract

Sentiment analysis in social media is harder than in other types of text due to limitations such as abbreviations, jargon, and references to existing content or concepts. Nevertheless, social media provides more information beyond text, such as linked media, user reactions, and relations between users. We refer to this information as social context. Recent works have successfully leveraged the fusion of text with social context for sentiment analysis tasks. However, these works are usually limited to specific aspects of social context, and there have not been any attempts to analyze and apply social context systematically. This work aims to bridge this gap by providing three main contributions: 1) a formal definition of social context; 2) a framework for classifying and comparing approaches that use social context; 3) a review of existing works based on the defined framework.

Keywords: sentiment analysis, social context, social network analysis, online social networks

1. Introduction

Recent years have witnessed the rise of social media. Platforms such as Twitter or Facebook have become the de facto way to share thoughts and opinions with a wide audience [41]. Studies of Twitter usage show that about 19% of tweets contain a reference to a brand or product, 20% of which also show some expression of brand sentiment [39]. As a consequence, companies and researchers have grown interested in social media as a way to monitor public opinion. The sheer amount of social media content makes it impractical or impossible to manually process it. Hence, automatic sentiment analysis has grown very popular.

Sentiment analysis has been applied for many years in other types of opinionated content, such as online reviews or news articles. However, social media

content poses several unique challenges to natural language processing in general, and to sentiment analysis in particular [64]. Some of these challenges are imposed by the very nature of social media platforms, such as limited length and relying on associated media. Other difficulties are caused by the characteristics of human interaction in these types of media. e.g., short attention span, need for immediacy, and use of specialized language. The result is a type of text that is short, full of jargon or abbreviations, ephemeral, and rife with references to contextual information.

There are different approaches to sentiment analysis in social media [3, 71, 14]. Most techniques are content-centric. They exploit specific linguistic characteristics of social media, just like previous research has done for other media (e.g., news articles) and domains (e.g., movie reviews). Some works try to overcome abbreviations and short texts in social media by finding external sources to link text to, such as news articles [32] or Wikipedia pages [29]. Other works leverage the specific language in these media by finding cues for sentiment (e.g., smileys and hashtags) [21]. When the textual content is also accompanied by multimedia, such as images or videos, the sentiment information in these media obtained with multimodal analysis [69] may also be exploited.

Nevertheless, these approaches fail to use the fact that information shared on social networks is not isolated. The meaning of a particular piece of content (e.g., a Tweet, a Facebook status or a blog post) may only be understood when its context is taken into consideration. This context includes visible information such as previous content that belongs to the same conversation, previous interactions between users, or people that interacted with the content (e.g., by liking it). It also includes seemingly unrelated social features. For instance, some demographic factors such as age and gender have been shown to correlate with sentiment and vocabulary [89], and they have been used to improve sentiment classification [37].

New sentiment analysis techniques are starting to incorporate the fusion of information from text and social context. Social context has also been introduced in other fields related to sentiment analysis, such as spam detection, where clues to identify spammers are usually hidden in multiple aspects of context, such as previous content, behavior, relationship, and interaction [15]. Unfortunately, the definition of social features, the methods employed to extract them, and how they are applied to sentiment analysis tasks vary greatly from work to work. These differences in notation and approaches are taxing, which makes comparing different works harder.

Thus, further research is needed to delve more deeply into the notion of social context and the fusion of social context with traditional textual sentiment analysis. This work seeks to answer the following questions:

- Q1. What is social context?
- Q2. Can social context improve sentiment analysis?
- Q3. What elements of social context are more relevant for sentiment analysis purposes?

As a result, the contributions herein are threefold. First, this work proposes a formal and general definition of social context. Secondly, a framework to compare existing works in the field is proposed. In this framework, each work is described using a multi-level taxonomy that classifies each approach in terms of the proposed definition of social context, and other factors such as the machine learning techniques applied. Thirdly, the state of the art in sentiment analysis using social context is organized and compared using the defined framework. Moreover, the results reported by each work in the analysis have been aggregated and analyzed, to simplify the comparison of approaches.

The remaining of this paper is structured as follows. Section 2 presents an overview of the state of the art in sentiment analysis prior to social context, and an introduction to social network analysis; Section 3 introduces a formal definition of social context; Section 4 presents the framework for comparison of approaches to sentiment analysis using social context; Section 5 provides an overview of the state of the art, using the framework presented in the previous section; Lastly, Section 6 discusses the main conclusions drawn from this work and future lines of research.

2. Related Work

This section is overview of relevant work in the fields of sentiment analysis and social network analysis. Each field is discussed in a separate section. The former discusses different approaches in sentiment analysis, including deep learning and ensemble techniques. The latter introduces Social Network Analysis (SNA), and it focuses on community detection due to its importance in several of the works reviewed.

2.1. Sentiment Analysis

Although sentiment analysis has been an active research topic for decades, it has grown in popularity with the advent of online opinion-rich resources [64]. In turn, these resources have also added their own set of limitations and challenges.

Over the last two decades, numerous works have explored sentiment analysis in different applications and using different approaches. These approaches can be grouped into machine learning, lexicon based, and hybrid [71]. Of the three, machine learning techniques and hybrid approaches seem to be dominant [3, 65, 90], and lexicon techniques are typically incorporated into machine learning approaches to improve their results. Machine learning approaches apply a predictor (a classifier, or an estimator) on a set of features that represent the input. The set of predictors is not very different from those used in other areas. Instead, the complexity in these approaches lies in extracting complex features from the text, filtering only relevant features, and selecting a good predictor [78].

One of the most straightforward features is the Bag Of Words (BOW) model. In BOW, each document is represented by the multiset (bag) of its constituent words. Word order is disrupted, and syntactic structures are broken. As a

result, a great deal of information from natural language is lost [94]. Therefore, various types of features have been exploited, such as higher order n-grams [63]. A more sophisticated feature is Part of Speech (POS) tagging [30]. In it, a syntactic analysis process is run, and each word is labeled (tagged) with its syntactic function (e.g., noun). Additionally, syntactic trees can be calculated. Using these trees, the words in the input can be rearranged to a more convenient position while still conveying the same meaning. Note how these two types of features only rely on lexical and syntactical information. For this reason, they are sometimes referred to as surface forms.

Surface forms can also be combined with other prior information, such as word sentiment polarity [28, 11, 44, 54, 57]. This prior knowledge usually takes the form of sentiment lexicons, i.e., dictionaries that associate words in a domain or language with a sentiment. Some lexicons also include non-words such as emoticons [40, 36] and emoji [60]. These alternative forms of writing have been shown very useful, as they can dominate textual cues and form a good proxy for text polarity [36].

The use of lexicon-based techniques has many advantages [82], most of which stem from their combination with other methods. For instance, it is possible to generate lexicons that are domain dependent or that incorporate language-dependent characteristics. Lexicons and syntactic information can also be combined with linguistic context to shift valence [68]. On the other hand, there are several disadvantages to lexicon approaches. First, creating lexicons is an arduous task, as it needs to be consistent and reliable [82]. It also needs to account for valence variability across domains, contexts, and languages. These dependencies make it hard to maintain domain-independent lexicons. An alternative to retain independence while encoding domain, language, and context variability is through semantic representation of the lexical resources in the form of ontologies. An ontology can encode both lexical [52] and affective [81] nuances, both in the lexicons and in the automatic annotations [9]. This is especially useful for aspect-based sentiment analysis, as the differences between aspects can be incorporated into the ontology [91].

In recent years, new approaches based on deep learning have shown excellent performance in Sentiment Analysis [19, 5]. In contrast with traditional techniques, deep learning techniques learn complex features from data with minimum human interaction. These algorithms do not need to be passed manually crafted features: they automatically learn new complex features. The downside is that the quality of the features heavily depends on the size of the training data set. Hence, they often require large amounts of data, which is not always available. They also raise other concerns such as interpretability [51, 49] or its inability to adapt to deal with edge cases [51]. In the realm of Natural Language Processing (NLP), most of the focus is on learning fixed-length word vector representations using neural language models [42]. These representations, also known as word embeddings, can then be fed into a deep learning classifier, or used with more traditional methods. One of the most popular approaches in this area is word2vec [55]. The downside of these methods is that they require enormous amounts of training data. Luckily, several researchers have already

applied these methods to large corpora such as Wikipedia and released the resulting embeddings.

Lastly, it is also possible to combine independent predictors to achieve a more accurate and reliable model than any of the predictors on their own. This approach is known as ensemble learning. Many ensemble methods have been previously used for sentiment analysis. Ensemble methods can be classified according to two main dimensions Rokach [73]: how predictions are combined (rule-based and meta-learning), and how the learning process is done (concurrent and sequential). A new application of ensemble methods is the combination of traditional classifiers based on feature selection and deep learning approaches [3].

2.2. Social Network Analysis and Community Detection

Social Network Analysis (SNA) is the investigation of social structures [62]. It provides techniques to characterize and study the connections between people, and their interactions. SNA is not limited to Online Social Network (OSN), but to any kind of social structure. Other examples of social network would be a network of citations in publications or a network of relatives. Through SNA techniques, it is possible to extract information from a social network that may be useful for sentiment analysis, such as chains of influence between users, groups of like-minded users, or metrics of user importance.

There are several ways in which SNA techniques can be exploited in sentiment analysis, but most of them fall under one of two categories: those that transform the network into metrics or features that can be used to inform a classifier; and those that limit the analysis to certain groups or partitions of the network.

A simple example of metrics provided by SNA could be user's follower in-degree (number of users that follow the user) and out-degree (number of users followed by the user), which could be used as features for each user [79]. However, these metrics are not very rich, as they only cover users directly connected to a user, and it does so in a very naive way: all connections are treated equally. Other more sophisticated metrics could be used instead of in/out-degree, such as centrality, a measure of the importance of a node within a network topology, or PageRank, an iterative algorithm that weights connections by the importance of the originating user. Several works have introduced alternative metrics for user and content influence in a network [33, 59].

The second category of approaches is what is known either as network partition or as community detection, depending on whether the groupings may overlap. Intuitively, community detection aims to find subgroups within a larger group. This grouping can be used to inform a classifier, or to limit the analysis to relevant groups only. More precisely, community detection identifies groups of vertices that are more densely connected to each other than to the rest of the network [66]. The motivation is to reduce the network into smaller parts that still retain some of the features of the bigger network. These communities may be formed due to different factors, depending on the type of link used to connect users, and the technique used to detect the communities. Each definition has

its own set of characteristics and shortcomings. For instance, if users are connected after messaging each other, community detection may reveal groups of users that communicate with each other often [22]. By using friendship relations, community detection may also provide the groups of contacts of a user [25].

The reader is referred to other publications [66, 61] for further details of the different definitions of community and algorithms to detect them.

3. A Definition of Social Context

This section introduces a novel definition of social context and its components. The definition is focused on OSN aspects, and it is based on previous definitions and on the observed usage of social context features in the state of the art.

Since the inception of Twitter and its API in late 2006, several works have used social features to complement text [6]. This section aims to introduce a general definition of social context that both encompasses existing definitions and formalizes the loose or implicit definitions used in most works.

To the best of our knowledge, the first formal definition of social context was introduced by Lu et al. [50]. They defined the social context of a set of Reviews R as *the triple* $C(R) = \langle U, A, S \rangle$*, of the set of reviewers* U*, the authorship function* A*, and the social network relation* S. Although their work is focused on reviews, it identifies the three main entities of this social context: the content (*review*), the content producer (*the author*) and the user-relations (*the social network relations*). Later works have also referred to social context in different terms [93, 58], but a formal definition is seldom provided. For instance, Ren and Wu [72] define both Social Context and Topical Context, based on the graph of relations and their adjacency matrix. Namely, Social Context is defined as $G_S = \{u, S\}$, where u is the set of users and S is the adjacency matrix between users, and Topical Context is defined as $G_t = \{t, T\}$, where t is the set of topics, and T is the adjacency matrix of topics.

Based on these definitions, and our analysis of the state of the art, we have identified four types of elements that make up Social Context (Fig. 1): content (C), users (U), relations (R) and interactions (I). These elements are related as follows.

Users are connected through relations and interactions. Relations are stable connections between two or more users (R^u). There are multiple types of relations, such as friendship, or belonging to the same group. Some types of relations are undirected or mutual, like kinship, whereas others are directed or asymmetrical, such as liking and following relations. Interactions appear when a user communicates with others (I^u). The types of interactions include direct messages, replies, and user mentions. Most of these types also involve the creation of content. When a user creates or posts new content, an authorship relation between the user and the content is formed (R^{uc}). New content may also be related to existing content (e.g., as a reply or a mention, R^c), or to other users (e.g., the user is mentioned in the content, R^{uc}). Users may then interact with the newly created content (I^{uc}), by replying to it, liking it, saving it, etc.

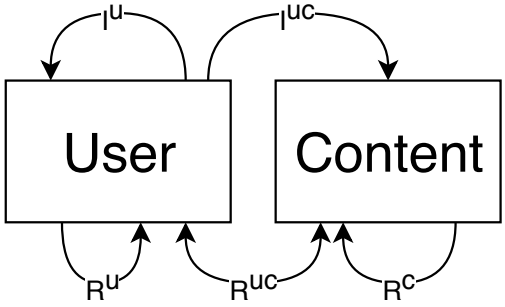


Figure 1: Model of Social Context, including: content (C), users (U), relations (R^c, R^u and R^{uc}), and interactions (I^u and I^{uc}).

All elements are rich entities with different attributes. The specific attributes that can be used depend on the type of element and the OSN. Content attributes (e.g., text, creation date) and user attributes (e.g., name, age, gender) are commonly used. Although interaction and relation attributes are not as widespread, they are also important. They provide information such as when the interaction happened, or the weight of the relation. These attributes make it possible to filter specific connections, and to apply algorithms that rely on weighted graphs.

An additional concept to take into account is temporal dependence. New content is continuously created, and existing content is changed or removed. Relations are similar, as they are forged and dissolved naturally; and users can join, delete their accounts or become inactive. The relevance of social context variation over time is illustrated in Section 4.3 with the introduction of dynamic approaches.

These ideas about the elements of Social Context and their dynamic nature are condensed in the following definitions. First, Definition 1 covers Social Context as a whole and establishes its constituent elements.

Definition 1. *Social Context is the collection of users, content, relations, and interactions which describe the environment in which social activity takes place. Namely:*

$$SocialContext(\tau) = \langle C, U, R, I \rangle(\tau) = \langle C(\tau), U(\tau), R(\tau), I(\tau) \rangle$$

At any point in time τ: $C(\tau)$ is the set of content (Definition 2) generated by these users; $U(\tau)$ is the set of users (Definition 3); $I(\tau)$ is the set of interactions (Definition 5) between users, and of users with content; $R(\tau)$ is the set of relations (Definition 4) between users, between pieces of content, and between users and content.

This is a very general definition which only sets up the main elements, and it relies on the definition of each element to fully characterize context. To simplify

the notation in the remaining definitions, time dependence will be implicit from here on: $SocialContext = \langle C, U, R, I \rangle$. This can be done without loss of generality. Whenever time dependence is relevant, we will refer to time-dependent social context as dynamic social context and to time-independent social context as static social context.

To illustrate the definitions, we will model an example of social context for a sentiment analysis task on Facebook content. For this analysis, we only need access to status updates by some users, and photos uploaded to a set of Facebook pages (groups).

The first element in social context is content:

Definition 2. *The collection of content is defined as:*

$$C = \{c_{t,i} \mid t \in T_c\} \tag{1}$$

Where T_c are all the types of content available, and each $c_{t,i}$ is a piece of content of a certain type t. Each piece of content should be unambiguously identified by its type and an identifier (i).

Our example context only includes two types of contents: status updates and photos. Each type of content may be given some attributes. Some of these attributes are common, such as the creation date. Others are specific for that type, such as the keywords for status updates, and the link to the image file for photos. Additionally, each photo and each status has to be given an identifier, which may also be the one given by the Facebook API. So far, the context defined is not very useful, as it would only allow us to analyze the sentiment of the status updates and the photos (using other modalities).

The next element in Social Context is the collection of users in the network.

Definition 3. *Let the set of users be:*

$$U = \{u_1, u_2, \ldots, u_n\} \tag{2}$$

Where each u_i is a specific user that is unambiguously identified by its user identifier i. Each user may have one or more roles. The set of roles for a user is:

$$\rho(u_i) = \{t \mid \rho_t(u_i) = 1, u_i \in U, t \in T_\rho\} \tag{3}$$

Where T_ρ are all possible roles in a context, and $\rho_t(u_i)$ is a function that determines whether user u_i has been assigned role t.

Roles define the function of users within the network. They usually restrict the type of interactions and relations a user may have, and with what content and users. e.g., online fora have the role of topic moderators, in addition to regular users. The aim of moderators is to decide what content should be allowed, to edit it, and to manage users that misbehave. Hence, new relations

(e.g., edited-by) and interactions (e.g., ban) are available to this specific role. If the user is a moderator of more than one topic, several roles will apply.

Our example context will include the profiles of the users in our study and their attributes. Since we are only interested in age and location, users will just have those attributes. Our users may also have roles. In our case, we will be interested in page administrators. At this point, the lack of connection between users and content hampers other types of analysis.

The categorization of connections in Social Context is based on the concept of social ties in the social sciences, i.e., dyadic relations [8]. Social ties are grouped into one of four categories: similarities, such as co-location or being the same gender; social relations, such as kinship (e.g., family ties), role (e.g., friendship), or affection (e.g., liking); interactions, such as having talked to each other, or harming one another; and flows, such as sharing information, beliefs, or resources. For the sake of simplicity, and based on the use of context in the state of the art, only two types of connections are modeled as part of Social Context: relations (Definition 4) and interactions (Definition 5). The remaining social ties (similarities and flows) can be modeled as an equivalent relation or interaction, depending on the case. Similarities are not typically considered as ties in themselves but rather as conditions or states that increase the probability of forming other kinds of ties. Flows are typically inferred from interactional and relational data [8] so, for the sake of simplicity, they can be thought of as another type of relation or interaction.

Hence, relations are connections such as friendship, kinship, group membership or liking each other, whereas interactions are connections such as getting in touch, re-sharing each other's content, etc. There are two main differences between relations and interactions that motivate their distinction. First, relations are few and slow-changing, whereas interactions are plentiful and short-lived. Secondly, content can be related to other content (e.g., a reply and the original content), while interactions are always performed by a user agent.

Formally, relations and interactions are defined as follows:

Definition 4. *Given a set of content C, and a set of users U. Relations are the connections between users (R^u), between users and content (R^{uc}) and between different content (R^c). Formally:*

$$R \equiv \{r_t \mid t \in T_r\} = R^u \cup R^{uc} \cup R^c \tag{4}$$

$$R^u_t = \{r^u_{t,u_i,u_j} \mid u_i, u_j \in U, u_i \neq u_j, t \in T_{r,u}\} \tag{5}$$

$$R^{uc}_t = \{r^{uc}_{t,u_i,c_j} \mid u_i \in U, c_j \in C, t \in T_{r,uc}\} \tag{6}$$

$$R^c_t = \{r^c_{t,c_i,c_j} \mid c_i, c_j \in C, c_i \neq c_j, t \in T_{r,c}\} \tag{7}$$

Where $T_{r,c}$ are the types of relations between two pieces of content, $T_{r,uc}$ are the types of relations between users and content, and $T_{r,u}$ are the types of relations between users.

Definition 5. *Given a set of content C, and a set of users U. Interactions are the activities carried on by a user that involve either another user (I^u), or a piece of content (I^{uc}). Formally:*

$$I \equiv \{i_t \mid t \in Ti\} = I^u \cup I^{uc} \tag{8}$$

$$I^u_t = \{i^u_{t,u_i,u_j,i} \mid u_i, u_j \in U, t \in T_{i,u}\} \tag{9}$$

$$I^{uc}_t = \{i^{uc}_{t,u_i,u_j,i} \mid u_i \in U, c_j \in C, t \in T_{i,uc}\} \tag{10}$$

Where $T_{i,uc}$ are the types of interactions between user and content, $T_{i,u}$ are the types of interactions between users, and i is an identifier for the interactions, as multiple interactions of the same type are possible.

With all elements defined, we can go back to the previous example of Social Context on Facebook. From the possible types of relations between users (R^u), we may add two: user friendship and kinship. These two relations would allow us to group users that are closely related. To link users with content, we will choose two types of user-content relations (R^{uc}): authorship, and mentions (i.e., the link between the content and the users it mentions). As for relations between content (R^c), we may choose replies (i.e., the link between two pieces of content when one mentions the other). Lastly, we will only have access to interactions between users and content (I^{uc}) in the form of likes, reactions, and replies. Due to technical limitations, we will not have access to user interactions, such as direct messages.

The resulting example context would allow for richer analyses that exploit information such as inferred groups of people based on how often they interact with each other or appear in photos together. Sentiment analysis may exploit prior knowledge about the sentiment of the user (via the authorship relation), or even knowledge about the sentiment of friends and acquaintances (through either relations or interactions between users). It may even be possible to find people within the group that have changed the opinion of the people with whom they interact.

Table 1 shows other types of user, content, relations and interactions found in popular OSN. It includes common elements in the OSN analyzed in the state of the art: Twitter, Weibo, Reddit, Facebook, blogging platforms and Wikipedia.

The tabular format does not capture how different types of relations or interactions are unique to certain types of content and/or user roles. We will exemplify this fact using Facebook since it has different types of content and users roles. In Facebook, we may consider four main types of content. There are statuses, which are posts by users which are shown on their own profile (i.e., user feed). Statuses are very rich, they may mention other users, include location information, link to other content, or even express the mood of the author. The visibility of the status is governed by the user's privacy settings, and the relationship of the user to others. For instance, privacy-minded users

may make their statuses only available to their close friends, while other users may make theirs public. Similarly, users can create pages, which are public profiles created around a specific topic, such as a business, a brand, or a cause. Pages are similar to user profiles, but they can be administered by one or more users. Another type of content is photos, which may be linked to a user profile or to a page. Photos can include information about the users that appear in them, which creates a relation between the photo and the users. Events are a different type of content that is used to organize gatherings and to give information about them. Users may indicate whether they will attend, comment on the event, and invite other users to join.

Users may interact with content to which they have access in different ways: by liking it; by commenting to it, which creates new content that other users may interact with; or by expressing their reaction or emotion to it, such as surprise. These types of interaction are common for all types of content. Some types of content provide other means of interaction, such as re-sharing of posts, which allows users to share a post by other user in their own profiles.

The primary means for interaction between users is through content, either by interacting with the content, e.g., users may reply to each other's content, by including other users in their content, e.g., by adding a mention in a comment or a tag in a photo. Lastly, they may interact through special actions such as poking each other, or through private instant messages. Since these interactions are private, they have not been included in the table.

OSN	Content (T_c)	User roles (T_ρ)	Relations (T_r)			Interactions (T_i)	
			User-User ($T_{r,u}$)	User-Content ($T_{r,uc}$)	Content-Content ($t_{r,c}$)	User-User ($t_{i,u}$)	User-Content ($t_{i,uc}$)
Twitter	Tweet	User	Follow Friend	Author Mentioned Favorite	Reply Retweet	Mention Reply	Reply Retweet Mention
Weibo	Weibo	User	Follow Friend	Author Mentioned Favorite	Reply Reshare	Mention Reply	Reply Reshare
Reddit	Post Comment	User Admin	Follow	Author Mentioned	Link Reply	Mention Reply	Vote Gild Reply Mention
Facebook	Status Page Comment Photo Event	User Page admin	Friend Relative	Author Admin Fan Own Tagged Attend Like React	Link Reply Contain	Mention Reply Tag	Comment Re-share
Blog	Post Comment	Author Reader	Follow	Author Like	Link Reply	Mention Reply	Reshare Comment
Wiki	Page Comment	Editor Reviewer	-	Author Edit Review	Link Parent Reply	-	Edit

Table 1: Types of Social Context elements in different OSN.

Some researchers are concerned that the typical follower-friend relation might

not be enough to capture the richness of relations in online media [20]. They also propose researching into new multifaceted approaches which take into consideration more aspects of the network simultaneously. Social context has been intentionally defined with those approaches in mind. The definition of Social Context can be interpreted in the form of sets, or in its equivalent graph form, where users and content are vertices, and both relations and interactions are edges. The graph form can be combined with different types of links (T_c, T_u, T_r, T_i) to generate multiplex networks [27] (i.e. a multilayered network of users and content), which can be exploited in multifaceted approaches.

To conclude, the usage of the social network [43] and the effect of the social network on user behaviour [18] depend on other aspects such as cultural differences, factual information and events. This type of information falls outside the scope of social context, and will need to be encoded through other means such as a knowledge graph, or a description of events. However, social context will capture information such as language of a user or creation time of content, which can be used to link the user or content to that external information. This concept will be further explained in Sect. 4.2.

4. Framework for Research on Social Context in Sentiment Analysis

This section defines a novel framework to compare sentiment analysis approaches that exploit social context. The framework is centered around a multilevelled taxonomy for structuring research in the field. The first level refers to the dataset used. The second level covers the scope of Social Context built from the dataset. The third level covers machine learning methods applied. The fourth level covers the type of social context used (static and dynamic). Each level is further explained in a separate section.

4.1. Dataset

The datasets used for analyzing social context can be identified by several characteristics. The first of them is the online social network from which the data was gathered. Twitter predominates in this area, due to its relatively open API and abundance of content. The second characteristic is the type of annotation on content. Likewise, the third characteristic is the type of annotation on users. In this work, we focus on sentiment (polarity), but other annotations such as stance, emotion, and quality of the content are often used. In the case of polarity, the classes used may also differ. i.e. positive (+), negative (−) and neutral (0). The fourth, fifth, and sixth characteristics are the type of link between users, between pieces of content, and between users and content. These links can stem either from a relation or from an interaction, as mentioned in the definition of social context.

4.2. Context Scope

Researchers have to choose what information from their datasets to select for the social context in their work. They may also complement the original data

with information from external sources. As a consequence, every work employs a different context. Nonetheless, a closer inspection reveals some patterns: some elements are commonly used together (e.g., users and friendships), and some elements are harder to obtain or rarer than others (e.g., follower-followee relations are more common than retweets or favorites). As contexts get more and more complex, they start including more unusual elements in addition to the more basic ones.

Hence, we propose a classification of works based on the complexity or scope of their context. Our proposal is inspired by the micro, meso and macro levels of analysis typically used in social sciences [7]. The two differences are: 1) a level of analysis is added to account for analysis without social context, and 2) the meso level is further divided into three sub-levels ($meso_r$, $meso_i$, and $meso_e$), to better capture the nuances at the meso level. The result is shown in Fig. 2, and the levels are:

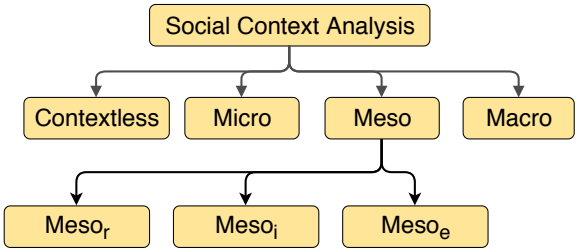


Figure 2: Taxonomy of approaches, and the elements of Social Context involved.

- Contextless: The approaches in this category do not use social context, and they rely solely on textual features.
- Micro: These approaches exploit the relation of content to its author(s), and may include other content by the same author. For instance, they may use the sentiment of previous posts [1] or other personal information such as gender and age to use a language model that better fits the user [88].
- Meso-relations ($Meso_r$): In this category, the elements from the micro category are used together with relations between users. This new information can be used to create a network of users. The slow-changing nature of relations makes the network very stable. The network can be used in two ways. First, to calculate user and content metrics, which can later be used as features in a classifier. e.g., a useful metric could be the ratio of positive neighboring users [1]. Second, the network can be actively used in the classification, with approaches such as label propagation [80].
- Meso-interactions ($Meso_i$): This category also models and utilizes interactions. Interactions can be used in conjunction with relations to create

a single network or be treated individually to obtain several independent networks. The resulting network is much richer than the previous category, but also subject to change. In contrast to relations, interactions are more varied and numerous. To prevent interactions from becoming noisy, they are typically filtered. For instance, two users may only be connected only when there have been a certain number of interactions between them.

- Meso-enriched ($Meso_e$): A natural step further from $Meso_i$, this category uses additional information inferred from the social network. A common technique in this area is community detection. Community partitions may inform a classifier, influence the features used for each instance [87], or be used to process groups of users differently [22]. Other examples would be metrics such as modularity and betweenness, which can be thought of as proxies for importance or influence. Some works have successfully explored the relationship between these metrics and user behavior, in order to model users. However, these results are seldom used in classification tasks.

- Macro: At this level, information from other sources outside the social network is incorporated. For instance, Li et al. [48] use public opposition of political candidates in combination with social theories to improve sentiment classification. Another example of external information is facts such as the population of a country, or current government, which can be combined with geo-location information in social media content. A more complex example would be events in the real world or in other types of media, such as television, which can be analyzed in combination with social media activity [34].

The six levels of approaches are listed in increasing order of detail, measured as the number of elements social context may include. The specific elements that are available at each level are represented in Fig. 3. The essential elements have already been covered in the definition of social context: content (C), users (U), relations (R^c, R^u and R^{uc}), and interactions (I^u and I^{uc}). Social Context can also be enriched through SNA with techniques such as community detection (CD). Additionally, external sources of information can be used at a macro level, such as facts or hyperlinks to external media, which are not part of the definition of Social Context.

4.3. Dynamic approaches

Social context can be represented and analyzed as static or dynamic, as mentioned in the definition. Static approaches present a quasi-static view of social context and do not take its evolution into account. Note that this does not prevent context from being updated at a later point. For instance, a user label may be changed, or more content may be added. However, these changes are not integrated into the model. In most of the works analyzed, context is modeled as static. Conversely, dynamic approaches both use and need a

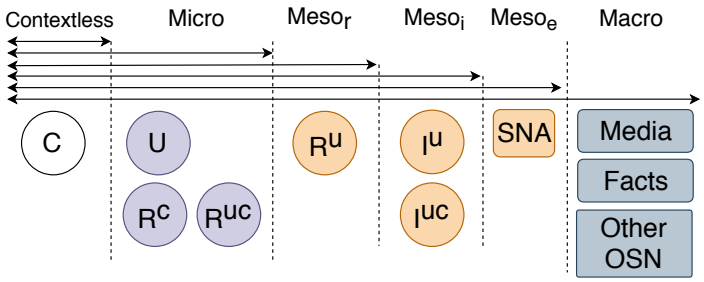


Figure 3: List of Social Context features available at each level of analysis

dynamic social context, as they exploit the changing nature of social networks. These changes are an intrinsic part of the analysis and need to be part of the model.

Although none of the surveyed works use dynamic social contexts for sentiment classification, several works use dynamic social context in tasks related to sentiment analysis. Based on those and related works, we suggest dynamic approaches for sentiment analysis may adhere to the following taxonomy, depending on the parts of social context that are dynamic.

At the *Micro-dynamic* level, content is dynamic, and the changes in its activity are taken into consideration. These changes could be the increase in some metrics such as retweets and likes. For instance, the evolution in content activity (number of retweets and mentions) can be used to classify content [96].

At the *Meso-dynamic* level, inter-personal communication starts to be apparent and available. Several elements of the context can be studied in a dynamic fashion. Two types of approaches could be considered, to subdivide this level.

First, approaches that focus on virality, and are content-centric. They use the evolution of interactions, and the links between users in the network, to measure and predict future activity, or to classify content according to the activity related to it. This classification may be useful for sentiment analysis. For instance, previous works have shown different types of content are linked to different temporal patterns [96]. And by using certain features of content and its activity, it is also possible to predict further spreading in the network (i.e., a cascade) [17]. These content cascades are also linked to specific sentiments [2]. Garas et al. [26] could be relevant in this area, as it studies emotion persistence in online communications (IRC).

Second, contagion-based approaches, which are user-centric. They focus on user sentiment and emotion, instead of content. They apply social theories and experimental results regarding sentiment and emotion contagion [35]. For instance, a massive experiment on Facebook showed that emotional states can be transferred to others via emotional contagion, leading people to experience the same emotions without their awareness [45]. Hence, it may be possible

to improve the prediction of a user's sentiment (and their content's) by using the sentiment of the content to which she is being exposed. On the other hand, studies of social media activity regarding grassroots movements have shown that social integration, as measured through social network metrics, increases with their level of engagement and of expression of negativity [2]. This suggests a connection between the groups to which a user belongs, and the sentiment the user expresses. The connection could be exploited for user classification and, in turn, for classification of the content created by them.

4.4. Analysis methods and Social Theories

Lastly, works differ in the type of classification performed. The options here range from using traditional classification algorithms (e.g., random forest, SVM) or neural networks, to network-based approaches such as label propagation. However, two types of algorithms stand out from those of contextless analysis: models that directly benefit from the networked nature of context, and deep learning approaches. Several works also use a hybrid approach, where traditional techniques are combined with network techniques, either via multiple processing steps or by combining the techniques into one.

There are several ways in which algorithms could leverage the networks in social context. Firstly, some algorithms are already network-oriented. Label propagation, in particular, has shown promising results [80], and it can be made to treat lexical resources and the subject of the analysis equally. Secondly, the structure of the network can be directly incorporated into the learning process through modified cost functions [38, 92]. Thirdly, the output of a classifier can be later complemented with a network-based algorithm. For example, Li et al. [48] apply standard classification, then tweets or users are clustered, and within each cluster, every piece of content or every user are given the same label according to different criteria (i.e., most confident result, majority label, and weighted majority). Fourthly, a multi-step or ensemble classification strategy can be used, where the structure of the network and social theories are used to combine the results of different classifiers.

On the deep learning front, recent works are incorporating different types of neural networks that have been used for contextless analysis and subjectivity analysis [14], such as convolutional neural networks (CNN). At the same time, concepts such as word embeddings have inspired network embedding as an alternative way of including features from social context in the analysis [97]. The range of features that can be captured through network embeddings is vast, including several types of relations [13]. Moreover, new research is complementing and extending node embedding (i.e., nodes are represented as vectors) with other methods such as edge and community embedding [10]. In particular, community embedding has shown promising results in community prediction and node classification [12].

In general, network approaches usually follow well-known social theories. Social theories usually model how users with different views or status arrange themselves in the network. In other words, they are rules of attachment. They may also model how users behave.

Some examples of social theories or attributes include homophily, consistency, social balance, and status theory. Homophily [53] is one of the commonly used theories in the works we have examined and in the social sciences. In simple terms, homophily means a connection between two people is more likely when they are similar in some aspects (i.e., birds of a feather flock together). Under the hypothesis of homophily, when two users are connected, certain features can be propagated. Consistency [50] usually means that users tend to maintain their views over time. So, two pieces of content shared by the same user in a short period are likely to express a similar sentiment or opinion if they are about the same topic. The social status theory [47] models the balance of power in social networks. It states that, if three nodes A, B and C form a clique, and the status relation between A and B is the same as between B and C, it must also be true of A and C. In other words, the superior of your superior is your superior, and the inferior of your inferior is your inferior. Social balance models the balance of opinions in cliques. The rules in social balance translate to: a friend of a friend is a friend, and an enemy of my enemy is my friend. Tang et al. [84] presents a more detailed explanation of social theories that can be used to mine social media.

5. Review of Social Context and Sentiment Analysis works

This section is the result of reviewing the state of the art in using social context for sentiment analysis. The review is composed of five subsections. The first one presents and compares the different works that have been reviewed. The second subsection describes and compares the datasets that have been used in these works. The third subsection covers common social context features that are useful for sentiment analysis. The fourth one presents a performance comparison of the works on different datasets. The last subsection discusses ways in which sentiment analysis has been used to improve social network analysis.

5.1. Works

This section introduces recent works in the area of sentiment analysis that use social context. The aim is to compare how social context is defined and exploited in each of them. The main features of each of the works are summarized in Table 2. The table shows the gradual introduction of interactions to complement interactions, as works evolve from $meso_r$ to $meso_i$ and $meso_e$ approaches. It also highlights the most commonly used types of elements and social theories used.

To the best of our knowledge, the first work to make explicit mention of social context in the context of sentiment analysis is Lu et al. [50]. Their goal was to predict the quality of reviews, rather than their sentiment, but the work is worth mentioning for three reasons. First of all, they provide the first formal mention of social context in the sense covered in this work. Secondly, their novelty is that they merge traditional features (text) with what they call *Social Network Features*. They provide a categorization of features, including author

Table 2: Comparison of works using sentiment analysis and social context. The number of polarity labels is shown in parentheses.

Reference	OSN	Level	l^u	l^c	$i^u.$	i^{uc}	r^c	$r^{u,c}$	r^u	Social Theories
Pennacchiotti and Popescu [67]	Twitter	$meso_i$	political orientation, ethnicity	polarity (3)		replies, retweets	retweet	authorship	friends	
Speriosu et al. [80]	Twitter	$meso_r$	polarity (2)	polarity (2)				authorship	follower	
Tan et al. [83]	Twitter	$meso_i$	polarity (2)	-	(mutual) mention			authorship	follower	consistency, homophily
Li et al. [48]	Twitter, Fora	$meso_r$, Macro	stance (targets)	polarity (2)	stance (targets)					balance, consistency
Aisopos et al. [1]	Twitter	$micro$, $meso_i$		polarity (2)	mention			authorship	follower	
Hu et al. [38]	Twitter	$meso_r$	polarity (3)	polarity (3)				authorship	follower	consistency and contagion
Pozzi et al. [70]	Twitter	$meso_i$	polarity (2)		retweet	retweet		authorship	mutual follower	
Ren and Wu [72]	Twitter	$meso_r$	polarity (2)							homophily
Deng et al. [23]	Fora	$meso_r$		polarity (3)			reply		friends, inferred friends	homophily, consistency
West et al. [92]	Wiki	$meso_i$	polarity (3)	polarity (3)	votes, mentions			authorship		social status, social balance
Yang and Eisenstein [97]	Twitter	$meso_i$		polarity (2)	retweet, mention		retweet		follow	language homophily
Cheng et al. [16]	Reddit	$meso_i$		polarity (2)		reply				
Sixto et al. [79]	Twitter	$meso_i$		polarity (5)		retweet		favorite	follow	
Xiaomei et al. [95]	Twitter	$meso_e$		polarity (2)				authorship	follow	emotion contagion

and social network features, which are calculated with social network analysis. Lastly, the network is used to extract constraints based on several hypotheses of consistency (of authors, links, citations, and trust).

On a related note, Pennacchiotti and Popescu [67] leverage replies, retweets and friendship relations to infer user attributes, such as ethnicity and political orientation. Their definition of political orientation can be considered stance detection. Although their work is implicitly motivated by a hypothesis of homophily, they do not make any mention of specific social theories, and no constraints or rules based on them are constructed. Instead, classification is achieved via Gradient Boosted Decision Trees.

Speriosu et al. [80] introduce an alternative approach to infer polarity that exploits the networked nature of social context. They compare three different approaches: a lexicon-based classifier (baseline), a maximum entropy classifier and Label Propagation (LPROP). The best results were achieved with LPROP, which is also appealing because it yields annotations for resources (e.g., lexicon), content and users indistinctly.

Similarly, Tan et al. [83] use a network approach based on SampleRank with a Markovian model. The model assumes that the sentiment of a given user is only influenced by the sentiment label of tweets generated by that user (consistency), and the sentiment of neighboring users (homophily).

Li et al. [48] compare an approach based on linguistic features with a combination of linguistic features and social features (referred to as global social evidence). The goal is sentiment analysis about political figures (targets) on Twitter and fora. In their hybrid approach, users, targets and issues (topics targets are vocal about) form a network. Three different hypotheses are then exploited on the data: 1) global consistency on indicative target-issue pairs, 2) global consistency on indicative target-target pairs, and 3) social balance. The results are slightly better than the baseline in the case of Twitter and widely better for forum data. A similar comparison of linguistic and social features is made by Aisopos et al. [1]. In their work, several classification algorithms are compared using different feature models, some of which include social context features.

Hu et al. [38] are the first in our review to include a classification algorithm specially tuned to incorporate social context. Their work is also interesting because they overcome the fact that most existing datasets only contain texts, which makes them unsuitable for social context analysis. They do so by combining text datasets with the friendship graph extracted from Kwak et al. [46].

Other works focus on user classification, such as Pozzi et al. [70]. They leverage connections in the network to infer user polarity, with highly positive results. User connections can also be exploited for content polarity classification. Ren and Wu [72] use both friendship and user-topic relations (calculated from user tweets) to calculate user-topic polarity. In addition to friendship, Deng et al. [23] use reply-to relations in online fora, as well as inferred friendship. West et al. [92] showed that the assumption of homophily in networks can improve polarity detection from short texts. They use social ties to infer the stance of users in Wikipedia. In particular, they exploit the social balance and

social status theories. They also point out the effect that the selection strategy of training and testing nodes has on accuracy. Tang et al. [84] use similar social theories to improve sentiment analysis on Twitter.

Lately, some works have introduced novel approaches such as Convolutional Networks [97]. In doing so, they add new types of features such as network embeddings, i.e., a vector representation of the network of a user, which can be fed into a classifier. The motivation behind these embeddings is to leverage language homophily in the analysis. Cheng et al. [16] follow in these steps, with a similar premise using content from a different social network (Reddit). In this case, the analysis also exploits the fact that comments are nested at different levels.

5.2. Datasets

The usual drawback with sentiment analysis datasets is that they rarely incorporate social context. This is either because social context was not taken into consideration when the dataset was collected or because of data protection policies and terms of use of the original OSN. The latter is usually easier to circumvent, as these datasets usually have IDs or pointers to the original resources, so that the necessary data can be recovered with the appropriate credentials and access to the OSN. This process is known as hydration, and it can be used to recover more data than was initially considered. i.e., it enables the expansion of the social context. The limitation is the fact that resources can be removed or made private before hydration. Table 3 shows basic statistics of the datasets used in the works reviewed.

RT Mind [70] contains a set of 62 users and 159 tweets, with positive or negative annotations. To collect this dataset, Pozzi et al. [70] crawled 2500 Twitter users who tweeted about Obama during two days in May 2013. For each user, their recent tweets (up to 3200, the limit of the API) were collected. At that point, only users that tweeted at least 50 times about Obama were considered. The tweets from those users that relate to Obama were kept and manually labeled by 3 annotators. The dataset contains ID of the tweet, ID of the author, text of the tweet, creation time, and sentiment (positive or negative).

The OMD dataset (Obama-McCain debate) [77] contains tweets about the televised debate between Senator John McCain, and then-Senator Barack Obama. The tweets were detected by following three hashtags: *#current*,*#tweetdebate*, and *#debate08*. The dataset contains tweets captured during the 97-minute debate, and 53 after it, to a total of 2.5 hours. There were 3238 tweets from 1160 people. There were 1824 tweets from 647 people during the actual debate and 1414 tweets from 738 people after it. Of those, only 1261 tweets, from 679 users, have sentiment annotations. The dataset includes tweet IDs, publication date, text, author name and nickname, and individual annotations of up to 7 annotators.

The Health Care Reform (HCR) [80] dataset contains tweets about the run-up to the signing of the health care bill in the USA on March 23, 2010. It was collected using the *#hcr* hashtag, from early 2010. A subset of the collected tweets were annotated with polarity (positive, negative, neutral and irrelevant)

Table 3: Datasets used in the experiments

	Source	Users	Entries
RT Mind [70]	Twitter	62	159
OMD [77]	Twitter	679	1261
HCR-DEV [80]	Twitter	806	1434
HCR-TEST [80]	Twitter	806	1434
STS [31]	Twitter	498	490
PF1901 [23]	Forum	412	1901
MF1560 [23]	Forum	320	1560
SemEval 2013 [56]	Twitter	3813	3813
SemEval 2014 [76]	Twitter	5749	5749
SemEval 2015 [75]	Twitter	2379	2379
Ciao [85]	Ciao	257682	10569
TASS [74]	Twitter	158	68017
YANG2011 [96]	Twitter	20M	476M
Li-Twitter [48]	Twitter	?	4646
Li-Forum [48]	Forum	?	762
AskMen [16]	Reddit	?	1057K
AskWomen [16]	Reddit	?	814K
Politics [16]	Reddit	?	2180K

and polarity targets (health care reform, Obama, Democrats, Republicans, Tea Party, conservatives, liberals, and Stupak) by Speriosu et al. [80]. The tweets were separated into training, dev (HCR-DEV) and test (HCR-TEST) sets. The dataset contains tweet ID, user ID and username, text of the tweet, sentiment, target of the sentiment, annotator and annotator ID.

The Stanford Twitter Sentiment (STS) [31] contains manually annotated tweets that mention a wide range of topics such as consumer products (40d, 50d, kindle2), companies (aig, at&t), and people (Bobby Flay, Warren Buffet). The version of the dataset used by Speriosu et al. [80] contains only 216 annotated tweets, 108 of which tweets are positive, and 75 are negative. However, the original paper [31] mentions 359 tweets with positive or negative sentiment. These figures are aligned with the content of the dataset at the authors' website[1], which also includes neutral tweets, to a total of 498 tweets by 490 authors. The discrepancy should be noted, both because Speriosu et al. [80] use the reduced dataset, and because they have released a collection of three datasets together with the source code they used to process it[2]. The collection is well documented, which might make it easier for other researchers to reuse their reduced dataset.

In their work, Deng et al. [23] include two datasets. The first dataset (PF1901) is crawled from the "Election & Campaigns" board of a political

[1] http://cs.stanford.edu/people/alecmgo/trainingandtestdata.zip
[2] https://bitbucket.org/speriosu/updown/

forum[3], There are 1901 labeled posts in total written by 232 unique users from March 2011 to April 2012. Out of those, 419 positive and 553 negative posts are also labeled with associated candidates. The rest are considered neutral or unsure. The second dataset (MF1560) is crawled from a military forum[4], containing 43 483 threads and 1 343 427 posts. In total, there are 1560 labeled posts written by 320 unique users, out of which 437 positive and 618 negative posts also had their topic labeled. The rest are considered neutral or unsure.

The collection of SemEval datasets originate from the competition set up for the different editions of the International Workshop on Semantic Evaluation (SemEval). SemEval includes several individual tasks, which focus on different types of classification, on different types of data. For this paper, we focus on the Tweet sentiment classification tasks. There is a dataset for each edition: SemEval 2013 [56], SemEval 2014 [76], SemEval 2015 [75]. For each tweet, the dataset contains the ID of the tweet, the ID of the author, and the sentiment label (positive, negative or neutral). To use the dataset, participants are encouraged to hydrate it, using the tools provided by the organizers of the competition.

The General Corpus TASS dataset is one of the three datasets created for the *Taller de análisis de sentimientos* (workshop on sentiment analysis) [74]. The other two datasets are the SocialTV dataset and the STOMPOL dataset, and they are focused on aspect based analysis. The dataset contains tweets in Spanish, authored by 150 well-known personalities and celebrities of the world of politics, economy, communication, mass media and culture. The original corpus is released in XML format, and it includes date, author and ID of each tweet.

The AskMen, AskWomen and Politics datasets Cheng et al. [16][5] contain posts from popular subreddits (subcategories within the Reddit OSN[6] with different topics and styles: AskWomen (814K comments), AskMen (1057K comments), and Politics (2180K comments).

Yang and Leskovec [96] collected a dataset of nearly 476 million Twitter posts from 20 million users covering eight months, from June 2009 to February 2010. Aisopos et al. [1] filter the dataset in their work down to 6.12 million negative and 14.12 million positive tweets using emoticons. From those tweets, they finally used a sample of 1 million tweets with each polarity.

Li et al. [48] collected datasets from two OSN: an online forum and Twitter. The forum dataset was collected from the most recent posts at the "Elections & Campaigns" forum (similarly to Deng et al. [23]), from March 2011 to December 2011. 97.3% of those posts subjective, i.e., they contain positive or negative sentiments. The tweet data set was automatically collected by retrieving positive instances with *#Obama2012* or *#GOP2012* hashtags, and negative instances

[3] http://www.politicalforum.com/elections-campaigns/
[4] http://forums.military.com/
[5] https://github.com/hao-cheng/factored_neural/
[6] https://reddit.com

with *#Obamafail* or *#GOPfail* hashtags. All tweets where the hashtags of interest were not located at the very end of the message were filtered.

Lastly, the Ciao dataset [85] includes opinions on the Ciao website[7] in May 2011. The authors started the collection of the dataset with a set of most active users and then did a breadth-first search until no new users could be found. The sentiment in the dataset is expressed with a 5-star rating system.

5.3. Features

This section briefly covers some of the features that can be extracted from social context at different levels.

5.3.1. Micro features

At the micro level, features may be related to the content author, or to the content itself. From the user, the main set of features is:

- Number of followees. In OSN such as Twitter, users (followers) are exposed only to the content of their followees. This is typically an asymmetrical relation. Following another user does not require the followee to accept, except for private accounts and blocked users. For this reason, it is typical for users to follow hundreds or even thousands of users [46]. Hence, this feature is rather noisy. Some works refer to followees as friends, whereas other works reserve the term friend for mutual followers.
- Number of followers. In contrast with the previous feature, only a fraction of users tend to accumulate most of the followers [46]. As a result, the number of followers is more informative.
- Number of friends. In some instances, the number of followers that the user follows back is known. Otherwise, it has to be calculated from the meso network.
- Ratio of positive / negative / neutral content (per topic). This may indicate the typical sentiment polarity for a user. Some theories such as author coherence indicate that the sentiment we show about a topic tends to be stable over short periods. Moreover, studies show that different types of users exhibit characteristic sentiment patterns in their posts. Namely, popular users are more likely to post positive content.
- Age, gender and nationality. All these features influence the way we communicate, from the language we use to the sentiment we are more likely to express, and they have been shown to help in sentiment analysis [88].

Content may also be linked to features such as:

[7] http://www.ciao.co.uk

- Number of favorites, retweets, and replies. These values gradually increase as more users interact with the content. For this reason, it may take some time for them to stabilize or become meaningful, and it is not available in online analysis unless some delay is added. By using specific time windows, it is also possible to snapshot the value of the metric at different times, to create derived metrics. e.g., number of replies during the first hour, and number of replies during the first day. This type of analysis also borders dynamic social context, which we have discussed earlier.

- Topic(s). The topic could either be extracted from content and metadata such as hashtags or automatically inferred with topic detection.

- Sentiment of the original message. It is only available for replies. It may be beneficial to know the original creator and the views of the creators, as that enables the use of social theories (e.g., Li et al. [48]).

- Sentiment ratio of replies. This information is not typically used because it requires a posteriori knowledge. However, for some types of offline classification, this information is known at the time of prediction.

Additionally, it is also possible to generate user and topic-specific models or to embed the context of the topical context of the content [23, 16]. Network-based algorithms such as label propagation and algorithms that take arbitrary input sizes, such as recurrent neural networks, are not constrained by a fixed input space. As a result, they can incorporate features of the context without aggregation, such as averaging.

5.3.2. $Meso_r$ features

At this level, a network of users and content also starts to form. Connections in this network may be directed or undirected. Some examples of relations that can originate a network are:

- Follower relation (directed). This is the relation that, when aggregated, gives rise to the number of followees and number of followers in the previous section. It is the most common type of relation, and it typically requires further filtering, given both the tendency of users to follow hundreds of users and the lack of confirmation from the other side.

- Mutual follower relation (undirected). A simple follower relation often yields poor results. The cause could be that this type of relation is too weak [20], and is non-reciprocal. Most works use mutual relations instead, where users are only connected if they follow each other.

- Ratio of Common Followers/followees relation (undirected). This is a measure of how many followers/followees two users have in common. Under the hypothesis of homophily, it may be a proxy for user similarity. More elaborate versions may take into account the number of followees/followers of the followers/followees, via a weighted sum.

- Ratio of Common Topics/Keywords relation (undirected). Similar to the ratio of Common Followers/followees, it is related to the similarity of two users, based on the content they share.

5.3.3. $Meso_i$ features

Interactions can also be used to create a network. For instance:

- Reply interaction (directed). The act of replying forms one relation between the original content, and the content to which it replies. However, two interaction links can be formed as well: one between both users, and another one between the user and the original content. Since replies are less likely to occur than retweets, they tend to be more informative.
- Mention interaction (directed). When a user mentions another user in their content, two links are formed: a mention interaction between the two users, and a relation between the content and the user that was mentioned.
- Like/favorite interaction (directed). In most OSN, users can mark content they like. As opposed to a reply, liking is usually achieved with a single click. Hence, this is amongst the most common types of interactions.
- Retweet/reshare interaction (directed). Retweeting is the act of sharing content from a different user verbatim.
- Shared a conversation (undirected). When two users engage in a conversation (a series of replies), it can be encoded as a new interaction between the users.

The ability to relate an author to other users enables the propagation of micro features over the meso network, which yields a new set of features, such as:

- Sentiment ratio of neighbors. The ratio of positive/negative/neutral neighbors. Neighbors could be adjacent users (those sharing an edge), or users that belong to the same group (e.g., the same community). These neighbors could be filtered, e.g., to only take new neighbors into account, or neighbors that have had recent activity. The sentiment for each neighbor could also be calculated in time windows or weighted so that recent content is more important.
- Sentiment ratio of content by neighbors. Similar to the previous one, without aggregating on the user level.

Lastly, some techniques allow embedding large information networks (be it content, user or mixed networks) into low-dimensional vector spaces. These types of techniques are increasingly popular in contextless analysis due to their excellent performance [3]. The components of the embedding can then be used as features, either on their own or combined with other features. One example

of network embedding is the LINE method [86], which is used in one of the works reviewed [16]. However, LINE does not take different types of nodes or relationships into account. The heterogeneous network embedding model [13] is an alternative. Although it was conceived to embed networks of text and images, it could be adapted to encode mixed networks of content and users.

5.3.4. $Meso_e$ features and Enrichment through Social Network Analysis

Social Network Analysis provides several methods to process, examine and describe a social network. These methods use the network topology and its attributes and infer information that could be useful for sentiment analysis tasks. For instance, there are several ways to measure user popularity and influence in a social network, according to different criteria. As a result, the impact of each user in the sentiment prediction can be weighted. Similarly, the importance of user connections (relations and interactions) can be measured. Thus, the granularity can be set at the connection level, where sentiment prediction is not only influenced by neighboring users, but also on the strength of the connection to those neighbors. Another example is community detection, which could help segment the user base into smaller groups that exhibit similar behavior.

5.3.5. Macro features

Macro features include any type of information that is outside of the realm of the OSN. Hence, the possibilities for features in this category are unlimited. Of all the works we have reviewed, only one [48] uses macro features. In particular, it uses known enmity or opposition between politicians, together with social theories about user and target consistency. Other possibilities include the analysis of links to external sources or attachments.

5.4. Performance

Having described these works, it is also important to compare their performance. Few works use the same dataset in the same conditions. Instead of providing that comparison, Table 4 summarizes the best results for content-level classification in every work surveyed, at every level of analysis identified in the taxonomy in Section 4. The table shows both results for F1-score and accuracy, when available. As expected, the results show that social context improves the performance over the contextless baseline.

For completeness, Figure 4 and Figure 5 show all the results reported in these works, grouped by the level of analysis. The performance is shown relative to the contextless baseline in every dataset.

5.5. Other Approaches

Although this paper focuses on using social context to improve sentiment analysis, there are other ways in which sentiment information can be fused with other sources or types of information [4]. For instance, sentiment information can be included into existing social network analysis. This can be done to characterize or explain a given phenomenon. When adding sentiment information,

Table 4: Maximum Accuracy score reported in each work, per level of analysis and dataset.

Work	Level Dataset	Metric	Baseline	micro	$meso_r$	$meso_i$	$meso_e$	macro
[1]	YANG2011	Acc.	97.42	60.40	-	80.08	-	-
[23]	MF1560	Acc.	46.64	-	55.60	-	-	-
	PF1901	Acc.	61.24	-	72.75	-	-	-
[48]	Li-Forum	Acc.	59.61	67.24	62.89	-	-	71.97
	Li-Twitter	Acc.	83.97	-	85.35	-	-	-
[79]	TASS	Acc.	79.30	-	-	89.80	-	-
[80]	HCR-DEV	Acc.	58.60	65.70	65.20	-	-	-
	HCR-TEST	Acc.	62.90	71.20	71.00	-	-	-
	OMD	Acc.	61.30	66.70	66.50	-	-	-
	STS	Acc.	83.10	84.70	84.70	-	-	-
[95]	HCR	Acc.	69.00	-	-	-	77.5	-
	OMD	Acc.	76.00	-	-	-	76.0	-
[16]	AskMen	F1	51.70	-	-	52.70	-	-
	AskWomen	F1	55.20	-	-	56.30	-	-
	Politics	F1	53.00	-	-	54.80	-	-
[79]	TASS	F1	69.20	-	-	90.20	-	-
[97]	Ciao	F1	-	-	-	80.19	-	-
	SE 2013	F1	69.31	-	71.49	71.91	-	-
	SE 2014	F1	72.73	-	74.17	75.07	-	-
	SE 2015	F1	63.24	-	66.00	66.75	-	-

some patterns and trends emerge, which would otherwise be lost in the global aggregate. For instance, sentiment information can be used to analyze different Twitter communities separately instead of aggregating their results [22].

Sentiment and social network analysis can also be combined to find potentially radicalized users [6], or to highlight emotionally charged content [24]. Additionally, sentiment information alone has proved to yield very high precision and a low recall in some user classification tasks [67]. This suggests that sentiment information could be crucial in positively identifying members of specific groups.

6. Conclusions and future work

The question that motivated this work was whether there is valuable information in social networks that has the potential to improve sentiment analysis in specific scenarios. We refer to this information as social context. To answer this question, three related questions need to be answered: "what is social context?"(Q1), "can social context improve sentiment analysis?"(Q2) and "what elements of social context are more relevant for sentiment analysis?"(Q3).

To answer the first question (Q1), we analyzed the use and definitions of

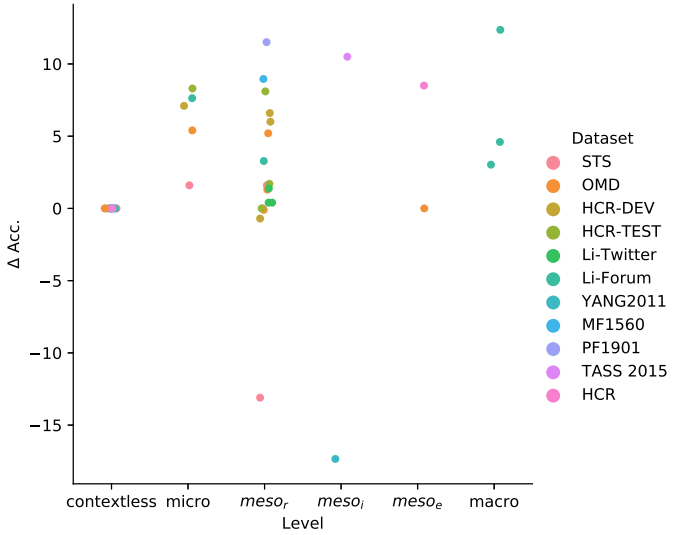


Figure 4: Difference in accuracy with respect to a contextless approach in all works analyzed, per dataset. The results for [1] have been removed due to their unusually high accuracy (Table 4).

social context in the state of the art. Our analysis revealed that there are commonalities between these works, despite differences in notation. We formalized these commonalities in a formal definition of social context. This definition enables a richer and more precise description of social media information.

We used this definition in a new framework for comparison of approaches to sentiment analysis using social context. Part of this framework is a taxonomy of approaches, which shows the different levels of social context that are possible. Using this taxonomy, we compared works in the literature. The results of this comparison, which are included in this work, support the notion that using social context may improve performance in sentiment analysis (Q2), both in content classification and user classification tasks.

Once these levels of analysis have been identified, the natural question is what performance gains can be achieved by using more complex features. Directly comparing their results is not straightforward, but the taxonomy can be used to group approaches and to compare these groups. Higher results correspond to more detailed definitions of Social Context, as shown by $meso_i$ approaches outperforming $meso_r$ ones in most works (Q3). The trend seems to support these results, as recent works are starting to incorporate $meso_i$ ap-

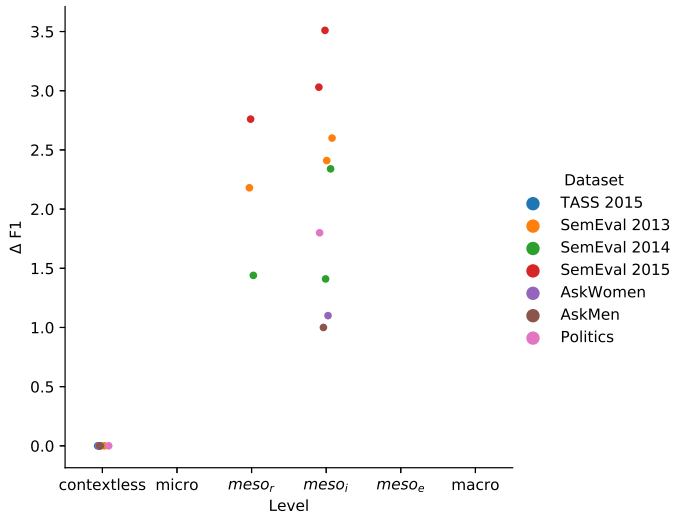


Figure 5: Difference in F1 score with respect to a contextless approach in all works analyzed, per dataset.

proaches. Unfortunately, the number of works in the field is not enough to provide an accurate evaluation of the specific elements of content (e.g., whether retweet interactions are more informative than community detection).

On the other hand, the trend suggests that there is room for improvement in the processing of social context and its use with different classifiers. For instance, techniques such as network embeddings could be used to condense several aspects of social context.

We expect that the formal definition of context and the framework in this work foster the use of social context in sentiment analysis in two ways. Firstly, by providing a common language to express social context. Secondly, by allowing future works to perform a more systematic comparison with existing approaches. As more works start leveraging social context, the taxonomy of approaches will likely grow and add novel ideas. Similarly, more elements may need to be included in the definition of social context to account for more complex scenarios.

Acknowledgments

This work is supported by the Spanish Ministry of Economy and Competitiveness under the R&D project SEMOLA (TEC2015-68284-R) and the Euro-

pean Union under the project Trivalent (H2020 Action Grant No. 740934, SEC-06-FCT-2016). The authors also want to mention earlier work that contributed to the results in this paper. More specifically, the MixedEmotions (European Union's Horizon 2020 Programme research and innovation programme under grant agreements No. 644632) and SoMeDi (ITEA3 16011) projects.

References

[1] Aisopos, F., Papadakis, G., Tserpes, K., Varvarigou, T., 2012. Content vs. context for sentiment analysis: a comparative analysis over microblogs. In: Proceedings of the 23rd ACM conference on Hypertext and social media. ACM, pp. 187–196.

[2] Alvarez, R., Garcia, D., Moreno, Y., Schweitzer, F., Dec. 2015. Sentiment cascades in the 15m movement. EPJ Data Science 4 (1).

[3] Araque, O., Corcuera-Platas, I., Sánchez-Rada, J. F., Iglesias, C. A., Jun. 2017. Enhancing Deep Learning Sentiment Analysis with Ensemble Techniques in Social Applications. Expert Systems with Applications.

[4] Balazs, J. A., Velásquez, J. D., Jan. 2016. Opinion Mining and Information Fusion: A survey. Information Fusion 27, 95–110.

[5] Bengio, Y., Nov. 2009. Learning Deep Architectures for AI. Foundations and Trends® in Machine Learning 2 (1), 1–127.

[6] Bermingham, A., Conway, M., McInerney, L., O'Hare, N., Smeaton, A. F., 2009. Combining social network analysis and sentiment analysis to explore the potential for online radicalisation. In: Social Network Analysis and Mining, 2009. ASONAM'09. International Conference on Advances in. IEEE, pp. 231–236.

[7] Bolíbar, M., Sep. 2016. Macro, meso, micro: broadening the 'social' of social network analysis with a mixed methods approach. Quality & Quantity 50 (5), 2217–2236.

[8] Borgatti, S. P., Mehra, A., Brass, D. J., Labianca, G., Feb. 2009. Network Analysis in the Social Sciences. Science 323 (5916), 892–895.

[9] Buitelaar, P., Arcan, M., Iglesias, C., Sanchez-Rada, F., Strapparava, C., 2013. Linguistic linked data for sentiment analysis. In: Proceedings of the 2nd Workshop on Linked Data in Linguistics (LDL-2013): Representing and linking lexicons, terminologies and other language data. pp. 1–8, 00015.

[10] Cai, H., Zheng, V. W., Chang, K. C., Sep. 2018. A Comprehensive Survey of Graph Embedding: Problems, Techniques, and Applications. IEEE Transactions on Knowledge and Data Engineering 30 (9), 1616–1637, 00123.

[11] Cambria, E., 2016. Affective computing and sentiment analysis. IEEE Intelligent Systems 31 (2), 102–107.

[12] Cavallari, S., Zheng, V. W., Cai, H., Chang, K. C.-C., Cambria, E., 2017. Learning community embedding with community detection and node embedding on graphs. In: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management. ACM, pp. 377–386.

[13] Chang, S., Han, W., Tang, J., Qi, G.-J., Aggarwal, C. C., Huang, T. S., 2015. Heterogeneous network embedding via deep architectures. In: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, pp. 119–128.

[14] Chaturvedi, I., Cambria, E., Welsch, R. E., Herrera, F., Nov. 2018. Distinguishing between facts and opinions for sentiment analysis: Survey and challenges. Information Fusion 44, 65–77.

[15] Chen, H., Liu, J., Lv, Y., Li, M. H., Liu, M., Zheng, Q., Nov. 2018. Semi-supervised clue fusion for spammer detection in Sina Weibo. Information Fusion 44, 22–32.

[16] Cheng, H., Fang, H., Ostendorf, M., 2017. A Factored Neural Network Model for Characterizing Online Discussions in Vector Space. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. pp. 2296–2306.

[17] Cheng, J., Adamic, L., Dow, P. A., Kleinberg, J. M., Leskovec, J., 2014. Can Cascades Be Predicted? In: Proceedings of the 23rd International Conference on World Wide Web. WWW '14. ACM, New York, NY, USA, pp. 925–936.

[18] Cho, H., Lee, J.-S., Aug. 2008. Collaborative Information Seeking in Intercultural Computer-Mediated Communication Groups: Testing the Influence of Social Context Using Social Network Analysis. Communication Research 35 (4), 548–573, 00090.

[19] Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., Kuksa, P., 2011. Natural language processing (almost) from scratch. The Journal of Machine Learning Research 12, 2493–2537.

[20] Darmon, D., Omodei, E., Garland, J., Aug. 2015. Followers Are Not Enough: A Multifaceted Approach to Community Detection in Online Social Networks. PLOS ONE 10 (8).

[21] Davidov, D., Tsur, O., Rappoport, A., 2010. Enhanced sentiment learning using twitter hashtags and smileys. In: Proceedings of the 23rd international conference on computational linguistics: posters. Association for Computational Linguistics, pp. 241–249.

[22] Deitrick, W., Hu, W., 2013. Mutually Enhancing Community Detection and Sentiment Analysis on Twitter Networks. Journal of Data Analysis and Information Processing 01 (03), 19–29.

[23] Deng, H., Han, J., Li, H., Ji, H., Wang, H., Lu, Y., 2014. Exploring and inferring user–user pseudo-friendship for sentiment analysis with heterogeneous networks. Statistical Analysis and Data Mining: The ASA Data Science Journal 7 (4), 308–321.

[24] Gamon, M., Basu, S., Belenko, D., Fisher, D., Hurst, M., König, A. C., 2008. BLEWS: Using blogs to provide context for news articles. In: ICWSM. pp. 60–67.

[25] Gao, B., Berendt, B., Clarke, D., De Wolf, R., Peetz, T., Pierson, J., Sayaf, R., 2012. Interactive grouping of friends in OSN: Towards online context management. In: Data Mining Workshops (ICDMW), 2012 IEEE 12th International Conference on. IEEE, pp. 555–562.

[26] Garas, A., Garcia, D., Skowron, M., Schweitzer, F., May 2012. Emotional persistence in online chatting communities. Scientific Reports 2.

[27] Garcia, D., Abisheva, A., Schweighofer, S., Serdült, U., Schweitzer, F., Mar. 2015. Ideological and Temporal Components of Network Polarization in Online Political Participatory Media: Ideological and Temporal Components of Network. Policy & Internet 7 (1), 46–79.

[28] García-Pablos, A., Cuadros Oller, M., Rigau Claramunt, G., 2016. A comparison of domain-based word polarity estimation using different word embeddings. In: Proceedings of the Tenth International Conference on Language Resources and Evaluation. Portoroz, Slovenia.

[29] Genc, Y., Sakamoto, Y., Nickerson, J., 2011. Discovering context: classifying tweets through a semantic transform based on wikipedia. Foundations of augmented cognition. Directing the future of adaptive systems, 484–492.

[30] Gimpel, K., Schneider, N., O'Connor, B., Das, D., Mills, D., Eisenstein, J., Heilman, M., Yogatama, D., Flanigan, J., Smith, N. A., 2011. Part-of-speech Tagging for Twitter: Annotation, Features, and Experiments. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers - Volume 2. HLT '11. Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 42–47.

[31] Go, A., Bhayani, R., Huang, L., 2009. Twitter sentiment classification using distant supervision. CS224N Project Report, Stanford 1 (12).

[32] Guo, W., Li, H., Ji, H., Diab, M. T., 2013. Linking Tweets to News: A Framework to Enrich Short Text Data in Social Media. In: ACL (1). pp. 239–249.

[33] Hajian, B., White, T., Oct. 2011. Modelling Influence in a Social Network: Metrics and Evaluation. In: 2011 IEEE Third International Conference on Privacy, Security, Risk and Trust and 2011 IEEE Third International Conference on Social Computing. pp. 497–500.

[34] Heo, Y.-C., Park, J.-Y., Kim, J.-Y., Park, H.-W., May 2016. The emerging viewertariat in South Korea: The Seoul mayoral TV debate on Twitter, Facebook, and blogs. Telematics and Informatics 33 (2), 570–583, 00014.

[35] Hill, A. L., Rand, D. G., Nowak, M. A., Christakis, N. A., Jul. 2010. Emotions as infectious diseases in a large social network: the SISa model. Proceedings of the Royal Society of London B: Biological Sciences, rspb20101217.

[36] Hogenboom, A., Bal, D., Frasincar, F., Bal, M., De Jong, F., Kaymak, U., 2015. Exploiting Emoticons in Polarity Classification of Text. J. Web Eng. 14 (1&2), 22–40, 00043.

[37] Hovy, D., 2015. Demographic Factors Improve Classification Performance. In: ACL (1). pp. 752–762.

[38] Hu, X., Tang, L., Tang, J., Liu, H., 2013. Exploiting Social Relations for Sentiment Analysis in Microblogging. In: Proceedings of the Sixth ACM International Conference on Web Search and Data Mining. WSDM '13. ACM, New York, NY, USA, pp. 537–546.

[39] Jansen, B. J., Zhang, M., Sobel, K., Chowdury, A., 2009. Twitter Power : Tweets as Electronic Word of Mouth. Journal of the American Society for Information Science 60 (11), 2169–2188.

[40] Jiang, F., Liu, Y.-Q., Luan, H.-B., Sun, J.-S., Zhu, X., Zhang, M., Ma, S.-P., 2015. Microblog sentiment analysis with emoticon space model. Journal of Computer Science and Technology 30 (5), 1120–1129, 00026.

[41] Kaplan, A. M., Haenlein, M., Jan. 2010. Users of the world, unite! The challenges and opportunities of Social Media. Business Horizons 53 (1), 59–68.

[42] Kim, Y., Oct. 2014. Convolutional neural networks for sentence classification. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). Association for Computational Linguistics, Doha, Qatar, pp. 1746–1751.

[43] Kim, Y., Sohn, D., Choi, S. M., Jan. 2011. Cultural difference in motivations for using social network sites: A comparative study of American and Korean college students. Computers in Human Behavior 27 (1), 365–372, 00708.

[44] Kiritchenko, S., Zhu, X., Mohammad, S. M., Aug. 2014. Sentiment Analysis of Short Informal Texts. Journal of Artificial Intelligence Research 50, 723–762.

[45] Kramer, A. D., Guillory, J. E., Hancock, J. T., 2014. Experimental evidence of massive-scale emotional contagion through social networks. Proceedings of the National Academy of Sciences.

[46] Kwak, H., Lee, C., Park, H., Moon, S., 2010. What is Twitter, a Social Network or a News Media? In: Proceedings of the 19th International Conference on World Wide Web. WWW '10. ACM, New York, NY, USA, pp. 591–600.

[47] Leskovec, J., Huttenlocher, D., Kleinberg, J., 2010. Signed networks in social media. In: Proceedings of the SIGCHI conference on human factors in computing systems. ACM, pp. 1361–1370.

[48] Li, H., Chen, Y., Ji, H., Muresan, S., Zheng, D., 2012. Combining Social Cognitive Theories with Linguistic Features for Multi-genre Sentiment Analysis. In: PACLIC. pp. 127–136.

[49] Lipton, Z. C., 2016. The mythos of model interpretability. arXiv preprint arXiv:1606.03490.

[50] Lu, Y., Tsaparas, P., Ntoulas, A., Polanyi, L., 2010. Exploiting social context for review quality prediction. In: Proceedings of the 19th international conference on World wide web. ACM, pp. 691–700.

[51] Marcus, G., 2018. Deep learning: A critical appraisal. arXiv preprint arXiv:1801.00631.

[52] McCrae, J., Spohr, D., Cimiano, P., 2011. Linking lexical resources and ontologies on the semantic web with lemon. In: Extended Semantic Web Conference. Springer, pp. 245–259, 00210.

[53] McPherson, M., Smith-Lovin, L., Cook, J. M., 2001. Birds of a feather: Homophily in social networks. Annual review of sociology 27 (1), 415–444.

[54] Melville, P., Gryc, W., Lawrence, R. D., 2009. Sentiment Analysis of Blogs by Combining Lexical Knowledge with Text Classification. In: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. KDD '09. ACM, New York, NY, USA, pp. 1275–1284.

[55] Mikolov, T., Chen, K., Corrado, G., Dean, J., 2013. Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.

[56] Nakov, P., Rosenthal, S., Kozareva, Z., Stoyanov, V., Ritter, A., Wilson, T., 2013. SemEval-2013 Task 2: Sentiment analysis in Twitter. In: Proceedings of the 7th International Workshop on Semantic Evaluation. SemEval '13. Vol. 7. Atlanta, Georgia, USA, pp. 312–320.

[57] Nasukawa, T., Yi, J., 2003. Sentiment Analysis: Capturing Favorability Using Natural Language Processing. In: Proceedings of the 2Nd International Conference on Knowledge Capture. K-CAP '03. ACM, New York, NY, USA, pp. 70–77.

[58] Nguyen, M.-T., Tran, D.-V., Nguyen, L.-M., Dec. 2017. Social context summarization using user-generated content and third-party sources. Knowledge-Based Systems.

[59] Noro, T., Tokuda, T., Jul. 2016. Searching for Relevant Tweets Based on Topic-related User Activities. J. Web Eng. 15 (3-4), 249–276.

[60] Novak, P. K., Smailović, J., Sluban, B., Mozetič, I., 2015. Sentiment of emojis. PloS one 10 (12), e0144296, 00226.

[61] Orman, G. K., Labatut, V., Cherifi, H., 2011. Qualitative comparison of community detection algorithms. In: International conference on digital information and communication technology and its applications. Springer, pp. 265–279.

[62] Otte, E., Rousseau, R., Dec. 2002. Social network analysis: a powerful strategy, also for the information sciences. Journal of Information Science 28 (6), 441–453.

[63] Pak, A., Paroubek, P., 2010. Twitter as a corpus for sentiment analysis and opinion mining. In: LREc. Vol. 10. pp. 1320–1326.

[64] Pang, B., Lee, L., 2008. Opinion mining and sentiment analysis. Foundations and Trends® in Information Retrieval 2 (1–2), 1–135.

[65] Pang, B., Lee, L., Vaithyanathan, S., 2002. Thumbs Up?: Sentiment Classification Using Machine Learning Techniques. In: Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing - Volume 10. EMNLP '02. Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 79–86.

[66] Papadopoulos, S., Kompatsiaris, Y., Vakali, A., Spyridonos, P., 2012. Community detection in social media. Data Mining and Knowledge Discovery 24 (3), 515–554.

[67] Pennacchiotti, M., Popescu, A.-M., 2011. A Machine Learning Approach to Twitter User Classification. Icwsm 11 (1), 281–288.

[68] Polanyi, L., Zaenen, A., 2006. Contextual valence shifters. In: Computing attitude and affect in text: Theory and applications. Springer, pp. 1–10.

[69] Poria, S., Cambria, E., Bajpai, R., Hussain, A., Sep. 2017. A review of affective computing: From unimodal analysis to multimodal fusion. Information Fusion 37, 98–125.

[70] Pozzi, F. A., Maccagnola, D., Fersini, E., Messina, E., 2013. Enhance user-level sentiment analysis on microblogs with approval relations. In: Congress of the Italian Association for Artificial Intelligence. Springer, pp. 133–144.

[71] Ravi, K., Ravi, V., Nov. 2015. A survey on opinion mining and sentiment analysis: Tasks, approaches and applications. Knowledge-Based Systems 89 (Supplement C), 14–46.

[72] Ren, F., Wu, Y., Oct. 2013. Predicting User-Topic Opinions in Twitter with Social and Topical Context. IEEE Transactions on Affective Computing 4 (4), 412–424.

[73] Rokach, L., Feb. 2010. Ensemble-based classifiers. Artificial Intelligence Review 33 (1-2), 1–39.

[74] Román, J. V., Cámara, E. M., Morera, J. G., Zafra, S. M. J., 2015. TASS 2014-the challenge of aspect-based sentiment analysis. Procesamiento del Lenguaje Natural 54, 61–68.

[75] Rosenthal, S., Nakov, P., Kiritchenko, S., Mohammad, S., Ritter, A., Stoyanov, V., 2015. Semeval-2015 task 10: Sentiment analysis in twitter. In: Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015). pp. 451–463.

[76] Rosenthal, S., Ritter, A., Nakov, P., Stoyanov, V., 2014. SemEval-2014 Task 9: Sentiment Analysis in Twitter. In: Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014). Dublin, Ireland, pp. 73–80.

[77] Shamma, D. A., Kennedy, L., Churchill, E. F., 2009. Tweet the Debates: Understanding Community Annotation of Uncollected Sources. In: Proceedings of the First SIGMM Workshop on Social Media. WSM '09. ACM, New York, NY, USA, pp. 3–10.

[78] Sharma, A., Dey, S., 2012. A comparative study of feature selection and machine learning techniques for sentiment analysis. In: Proceedings of the 2012 ACM research in applied computation symposium. ACM, pp. 1–7.

[79] Sixto, J., Almeida, A., López-de Ipiña, D., 2018. Analysis of the Structured Information for Subjectivity Detection in Twitter. Transactions on Computational Collective Intelligence XXIX, 163–181.

[80] Speriosu, M., Sudan, N., Upadhyay, S., Baldridge, J., 2011. Twitter Polarity Classification with Label Propagation over Lexical Links and the Follower Graph. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, pp. 53–56.

[81] Sánchez-Rada, J. F., Iglesias, C. A., 2016. Onyx: A linked data approach to emotion representation. Information Processing & Management 52 (1), 99–114, 00026.

[82] Taboada, M., Brooke, J., Tofiloski, M., Voll, K., Stede, M., Apr. 2011. Lexicon-Based Methods for Sentiment Analysis. Computational Linguistics 37 (2), 267–307.

[83] Tan, C., Lee, L., Tang, J., Jiang, L., Zhou, M., Li, P., 2011. User-level Sentiment Analysis Incorporating Social Networks. In: Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. KDD '11. ACM, New York, NY, USA, pp. 1397–1405.

[84] Tang, J., Chang, Y., Liu, H., Jun. 2014. Mining Social Media with Social Theories: A Survey. SIGKDD Explor. Newsl 15 (Iid), 20–29.

[85] Tang, J., Gao, H., Liu, H., 2012. mTrust: discerning multi-faceted trust in a connected world. In: Proceedings of the fifth ACM international conference on Web search and data mining - WSDM '12. ACM Press, Seattle, Washington, USA, p. 93.

[86] Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J., Mei, Q., 2015. Line: Large-scale information network embedding. In: Proceedings of the 24th International Conference on World Wide Web. International World Wide Web Conferences Steering Committee, pp. 1067–1077.

[87] Tommasel, A., Godoy, D., Mar. 2018. A Social-aware online short-text feature selection technique for social media. Information Fusion 40, 1–17.

[88] Volkova, S., Wilson, T., Yarowsky, D., 2013. Exploring demographic language variations to improve multilingual sentiment analysis in social media. In: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing. pp. 1815–1827.

[89] Volkova, Svitlana, 2015. Predicting Demographics and Affect in Social Networks. Ph.D. thesis, Johns Hopkins University, Baltimore, Maryland.

[90] Wang, S., Manning, C. D., 2012. Baselines and Bigrams: Simple, Good Sentiment and Topic Classification. In: Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers - Volume 2. ACL '12. Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 90–94.

[91] Wei, W., Gulla, J. A., 2010. Sentiment learning on product reviews via sentiment ontology tree. In: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, pp. 404–413, 00134.

[92] West, R., Paskov, H. S., Leskovec, J., Potts, C., 2014. Exploiting Social Network Structure for Person-to-Person Sentiment Analysis. CoRR abs/1409.2450.

[93] Wu, F., Shu, J., Huang, Y., Yuan, Z., Aug. 2016. Co-detecting social spammers and spam messages in microblogging via exploiting social contexts. Neurocomputing 201, 51–65.

[94] Xia, R., Zong, C., 2010. Exploring the Use of Word Relation Features for Sentiment Classification. In: Proceedings of the 23rd International Conference on Computational Linguistics: Posters. COLING '10. Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 1336–1344.

[95] Xiaomei, Z., Jing, Y., Jianpei, Z., Hongyu, H., Feb. 2018. Microblog sentiment analysis with weak dependency connections. Knowledge-Based Systems 142, 170–180.

[96] Yang, J., Leskovec, J., 2011. Patterns of temporal variation in online media. In: Proceedings of the fourth ACM international conference on Web search and data mining. ACM, pp. 177–186.

[97] Yang, Y., Eisenstein, J., Aug. 2017. Overcoming Language Variation in Sentiment Analysis with Social Attention. Transactions of the Association for Computational Linguistics 5, 295–307.

3.3.2 CRANK: A Hybrid Model for User and Content Sentiment Classification Using Social Context and Community Detection

Title	CRANK: A Hybrid Model for User and Content Sentiment Classification Using Social Context and Community Detection
Authors	Sánchez-rada, J. Fernando and Carlos A. Iglesias
Journal	Applied Sciences
Impact factor	JCR 2018 Q2 (2.217)
ISSN	2076-3417
Year	2020
Keywords	
Pages	21
Abstract	Recent works have shown that sentiment analysis on social media can be improved by fusing text with social context information. Social context is information such as relationships between users, and interactions of users with content. Although existing works have already exploited the networked structure of social context by using graphical models or techniques such as label propagation, more advanced techniques from Social Network Analysis remain unexplored. Our hypothesis is that these techniques can help reveal underlying features that could help with the analysis. In this work, we present a sentiment classification model (CRANK) that leverages community partitions to improve both user and content classification. We have evaluated this model on existing datasets, and compared it to other approaches.

MDPI

Article

CRANK: A Hybrid Model for User and Content Sentiment Classification Using Social Context and Community Detection

J. Fernando Sánchez-Rada[1,†,‡] and Carlos A. Iglesias[1,‡]

[1] Grupo de Sistemas Inteligentes, Universidad Politécnica de Madrid; {jf.sanchez,carlosangel.iglesias}@upm.es

Version February 4, 2020 submitted to Appl. Sci.

Abstract: Recent works have shown that sentiment analysis on social media can be improved by fusing text with social context information. Social context is information such as relationships between users, and interactions of users with content. Although existing works have already exploited the networked structure of social context by using graphical models or techniques such as label propagation, more advanced techniques from Social Network Analysis remain unexplored. Our hypothesis is that these techniques can help reveal underlying features that could help with the analysis. In this work, we present a sentiment classification model (CRANK) that leverages community partitions to improve both user and content classification. We have evaluated this model on existing datasets, and compared it to other approaches.

Keywords: sentiment analysis; social context; social network analysis; online social networks

1. Introduction

State of the art in the field of sentiment analysis has improved considerably in recent years, partly due to the advent of social media. Social media text imposes several limitations that are hard to overcome even for human annotators, such as the extensive use of annotations, jargon and heavy reliance on context. Moreover, understanding a piece of content often requires following a conversation (i.e., a thread of replies), or the style and stance of the author of the content.

To solve these limitations, new approaches are starting to combine text with additional information from the social network, such as links between users, and previous posts by each user. The blend of all this information can be referred to as social context. A recent work [1] analyzes the use of social context in the sentiment analysis literature, and it shows that context-based approaches perform better than traditional analysis without social context (i.e., contextless approaches). It also provides a taxonomy of approaches based on the types of features included in the context: *contextless* approaches do not use social context at all; *micro* approaches only use features from the user and their content; *meso* approaches include features from other users and content, as well as connections between different users and content; and *macro* approaches also exploit other sources such as knowledge graphs. *meso* approaches are further divided into three categories: $meso_r$ only use relations (e.g., follower-followee); $meso_i$ add interactions (e.g., replies and likes); and $meso_e$ use Social Network Analysis (SNA) techniques to process other elements of the context and generate additional features. Comparing the performance of existing approaches seems to show that more elaborate features provide an advantage over simpler features. Simpler features are those directly extracted from the network, such as follower-followee relations ($meso_r$). More complex features can be obtained from applying further processing, typically through filtering and aggregating information from the network ($meso_i$), or through SNA techniques such as calculating user centrality or unsupervised community detection ($meso_e$) . Unfortunately, these features remain mostly unexplored and show higher variability.

In this work, we present a model that takes advantage of community detection for sentiment classification. The model uses social context in the form of a network of users and content for a topic, where some of the users and content have known sentiment labels. This network is then used to estimate the sentiment of the missing labels for users and content. i.e., it performs both user-level and content-level classification. The estimation is based on maximizing a metric that is inspired by sentiment consistency and homophily theories. Sentiment consistency implies that the sentiment of a user on a given topic is stable over time. The homily theory dictates that similar users are more likely to form connections. In our case, two users are similar if they share the same sentiment on a given topic.

The classification model is based on an earlier model by [2], which our model improves in two significant ways: 1) it can be used for content-level classification, and 2) in addition to using the raw relations from the social network, it can also use community detection to find weak relations between users.

Our proposal is based on the following hypotheses:

Hypothesis 1. *meso features improve user classification in the absence of micro features.*

Hypothesis 2. *micro features improve classification over pure contextless features.*

Hypothesis 3. *meso features improve content classification in the absence of micro features.*

Hypothesis 4. *meso$_e$, and community detection in particular, can improve classification compared to only using meso$_i$ and meso$_r$ features.*

These hypotheses will be tested in the evaluation of the model.

The rest of the paper is structured as follows: Section 2 covers related works and concepts; Section 3 describes the classification model; Section 4 is dedicated to a description of the datasets used for evaluation, and how they have been enriched with social context; Section 5 presents the evaluation of the model; Section 6 closes with our conclusions and future lines of work.

2. Related Work

This section summarizes state of the art in the fields of sentiment analysis and Social Network Analysis (SNA). It also provides a summary of the definitions and nomenclature on social context.

2.1. Sentiment Analysis

Sentiment analysis, or the process of assessing attitude expressed in a text, is hardly a new field, buts its popularity has grown due to the increasing availability and popularity of opinion-rich resources such as online review sites and personal blogs [3].

The approaches in this field can be grouped into three main categories: lexicon-based, machine learning-based, and hybrid [4]. In this section, we will focus on lexicon and machine learning-based approaches, as hybrid approaches use a combination of both.

Lexicon-based approaches are potentially the simplest. They estimate the sentiment of a text using a lexicon, or associations of words in a domain with one or more sentiments. Machine learning approaches apply a predictor on a set of features that represent the input. The predictors used for sentiment analysis are not very different from those used in other areas. The complexity lies extracting useful features from the text, curating them and applying them with the appropriate predictor [5].

Lexicon-based approaches are heavily limited by the quality of the lexicon at hand, and creating consistent and reliable lexicons for a domain is an onerous task [6]. As a consequence, pure lexicon techniques are seldom used. Instead, lexicons typically combined with machine learning techniques [7–11]. Hence, machine learning techniques and hybrid approaches dominate the state of the art [12–14],

Machine learning techniques can use different types of features for their predictions. These features are manually crafted and picked for the specific application. The simplest types of feature, which rely solely on lexical and syntactical information (e.g., bag-of-words, syntactic trees), are often referred to as surface forms. Surface forms can also be combined with other prior information, such as lexicons with word sentiment polarity [7–11]. Some lexicons also include non-words such as emoticons [15,16] and emoji [17]. The combination of the resulting features is fed into a classifier, which can be trained on a known dataset or part of it.

The main disadvantage of these approaches is that each feature needs to be conceived and added by an operator. Although there are processes to select the most informative (i.e., best) features for a given combination of dataset and classifier, the problem of finding and calculating new features still remains.

In contrast, deep learning techniques can automatically learn complex features from data. New approaches based on deep learning have shown excellent performance in Sentiment Analysis in recent years [18,19]. The downside is that they usually require large amounts of data, which is not always available. They also raise other concerns such as interpretability [20,21] or the inability of a model to adapt to deal with edge cases [20]. In the realm of Natural Language Processing (NLP), most of the focus is on learning fixed-length word vector representations using neural language models [22]. These representations, also known as word embeddings, can then be fed into a deep learning classifier, or used with more traditional methods. One of the most popular approaches in this area is word2vec [23]. Although training these models requires enormous amounts of data and fair amounts of computation, there are several publicly available models that have already been trained on large corpora such as Wikipedia.

Lastly, it is also possible to combine independent predictors to achieve a more accurate and reliable model than any of the predictors on their own. This approach is known as ensemble learning. Many ensemble methods have been previously used for sentiment analysis. An exciting new application of ensemble methods is the combination of traditional classifiers based on feature selection and deep learning approaches [12].

2.2. Social Network Analysis

Social Network Analysis (SNA) is the investigation of social structures through a combination of social science and graph theory [24]. It provides techniques to characterize and study the connections and interactions between people, using any kind of social (human) network. The mathematical analysis of social network using graph theory predates the appearance of Online Social Network (OSN) by more than a hundred years. The same techniques have been applied successfully on other types of social networks such as citation networks in academia and call records in mobile networks.

Through SNA techniques, it is possible to extract useful information from a social network, such as chains of influence between users, groups of like-minded users, or metrics of user importance. This information may be useful for many applications, including sentiment analysis. There are several ways in which SNA techniques can be exploited in sentiment analysis, but the analysis of current approaches [1] shows that they can be grouped into one of two categories: those that transform the network into metrics or features that can be used to inform a classifier; and those that limit the analysis to certain groups or partitions of the network.

A simple example of metrics provided by SNA could be user's follower in-degree (number of users that follow the user) and out-degree (number of users followed by the user), which could be used as features for each user [25]. However, these metrics are not very rich, as they only cover users directly connected to a user, and it does so in a very naive way: all connections are treated equally. Other more sophisticated metrics could be used instead of in/out-degree, such as centrality, a measure of the importance of a node within a network topology, or PageRank, an iterative algorithm that weights connections by the importance of the originating user. Several works have introduced alternative metrics for user and content influence in a network [26,27].

The second category of approaches is what is known either as network partition or as community detection, depending on whether the groupings may overlap. Intuitively, community detection aims to find subgroups within a larger group. This grouping can be used to inform a classifier, or to limit the analysis to relevant groups only. More precisely, community detection identifies groups of vertices that are more densely connected to each other than to the rest of the network [28]. The motivation is to reduce the network into smaller parts that still retain some of the features of the bigger network. These communities may be formed due to different factors, depending on the type of link used to connect users, and the technique used to detect the communities. Each definition has its own set of characteristics and shortcomings. For instance, if users are connected after messaging each other, community detection may reveal groups of users that communicate with each other often [29]. By using friendship relations, community detection may also provide the groups of contacts of a user [30].

Other publications [28,31] cover further details of the different definitions of community and algorithms to detect them.

2.3. Social Context

Social context [1] is the collection of users, content, relations, and interactions which describe the environment in which social activity takes place. It encapsulates the frame in which communication in social media takes place.

Social context is used in sentiment analysis for two reasons that are subtly different. First, it can be used to compensate for implicit elements in the text. An example of this is how slang, abbreviations or semantic variations can be detected and accounted for in the classification. Humans apply a similar process when trying to understand content. Content authors also unconsciously rely on this fact and they assume certain prior knowledge. The second motivation to add social context is that it may help correct ambiguity or situations where textual queues are lacking. For example, a classifier may use the sentiment of earlier posts by the user and similar users on the same topic.

For the sake of clarity and for ease of comparison with other works, we will employ the following general definition Social Context [1]:

$$SocialContext = \langle C, U, R, I \rangle \tag{1}$$

Where: U is the set of content generated; C is the set of users; I is the set of interactions between users, and of users with content; R is the set of relations between users, between pieces of content, and between users and content.

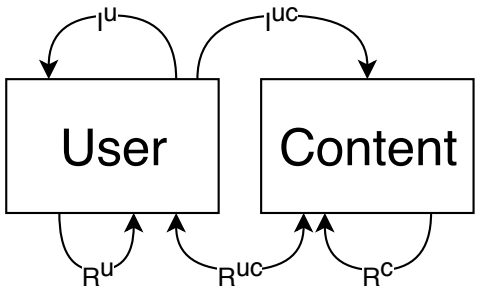


Figure 1. Model of Social Context, including: content (C), users (U), relations (R^c, R^u and R^{uc}), and interactions (I^u and I^{uc}).

Figure 1 provides a graphical representation of the possible links between entities of the two available types. Users may interact (i) with other users (I^u), or with content (I^c).

$$I \equiv \{i_t \mid t \in Ti\} = I^u \cup I^{uc} \tag{2}$$

$$I_t^u = \{i_{t,u_i,u_j,i}^u \mid u_i, u_j \in U, t \in T_{i,u}\} \tag{3}$$

$$I_t^{uc} = \{i_{t,u_i,u_j,i}^{uc} \mid u_i \in U, c_j \in C, t \in T_{i,uc}\} \tag{4}$$

Relations (R) can link any two elements: two users (R^u), a user with content (R^{uc}), or two pieces of content (R^c).

$$R \equiv \{r_t \mid t \in T_r\} = R^u \cup R^{uc} \cup R^c \tag{5}$$

$$R_t^u = \{r_{t,u_i,u_j}^u \mid u_i, u_j \in U, u_i \neq u_j, t \in T_{r,u}\} \tag{6}$$

$$R_t^{uc} = \{r_{t,u_i,c_j}^{uc} \mid u_i \in U, c_j \in C, t \in T_{r,uc}\} \tag{7}$$

$$R_t^c = \{r_{t,c_i,c_j}^c \mid c_i, c_j \in C, c_i \neq c_j, t \in T_{r,c}\} \tag{8}$$

From these definitions, it is obvious that interactions and relations are very similar, and a network of users and content can be created using either one or both of them. In the parts of the model where a relation (R) or an interaction (I) can be used, the term edge (E) can be used instead.

There are countless ways to construct a social context for the piece of text, depending on the types of information included, and how it is gathered. The richness of context influences the type of analysis that can be performed. For the sake of comparison, the ways in which social context is constructed and analyzed can be grouped into one of several categories, according to a taxonomy of approaches [1]. The categories are, from simpler to more complex: *micro* approaches, in which only one user is included along with the content he or she created; *meso* approaches, which also add other users and relations or interactions with them; and *macro* approaches, which include information from outside the OSN, such as facts or encyclopedic knowledge. The *meso* level is further divided: $meso_r$ only use relations; $meso_i$ also include interactions; and $meso_e$ add information from social network analysis, such as partitions, modularity or betweenness.

2.4. Sentiment Analysis using Social Context

This section provides a brief summary of works that have leveraged social context for sentiment analysis, following the taxonomy of approaches by Sánchez-Rada and Iglesias [1].

Tan *et al.* [32] is one of the first works to incorporate social context information, which the authors called heterogeneous graph on topic, to infer (user) sentiment. The underlying ideas behind that work are user consistency and homophily. A function to measure each of those attributes is provided, and the model tries to maximize the overall value. The authors compare alternative ways to construct the user network, using variations of follower-followee relations and direct replies (interactions). However, the approach can be categorized as $meso_r$, for two reasons. Firstly, in their work, relations and interactions yield similar results. Secondly, in the original formulation edges (relations or interactions) are not weighted, so users are influenced equally by all their neighbors. Interactions are bound to be noisy, and aggregating them in this fashion is likely to provide little or no advantage over a simple relation. The SANT model [33] follows similar ideas but for content classification. It is also a $meso_r$ approach that combines sentiment consistency, emotion contagion and a unigram model in a classifier.

Pozzi *et al.* [2] extended the model by Tan *et al.* [32]. Their model uses what they call an approval network, which effectively add weights for edges between users. The rationale for that change is that friendship does not imply approval, and that a weighted network of interactions should better capture emotion contagion. This addition invalidates the two reasons for not considering it a $meso_i$ approach.

Other models have exploited community detection, which includes them into the $meso_e$ category. An example is Xiaomei *et al.* [34], which incorporate weak dependencies between microblogs, using community detection (different algorithms) on a network of microblogs. In their work, microblogs are connected if their authors are (i.e., there is a follower-followee relation).

3. Sentiment classification

The sentiment classification task consists in finding all the sentiment labels for users ($L^u = \{l_i^u \mid u_i \in U\}$) and content ($L^c = \{l_i^c \mid c_i \in C\}$) in a given social context, where the labels of a sub-set of users (B^u) and a sub-set of content (B^c) are known in advance. The social context is made up of a set of content (C), a set of users (U), relations between both users and content (R) and interactions between users and content (I). This is illustrated in Figure 2, where relations and interactions are simplified as undirected edges between nodes (i.e., users and content). For the sake of simplicity, we will only consider two possible labels: Positive and Negative. However, the model can be used with an arbitrary number of labels.

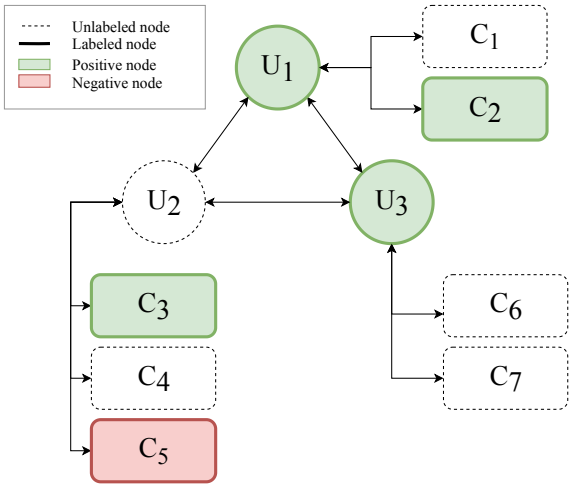


Figure 2. Problem definition. The task is to predict the missing labels.

To solve the classification problem, we propose a classification model that uses a combination of a probability model for a given configuration of user and content labels, and a classification algorithm that finds the set of labels with the highest probability. In other words, we define a metric that, based on a given social context, estimates the likelihood that users and content are labeled in a specific configuration. The metric incorporates homophily and consistency assumptions. It also involves several parameters that need to be adjusted or trained. We propose a classification method that estimates the parameters and the labels at the same time, by employing a modified version of SampleRank [35], an algorithm to estimate parameters in complex graphical models.

Both the probability model and the classification algorithm are based on two earlier works [2,32], which are described in Section 2.4. However, this section does not assume prior knowledge of these models.

3.1. Probability model

In order to find the best configuration of user and content labels, the classification model uses a probability model which estimates the likelihood of a given distribution of user and content labels. This probability model is based on the Markov assumption that the sentiment of user u_i (l_i^u) is influenced only by the sentiment of every piece of content c_i (l_j^c) authored by the user (P_i) and the sentiment labels

of its neighbors in the network (N_i). Likewise, the sentiment of a piece of content c_i (l_i^c) is influenced by the sentiment label of its author. The label of a node (i.e., user or piece of content) may or may not be known in advance. If a label for a node is known, that node is said to be labeled. Labeled users (B^u) and content (B^c) are assigned a higher weight or influence on global probability.

The model is defined as follows. Let l_i^u be the label for user u_i, and let L^u be the vector of labels for all users. Let l_i^c be the label for content u_c, and L^c be the vector of labels for all content. To simplify our notation, we will also use P_i as the subset of content which has been authored by user u_i, and N_i as the subset of users who are connected to user u_i in the social context graph. Two users are connected when there is an edge between them, which can be chosen from the different types of relations and interactions available in the context. i.e., $\{u_i, u_j\} \in E, E \in \{R, I\}$. The probability of a configuration of labels (L^u, L^c) is given by Equation 9:

$$\begin{aligned} log(P(L^u, L^c)) = & \sum_{u_i \in U} \sum_{c_j \in P_i} \mu(l_i^u, l_j^c) \frac{\rho_u(u_i) \cdot \rho_c(c_j)}{|P_i|} \\ & + \sum_{u_j \in N_i} \lambda(l_i^u, l_j^u) \frac{\rho_{neigh} \cdot e_{i,j}}{\sum_{u_k \in N_i} e_{i,k}} \\ & - log(Z) \end{aligned} \tag{9}$$

Where ρ_{neigh} is a constant that controls the weight of the effect of neighboring users, ρ_u and ρ_c determine the weight of each piece of content and each user, respectively, and $e_{i,j}$ is the weight of the edge between neighboring users u_i and u_j. The value of $\mu(\alpha, \beta)$ and $lambda(\alpha, \beta)$ models how a node labeled β affects a node labeled α ($\alpha, \beta \in Polarities$). For the typical case, where $Polarities = \{positive, negative\}$, μ and λ can be thought of as an array with 4 values, one per combination of the two polarities. For instance, the value of $\mu_{positive,positive}$ is the weight given to positive content by positive users.

The weight of a specific user is controlled through ρ_u (Equation 10), and ρ_c (Equation 11) controls the weight of each piece of content. The values of both functions depend on whether the label for the specific user and or content is known a priori. For users with a known sentiment, the weight is $\rho_{labeled}$, and for unknown values, it is $\rho_{unlabeled}$. Based on previous works, we use the following values: $\rho_{u,labeled} = \rho_{c,labeled} = 1$, $\rho_{u,unlabeled} = \rho_{c,unlabeled} = 0.2$ Once again, $e_{i,j}$ is the weight of the edge between users u_i and u_j. Intuitively, this allows for some specific edges to represent stronger bonds and, hence, have a bigger impact on the result. The influence of neighboring agents ρ_{neigh} is a parameter that can be adjusted.

$$\rho_u(u) = \begin{Bmatrix} \rho_{u,labeled} & : \text{if } u \in B^u \\ \rho_{u,unlabeled} & : \text{otherwise} \end{Bmatrix} \tag{10}$$

$$\rho_c(c) = \begin{Bmatrix} \rho_{c,labeled} & : \text{if } u \in B^c \\ \rho_{c,unlabeled} & : \text{otherwise} \end{Bmatrix} \tag{11}$$

3.2. Parameter estimation and classification

Some parameters in the probability model in the previous section are manually set, such as ρ_{neigh} or $\rho_{u,labeled}$, whereas other values are to be calculated. More specifically, the classification process would consist in calculating the values for μ and λ, and then maximizing the log-likelihood of a given distribution of labels (L^u and L^c).

In order to explain the classification process, it is useful to decompose the log-likelihood into a dot product of a matrix of constants and a function of the set of labels:

$$log(P(L^u, L^c)) = \phi \cdot \psi(L^u, L^c) - log(Z) \tag{12}$$

Where ϕ (Equation 13) is constant, and the value of ψ (Equation 13) only depends on the labels and the pre-set parameters. In Equation 13, the μ and λ functions are represented as matrices, where $\mu_{\alpha,\beta} = \mu(\alpha, \beta)$. In Equation 14, we simply introduced an auxiliary function, γ (Equation 15), to separate the summations into components, just like μ and λ.

$$\phi = \{\mu, \lambda\} \tag{13}$$

$$\begin{aligned}\psi(L^u, L^c) = \{ &\sum_{u_i \in U} \sum_{c_j \in P_i} \gamma_{\alpha,\beta}(l_i^u, l_j^c) \frac{\rho_u(u_i) \cdot \rho_c(c_j)}{|P_i|}, \\ &\sum_{u_i \in U} \sum_{u_j \in N_i} \gamma_{\alpha,\beta}(l_i^u, l_j^u) \frac{\rho_{neigh} \cdot e_{i,j}}{\sum_{u_k \in N_i} e_{i,k}} \}\end{aligned} \tag{14}$$

$$\gamma_{\alpha,\beta}(a, b) = \begin{Bmatrix} 1 : a = \alpha \wedge b = \beta \\ 0 : otherwise \end{Bmatrix} \tag{15}$$

The model is thus trained by inferring the values of ϕ, and the Z constant. As we explained earlier, the value of ϕ roughly encodes the expected likelihood of finding a given combination of labels for two nodes. For instance, $\lambda_{positive,positive}$ is the likelihood of positive content on positive users, which is expected to be lower than $\lambda_{negative,positive}$, under assumption of consistency. Once these parameters are calculated for a given domain, the classification consists in maximizing the log-likelihood of a given distribution of labels.

SampleRank can be used to determine the value of ϕ, which is divided into $\mu_{\alpha,\beta}$ and $\lambda_{\alpha,\beta}$. Ideally, the value of Z could be obtained through regularization, but in practice this can be costly. This need can be circumvented by using other methods that calculate the labels for all unknown elements, such as loopy belief propagation. Alternatively, some works exploit the fact that SampleRank can also produce output the set of labels in addition to the value for ϕ [2]. When used in this manner, training can be interpreted as a search in the space of possible labels, and the log-likelihood function is a heuristic that restricts the search. This method has been used successfully for user classification [2], and its main advantage is that it is simpler than using an additional layer of label-propagation.

Our proposed classification algorithm (Algorithm 1) is a modified version of SampleRank, which returns the labels for both users and content.

In this algorithm, the $Random(L_u, L_c)$ function returns a random set of user and content labels (within the range of $Polarities$, which in a simple case would just be negative and positive). E_u represents edges between users, i.e., either relations or interactions. The $CD(E_u)$ function performs community detection given a set of edges, and returns the set of edges between all users within the same community. In particular, we are using the Louvain method [36]. The $Sample(L_u, L_c)$ function changes one of the labels from either L_u or L_c, at random. Since the SampleRank algorithm is inherently stochastic, the model should be run several times, and the results of each run should be aggregated. In our case, we use a number of 21 iterations, based on earlier works [32], and simple majority over all iterations.

4. Data

4.1. Datasets

Table 1 provides basic information about the datasets used in the evaluation. Since the model used in this work requires a social context with interactions or relations, the list is limited to datasets that either contained this information or that could be extended using other sources (Section 4.2).

The OMD dataset (Obama-McCain debate) [37] contains tweets about the televised debate between Senator John McCain, and then-Senator Barack Obama. The tweets were detected by following three

Algorithm 1 Sentiment Detection

Input
$B_u : \{(u, p) \mid u \in U, p \in P\}$
$B_c : \{(c, p) \mid c \in C, p \in P\}$
$E_u : \{(i, j) \mid i, j \in U\}$
$E_{uc} : \{(u, c) \mid u \in U, c \in C\}$
$P : L^N \to \mathbb{R}$
$\psi : L^N \times P^N \to \mathbb{R}$
Output
L_u, estimated user labels.
L_c, estimated content labels.
ϕ, learned weights.

$E_u \leftarrow CD(E_u)$ ▷ Community detection. This is skipped in CrankNoComm
$L_u, L_c \leftarrow Random(L_u, L_c)$
$Stale \leftarrow 0$
for $step \leftarrow 1$ to $MaxSteps$ **do**
  $L_{unew}, L_{cnew} \leftarrow Sample(L_u, L_c)$ ▷ Randomly modify only one label
  $\nabla \leftarrow \psi(L_{unew}, L_{cnew} B_u, B_c, E_u, E_{uc}) - \psi(L_u, L_c, B_u, B_c, E_u, E_{uc})$
  $\Delta P \leftarrow P(L_{unew}, L_{cnew}) - P(L_u, L_c)$
  if $\phi \cdot \nabla > 0 \wedge \Delta P < 0$ **then** ▷ Performance is worse, objective is better
    $\phi \leftarrow \phi - \eta \nabla$ ▷ Performance is better, objective is worse
  else if $\phi \cdot \nabla < 0 \wedge \Delta P > 0$ **then**
    $\phi \leftarrow \phi + \eta \nabla$ ▷ Converge if there are no changes in a given number of steps
  if $\nabla \leq 0 \wedge P(L_{cnew}, L_u) \leq 0$ **then**
    $Stale \leftarrow Stale + 1$
    if $Stale >= Convergence$ **then return**
  else
    $Stale \leftarrow 0$
  if $\Delta P > 0 \vee (\Delta P = 0 \wedge \phi \cdot \nabla > 0)$ **then** ▷ Performance is better, and objective function is at least the same
    $L_u \leftarrow L_{unew}$
    $L_c \leftarrow L_{cnew}$

Table 1. Datasets used in the experiments

	Source	Users	Entries	Year
OMD [37]	Twitter	893	1261	2009
HCR [38]	Twitter	277	1434	2011
RT Mind [2]	Twitter	62	159	2013

hashtags: *#current*,*#tweetdebate*, and *#debate08*. The dataset contains tweets captured during the 97-minute debate, and 53 after it, to a total of 2.5 hours. The dataset includes tweet IDs, publication date, text, author name and nickname, and individual annotations of up to 7 annotators.

The Health Care Reform (HCR) [38] dataset contains tweets about the run-up to the signing of the health care bill in the USA on March 23, 2010. It was collected using the *#hcr* hashtag, from early 2010. A subset of the collected tweets were annotated with polarity (positive, negative, neutral and irrelevant) and polarity targets (health care reform, Obama, Democrats, Republicans, Tea Party, conservatives, liberals, and Stupak) by Speriosu *et al.* [38]. The tweets were separated into training, dev (HCR-DEV) and test (HCR-TEST) sets. The dataset contains tweet ID, user ID and username, text of the tweet, sentiment, target of the sentiment, annotator and annotator ID.

RT Mind [2] contains a set of 62 users and 159 tweets, with positive or negative annotations. To collect this dataset, Pozzi *et al.* [2] crawled 2500 Twitter users who tweeted about Obama during two days in May 2013. For each user, their recent tweets (up to 3200, the limit of the API) were collected. At that point, only users that tweeted at least 50 times about Obama were considered. The tweets from those users that relate to Obama were kept and manually labeled by 3 annotators. Then, a synthetic network of following relations was generated based on a homophily criterion. i.e., users with a similar sentiment are more likely to be connected. The dataset contains ID of the tweet, ID of the author, text of the tweet, creation time, and sentiment (positive or negative).

4.2. Gathering and analyzing social context

The model proposed needs to access the network of users. Since all datasets provide both tweet and user IDs, it would be possible to access Twitter's public API to retrieve the network. However, that approach has several disadvantages that stem from the fact that these datasets were originally captured circa 2010 [1], such as the fact that the relationships between users have likely changed, and that many of the original tweets and users have been deleted or made private, making it impossible to fetch them. Alternatively, we decided to retrieve the follower network from a snapshot of the whole Twitter network in summer of 2009 [39]. Since the datasets used were gathered around the same time period as the snapshot, this should provide a more reliable list of followers than other methods. We refer to the the resulting network as *relations*.

Upon realizing that the *relations* network is rather sparse for the OMD and HCR datasets, we investigated an alternative to find hidden links between users: connecting users that follow similar people. To do so, we extracted the list of users followed by each author, and we compared the list of followees for each pair of users in the dataset. Users that share at least a given ratio of their followees were considered similar, and an edge between them was drawn. After evaluating different values for the threshold ratio, it was set to 15%, as it results in a degree similar to the RT Mind dataset. We refer to this network as *common*.

To compare the two network variants, *relations* and *common*, we will use some basic statistics of each network, shown in Table 2. The table includes the average degree of each node in the network (i.e., mean number of edges per node), the ratio of users that have the same label as the majority of their neighbors in the network (majority agreement), the ratio of users that have the same label as all their neighbors (total agreement), and the ratio of users that do not have any neighbors. The degree measures the density of the network. The majority and total agreement metrics are a measure of homophily in the network.

We observe that the RT Mind dataset is the most promising of all the networks, as it has high density and homophily, higher content count per user, and all of its users are connected. The OMD networks are the densest, but their agreement is very low, and a fourth of its users are not connected to others. Moreover, we observe that the *common* extension of this dataset has a lower agreement ratio and fewer edges, whereas the isolation ratio remains the same as in the relations network. Lastly, the HCR dataset shows the lowest agreement of the datasets, and the relations network is almost non-existent. Although the common network significantly improves every metric, the majority agreement is still

dataset	variant	content mean	content median	degree	isolation ratio	majority agreement	# edges	# nodes	total agreement
RT Mind	relations	2.56	3.00	8.61	0.00	0.90	267	62	0.52
OMD	relations	2.56	1.00	14.25	0.24	0.39	6364	893	0.16
	common	2.56	1.00	9.59	0.24	0.30	4280	893	0.15
HCR	relations	1.21	1.00	0.02	0.99	0.01	3	277	0.01
	common	1.21	1.00	2.89	0.80	0.19	400	277	0.18

Table 2. Statistics of the networks gathered for each dataset.

very low (0.29). This means that the additional links are connecting users that are dissimilar, which negates the homophily assumption.

In summary, we conclude that this particular strategy to extend social context does not work for these datasets. The statistics for the RT Mind dataset make it ideal for the evaluation of our proposed model. The results for the OMD dataset may indicate how the model works in scenarios with a higher degree, but relatively low homophily. In that scenario, the *meso* features may interfere with *micro* features. And, lastly, the HCR dataset could show how the model works with an almost complete lack of *meso* features.

5. Evaluation

The sentiment classification task can be divided into two sub-tasks: user-level classification, which only focuses on predicting user labels (L^u); and content-level classification, which focuses on content labels (L^c). Since these two tasks are seldom tackled at the same time, we will evaluate how the model performs in each of them independently. The datasets used have been described in Sec. 4.

First, we focus on user-level classification (Sec. 5.1). The main goal is to evaluate the effect of adding community detection to the samplerank algorithm, and to compare the performance of the model to others. Then, we evaluate content-level classification (Sec. 5.2) with varying levels of certainty about user and content labels.

We will compare the performance of CRANK to other classifiers that will serve as the baseline, and to the results of other works in the state of the art. Each model will be evaluated on different scenarios, i.e., different social contexts. The ratio of labeled (i.e., known) users and content has a significant impact on the performance of the model. Thus, we have evaluated each model with different ratios of known labels for both users ($ratio_u$) and content ($ratio_c$). In each scenario, a random set of labels has been kept, according to $ratio_u$ and $ratio_c$. This process has been repeated several times to ensure that the results are not too biased by the random partition. For each combination of *model*, *dataset*, $ratio_u$ and $ratio_c$, the results are aggregated and the mean accuracy and its standard deviation are calculated.

5.1. User-level classification

For the evaluation of user classification we wanted to test whether Hypotheses 1 and 4 hold. i.e., whether *meso* features improve accuracy over *micro* features (Hypothesis 1), and whether $meso_e$ features improve it even further (Hypothesis 4). In our case, Hypothesis 1 is tested by comparing the accuracy of the CRANK model to a simpler model that labels each users using the majority label of their content. Hypothesis 4 is tested by comparing the CRANK model to CRANK without community detection.

The following models were compared:

- Average Content (AvgContent) (*micro*) content is applied the same label as the majority of content by the same user, and users are labeled according to the majority label of their content.

- Naive majority (AvgNeigh)($meso_i$ or $meso_r$, depending on the context). Users are labeled with the majority label in their group of neighbors in the network. Unlabeled content is given the label of its creator.
- Majority in the community (AvgComm) ($meso_e$). Users are grouped into communities, and each user is given the majority label of the users in their community. Content is given the label of its creator.
- CRANK without community detection ($meso_r$ or $meso_i$, depending on the context). The CRANK model described in Algorithm 1, but using original edges instead of applying community detection.
- CRANK ($meso_e$). Before applying Algorithm 1, the communities between users are extracted and converted to user edges. i.e., users in the same community are connected by an edge.

The results of the evaluation are shown in Table 3, where the highest value for each row is presented in bold. It also highlights in grey the highest value when the Average Content is ignored.

If we focus on the results for the RT Mind dataset, we conclude that CRANK significantly improves the classification in all scenarios, especially with lower $ratio_c$ values. In other datasets, where the network of users is more sparse and less cohesive, CRANK outperforms all the models, except for the average of content. This is expected, since *meso* features in these datasets are rather weak, and the content mean and median values are close to 1. In particular, the difference between the CRANK model and the baseline in the HCR dataset is is relatively small (.02). That indicates that there is little penalty to using CRANK even when there are few *meso* edges between users. In the OMD dataset, which had low agreement between neighbors, the difference between CRANK and the baseline is higher, and it does not decrease with higher values of $ratio_u$. This confirms our suspicions that the *meso* features in this dataset are not useful for our purposes.

Regarding Hypothesis 4, we observe that CRANK outperforms its variant without community detection in most of the cases. The exceptions are cases where most of the user labels are known. In those cases, the accuracy of both methods is extremely high (above 0.95). This difference can be explained by interpreting community detection as an aggregate over several users. In general, all the users in a community share the same sentiment. But some members will have a different label from the majority in their community (i.e., outliers). Often those outliers are users that are connected to users of other communities with a different sentiment. That information is lost when aggregating, so for those outliers community detection is actually detrimental. The fewer users that are left unlabelled, the higher the effect of those outliers will be. Aggregating in those cases present higher variance which, combined with the high accuracy values, also lowers the mean compared to not aggregating. Nevertheless, we can conclude that $meso_e$ features improve user classification in most cases.

5.2. Content-level classification

In the context-level task, the following classifiers were used:

- Simon [40] (*contextless*), a sentiment analysis model based on semantic similarity. The model can be trained with different datasets. In our evaluation, we compared with the Simon model trained on different datasets: STS, Vader, Sentiment140, and a combination of all three.
- Sentiment140 [1] service (*contextless*). This is a public sentiment analysis service, tailored for Twitter. It outputs three labels: positive, negative and neutral. This results in lower accuracy for the negative and positive labels. In fact, of all the models tested, this is the one with the lowest accuracy. If all tweets labeled neutral by the service are ignored, its accuracy reaches standard levels (around 60%). Unfortunately, this means that around 80% of tweets have to be ignored.

[1] https://www.sentiment140.com

Table 3. User-level classification accuracy for each model.

dataset	$ratio_c$	model $ratio_u$	AvgComm	AvgContent	AvgNeigh	CRANK	CrankNoComm
RT Mind	0.25	0.25	.536	.692	.540	**.883**	.815
		0.50	.661	.670	.651	**.950**	.939
		0.75	.954	.642	.791	.962	**.985**
	0.50	0.25	.536	.860	.540	**.933**	.828
		0.50	.663	.843	.651	**.964**	.961
		0.75	.951	.861	.791	.965	**.985**
HCR	0.25	0.25	.597	**.713**	.597	.681	.660
		0.50	.608	**.712**	.607	.698	.681
		0.75	.636	**.742**	.636	.697	.684
	0.50	0.25	.597	**.807**	.597	.789	.789
		0.50	.610	**.816**	.610	.795	.791
		0.75	.636	**.814**	.636	.796	.767
OMD	0.25	0.25	.701	**.756**	.699	.710	.674
		0.50	.706	**.763**	.704	.720	.706
		0.75	.703	**.763**	.699	.724	.708
	0.50	0.25	.702	**.811**	.700	.712	.684
		0.50	.706	**.811**	.705	.736	.724
		0.75	.701	**.819**	.699	.731	.731

- Meaningcloud [2] Sentiment Analysis (*contextless*), an enterprise service that provides several types of text analysis, including sentiment analysis. It poses the same restrictions for evaluation as Sentiment140, as it provides positive, negative and neutral labels. Fortunately, the subjectivity detection of this service for our datasets is better than that of Sentiment140.
- Average Content (AvgContent) (*micro*) content is applied the same label as the majority of content by the same user, and users are labeled according to the majority label of their content.
- Naive majority (AvgNeigh) ($meso_i$ or $meso_r$, depending on the context). Users are labeled with the majority label in their group of neighbors in the network. Unlabeled content is given the label of its creator.
- Majority in the community (AvgComm) ($meso_e$). Users are grouped into communities, and each user is given the majority label of the users in their community. Content is given the label of its creator.
- CRANK without community detection ($meso_r$ or $meso_i$, depending on the context). The CRANK model described in Algorithm 1, but using original edges instead of applying community detection.
- CRANK ($meso_e$). Before applying Algorithm 1, the communities between users are extracted and converted to user edges. i.e., users in the same community are connected by an edge.
- Label propagation [38] (Speriosu), based on the results reported in the original paper for these datasets.

We compared the accuracy of each of these models at several combinations of known content and user labels ($ratio_c$ and $ratio_u$). Table 4 shows a summary of the mean accuracy for each combination.

[2] https://www.meaningcloud.com/

Table 4. Content-level classification accuracy of each model.

dataset	$ratio_u$	algo / $ratio_c$	AvgComm	AvgContent	AvgNeigh	CRANK	CrankNoComm	meaningcloud	sentiment140	simon_all_train_data	simon_sentiment140	simon_sts	simon_vader
RT Mind	0.25	0.25	.56	.65	.56	**.88**	.81	.51	.59	.56	.56	.62	.58
		0.50	.57	.78	.58	**.89**	.81	.54	.60	.57	.57	.64	.60
		0.75	.54	.76	.54	**.85**	.80	.51	.58	.53	.53	.57	.52
	0.50	0.25	.69	.65	.64	**.90**	**.90**	.51	.59	.56	.56	.62	.58
		0.50	.69	.78	.64	**.90**	**.90**	.54	.60	.57	.57	.64	.60
		0.75	.67	.76	.61	.88	**.89**	.51	.58	.53	.53	.57	.52
	0.75	0.25	.89	.65	.78	.91	**.92**	.51	.59	.56	.56	.62	.58
		0.50	.88	.78	.78	**.91**	**.91**	.54	.60	.57	.57	.64	.60
		0.75	.85	.76	.76	.88	**.90**	.51	.58	.53	.53	.57	.52
HCR	0.25	0.25	.63	.64	.63	**.69**	.67	.60	.62	.65	.65	.66	.57
		0.50	.62	.64	.62	**.70**	**.70**	.59	.62	.65	.66	.65	.57
		0.75	.61	.65	.61	**.73**	.71	.59	.57	.63	.63	.64	.56
	0.50	0.25	.63	.64	.63	**.80**	.78	.60	.62	.65	.65	.66	.57
		0.50	.62	.64	.62	**.80**	.79	.59	.62	.65	.66	.65	.57
		0.75	.61	.65	.61	**.80**	**.80**	.59	.57	.63	.63	.64	.56
	0.75	0.25	.63	.64	.63	**.90**	.89	.60	.62	.65	.65	.66	.57
		0.50	.62	.64	.62	**.89**	.88	.59	.62	.65	.66	.65	.57
		0.75	.61	.65	.61	**.89**	**.89**	.59	.57	.63	.63	.64	.56
OMD	0.25	0.25	.64	.61	.64	.64	.62	.69	.63	.65	.65	**.70**	.64
		0.50	.64	.60	.63	.64	.62	**.70**	.64	.65	.65	.69	.63
		0.75	.64	.61	.63	.65	.63	.67	.62	.66	.66	**.70**	.63
	0.50	0.25	.64	.60	.64	.67	.64	.69	.63	.65	.65	**.70**	.64
		0.50	.63	.60	.63	.67	.65	**.70**	.64	.65	.65	.69	.63
		0.75	.63	.61	.63	.68	.66	.67	.62	.66	.66	**.70**	.63
	0.75	0.25	.64	.61	.63	.69	.67	.69	.63	.65	.65	**.70**	.64
		0.50	.63	.60	.63	.69	.68	**.70**	.64	.65	.65	.69	.63
		0.75	.63	.61	.63	**.71**	.70	.67	.62	.66	.66	.70	.63

We also provide a graph of the mean accuracy and standard deviation of each model at (Fig. Fig. 3, Fig. 4 and Fig. 5).

Similarly to the user-classification case, if we focus on the RT Mind dataset, the CRANK algorithm outperforms all other models by a wide margin. In general, the baseline models that use social context have higher accuracy in this dataset than any contextless approach. This is more obvious when either more content is known (better *micro* features), or more users are known (better *meso* features). This evidence supports Hypotheses 2 and 3.

In this case, averaging the content of a user yields poor results for all datasets, due to the low content count per user. If we look at all the results, we observe once again that the version of CRANK with community detection has consistently better accuracy, supporting Hypothesis 4. It should be noted that the Simon model [40] achieves the best performance among the contextless models and the overall best in the OMD dataset. Unfortunately, the results for that dataset are very similar for all the models, and the margins are small, so we cannot draw any conclusions from that dataset.

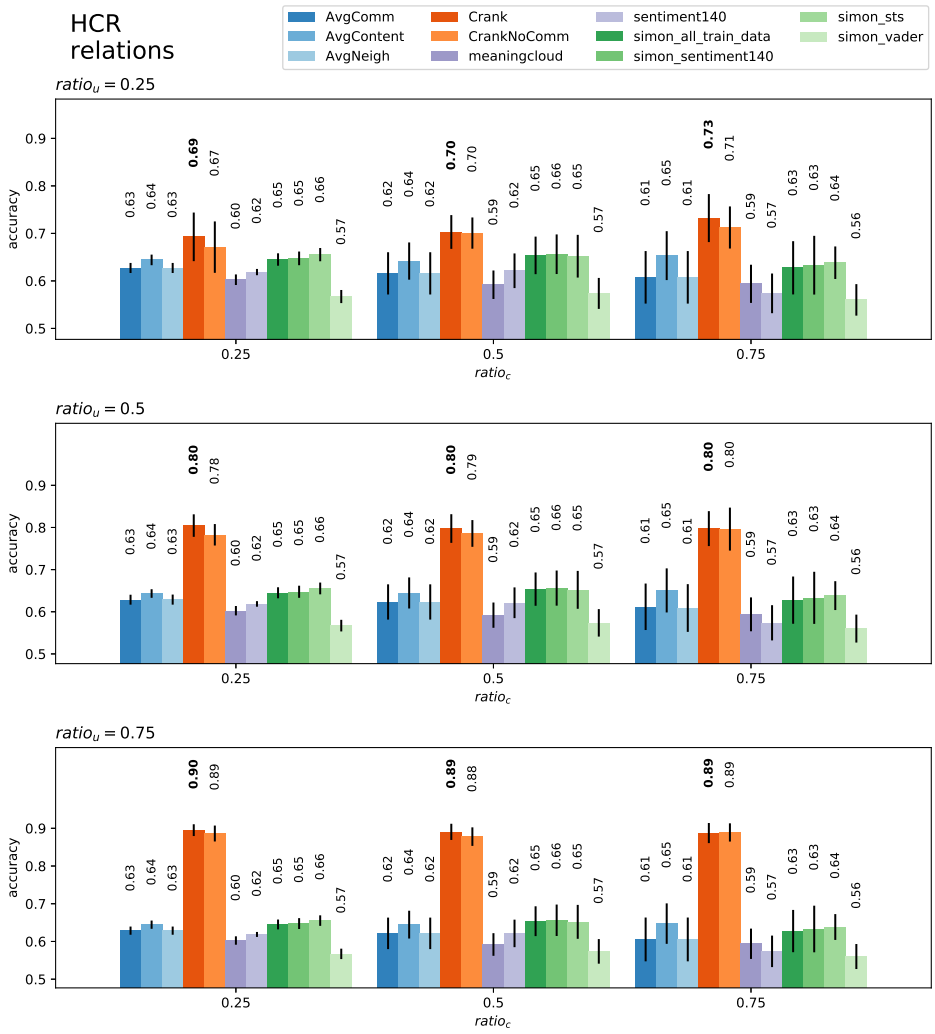


Figure 3. Content-level classification mean accuracy and standard deviation in the HCR dataset for each model at each level of certainty ($ratio_u$ and $ratio_c$)

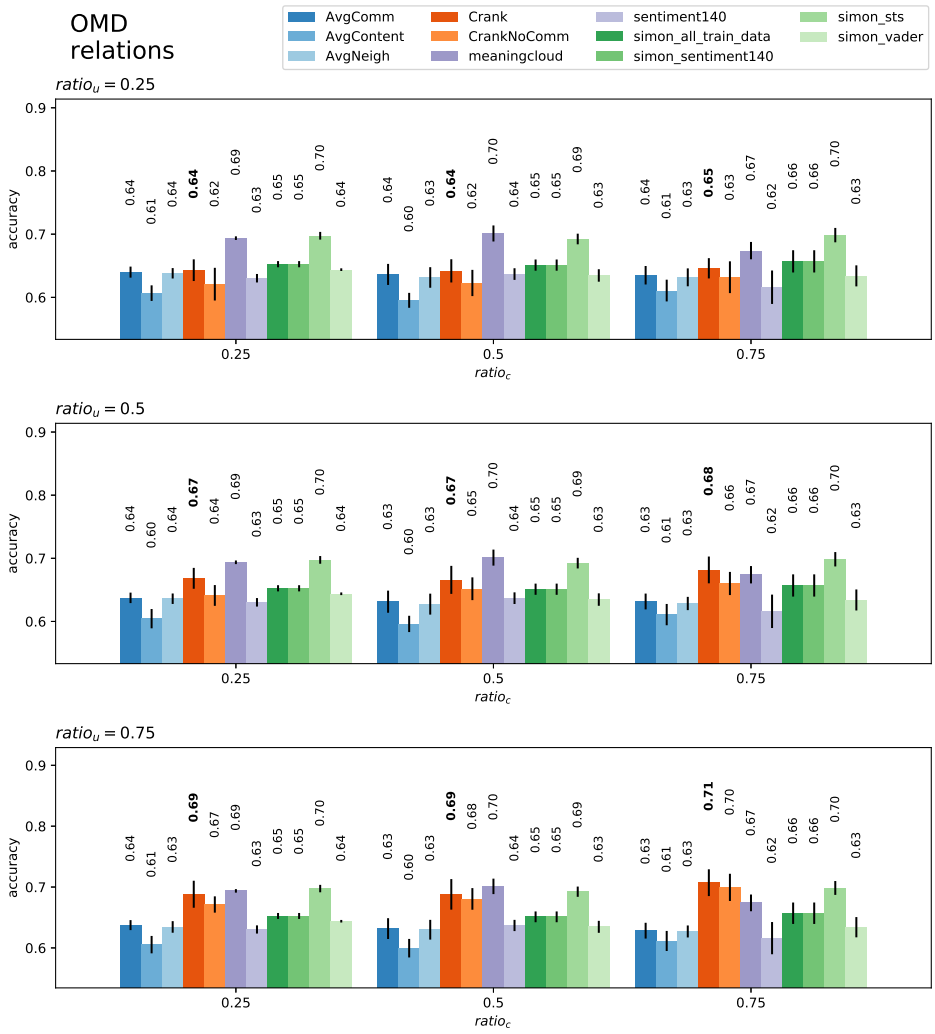


Figure 4. Content-level classification mean accuracy and standard deviation in the OMD dataset for each model at each level of certainty ($ratio_u$ and $ratio_c$)

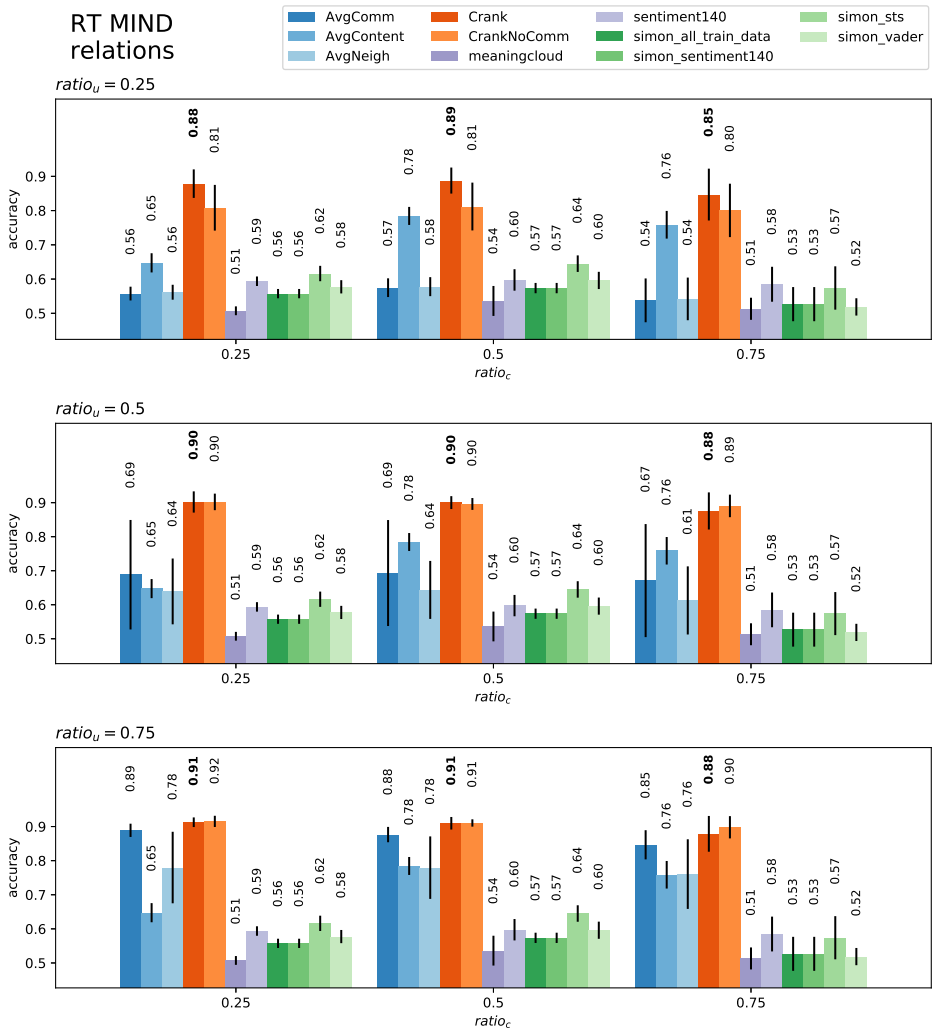


Figure 5. Content-level classification mean accuracy and standard deviation in the RT Mind dataset for each model at each level of certainty ($ratio_u$ and $ratio_c$)

	sentiment140	AvgContent	CRANK	CrankNoComm	Speriosu	meaningcloud	simon_all_train_data	simon_sentiment140	simon_sts	simon_vader
Pass	Baseline	No	Yes	Yes	No	No	No	No	Yes	No
Diff.	0.0	0.81	4.96	4.04	1.52	0.56	0.85	1.19	3.52	-1.0

Table 5. Ranking from Friendman's test in content-level classification

5.3. Statistical analysis

In order to assess the value of the comparison of the models, a statistical test has been to on the experimental results. More specifically, we used a combination of Friedman's test with the corresponding Bonferroni-Dunn post-hoc test, which is oriented to the comparison of several classifiers on multiple data sets [41].

First of all, in section 5.1, we claim that the version of CRANK with community detection outperforms the version without it. To assess that claim, we have compared all the user and content-level classification cases for both models. Friedman's test reveals the difference between both models is statistically different, with a chi-squared of 104, and a p-value of $2.9e^{-5}$. The post-hoc Bonferroni-Dunn test also passes with a calculated difference of 0.63, which is above a critical difference of 0.27.

Secondly, we compare all the user-level models, ignoring the Average Content classifier. In that case, Friedman's test also rejects the null hypothesis, with a chi-squared of 27.4, and a p-value of 0.0006. In this case, we performed the Bonferroni-Dunn test, with Average of Neighbors as the baseline, and it both CRANK and CRANK without communities pass it. The results for Average in Community and Average of Neighbors are not conclusive.

Secondly, we performed a similar comparison for content-level classification. We compared the following approaches to the sentiment140 baseline. The calculated critical difference for this case is 3.299. The results are that only CRANK, CRANK without communities and Simon trained with the STS dataset are better than the baseline (Table 5). Unfortunately, we cannot reject the null hypothesis for CRANK and Simon STS alone at the desired level of confidence, given the number of datasets. Nevertheless, if we reduce our test to the scenarios with the RT Mind dataset at different ratios of r_u and r_c, the null hypothesis can be rejected with $\alpha = 0.1$.

6. Conclusions and future work

In this work, we have proposed a model that unites features from different levels of social context (*micro*, *meso* and $meso_e$). This model is an extension of earlier models that were limited to user-level classification. Moreover, it employs community detection, which finds weak relationships between users that are not directly connected in the network. We expected the combination to have an advantage at different levels of certainty about the labels in the context, and with varying degrees of sparsity in the social network. The proposed model has been shown to work for both types of classification in different scenarios.

To evaluate the model, we looked at different datasets. The need for a social context has restricted the number of datasets that could be used in the evaluation. Of the three datasets included, the RT Mind dataset seems to be the most appropriate, as it contains a more densely connected network of users. The results of evaluating CRANK other baseline models in that dataset provide limited support for Hypothesis 4 ($meso_e$ features improve user classification). Moreover, the evidence from evaluating all the datasets supports Hypotheses 2 (*micro* features improve content classification) and 3

(*meso* features improve content classification). By comparing the two versions of CRANK (with and without community detection) in both user and content-level classification, we have also validated Hypothesis 4 ($meso_e$ features improve user and content classification), Nonetheless, the analysis of the datasets in Section 4.2 reveals the need for better datasets, which can be enriched with context. i.e., datasets with inter-connected users and more content per user. Hence, further evaluation would be needed, once richer datasets become available.

In addition to evaluating in more domains and datasets, there are several lines of future research. In this work, we used a random user and content selection strategy to generate the evaluation datasets. A random sampling strategy for users and content leads to higher sparsity. Since the performance of the model depends on having a densely connected graph, it would be interesting to evaluate effect of different sampling algorithms, such as random walk, breadth-first search, and depth-first search. In particular, breadth-first search (BFS) sampling may be more appropriate for this scenario [42].

It would also be interesting to analyse different community detection strategies. The simplest improvement in this regard would be using other community detection algorithms. There are several methods that produce overlapping partitions, which may help alleviate the negative effect of users in the edge of two communities. More sophisticated strategies are also possible, such as automatically deciding to apply community detection based on the network and the ratio of known users, or only adding edges for some users.

Acknowledgments

This work is supported by the Spanish Ministry of Economy and Competitiveness under the R&D project SEMOLA (TEC2015-68284-R) and the European Union under the project Trivalent (H2020 Action Grant No. 740934, SEC-06-FCT-2016). The authors also want to mention earlier work that contributed to the results in this paper. More specifically, the MixedEmotions (European Union's Horizon 2020 Programme research and innovation programme under grant agreements No. 644632) project.

1. Sánchez-Rada, J.F.; Iglesias, C.A. Social Context in Sentiment Analysis: Formal Definition, Overview of Current Trends and Framework for Comparison. *Information Fusion* **2019**. doi:10.1016/j.inffus.2019.05.003.
2. Pozzi, F.A.; Maccagnola, D.; Fersini, E.; Messina, E. Enhance user-level sentiment analysis on microblogs with approval relations. Congress of the Italian Association for Artificial Intelligence. Springer, 2013, pp. 133–144.
3. Pang, B.; Lee, L. Opinion Mining and Sentiment Analysis. *Foundations and Trends® in Information Retrieval* **2008**, *2*, 1–135.
4. Ravi, K.; Ravi, V. A survey on opinion mining and sentiment analysis: Tasks, approaches and applications. *Knowledge-Based Systems* **2015**, *89*, 14–46.
5. Sharma, A.; Dey, S. A comparative study of feature selection and machine learning techniques for sentiment analysis. Proceedings of the 2012 ACM research in applied computation symposium. ACM, 2012, pp. 1–7.
6. Taboada, M.; Brooke, J.; Tofiloski, M.; Voll, K.; Stede, M. Lexicon-Based Methods for Sentiment Analysis. *Computational Linguistics* **2011**, *37*, 267–307.
7. García-Pablos, A.; Cuadros Oller, M.; Rigau Claramunt, G. A comparison of domain-based word polarity estimation using different word embeddings. Proceedings of the Tenth International Conference on Language Resources and Evaluation; , 2016.
8. Cambria, E. Affective computing and sentiment analysis. *IEEE Intelligent Systems* **2016**, *31*, 102–107.
9. Kiritchenko, S.; Zhu, X.; Mohammad, S.M. Sentiment Analysis of Short Informal Texts. *Journal of Artificial Intelligence Research* **2014**, *50*, 723–762.
10. Melville, P.; Gryc, W.; Lawrence, R.D. Sentiment Analysis of Blogs by Combining Lexical Knowledge with Text Classification. Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining; ACM: New York, NY, USA, 2009; KDD '09, pp. 1275–1284.

11. Nasukawa, T.; Yi, J. Sentiment Analysis: Capturing Favorability Using Natural Language Processing. Proceedings of the 2Nd International Conference on Knowledge Capture; ACM: New York, NY, USA, 2003; K-CAP '03, pp. 70–77.
12. Araque, O.; Corcuera-Platas, I.; Sánchez-Rada, J.F.; Iglesias, C.A. Enhancing Deep Learning Sentiment Analysis with Ensemble Techniques in Social Applications. *Expert Systems with Applications* **2017**.
13. Pang, B.; Lee, L.; Vaithyanathan, S. Thumbs Up?: Sentiment Classification Using Machine Learning Techniques. Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing - Volume 10; Association for Computational Linguistics: Stroudsburg, PA, USA, 2002; EMNLP '02, pp. 79–86.
14. Wang, S.; Manning, C.D. Baselines and Bigrams: Simple, Good Sentiment and Topic Classification. Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers - Volume 2; Association for Computational Linguistics: Stroudsburg, PA, USA, 2012; ACL '12, pp. 90–94.
15. Jiang, F.; Liu, Y.Q.; Luan, H.B.; Sun, J.S.; Zhu, X.; Zhang, M.; Ma, S.P. Microblog sentiment analysis with emoticon space model. *Journal of Computer Science and Technology* **2015**, *30*, 1120–1129. 00026.
16. Hogenboom, A.; Bal, D.; Frasincar, F.; Bal, M.; De Jong, F.; Kaymak, U. Exploiting Emoticons in Polarity Classification of Text. *J. Web Eng.* **2015**, *14*, 22–40. 00043.
17. Novak, P.K.; Smailović, J.; Sluban, B.; Mozetič, I. Sentiment of emojis. *PloS one* **2015**, *10*, e0144296. 00226.
18. Collobert, R.; Weston, J.; Bottou, L.; Karlen, M.; Kavukcuoglu, K.; Kuksa, P. Natural language processing (almost) from scratch. *The Journal of Machine Learning Research* **2011**, *12*, 2493–2537.
19. Bengio, Y. Learning Deep Architectures for AI. *Foundations and Trends® in Machine Learning* **2009**, *2*, 1–127.
20. Marcus, G. Deep learning: A critical appraisal. *arXiv preprint arXiv:1801.00631* **2018**.
21. Lipton, Z.C. The mythos of model interpretability. *arXiv preprint arXiv:1606.03490* **2016**.
22. Kim, Y. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882* **2014**.
23. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* **2013**.
24. Otte, E.; Rousseau, R. Social network analysis: a powerful strategy, also for the information sciences. *Journal of Information Science* **2002**, *28*, 441–453.
25. Sixto, J.; Almeida, A.; López-de Ipiña, D. Analysis of the Structured Information for Subjectivity Detection in Twitter. *Transactions on Computational Collective Intelligence XXIX* **2018**, pp. 163–181.
26. Hajian, B.; White, T. Modelling Influence in a Social Network: Metrics and Evaluation. 2011 IEEE Third International Conference on Privacy, Security, Risk and Trust and 2011 IEEE Third International Conference on Social Computing, 2011, pp. 497–500.
27. Noro, T.; Tokuda, T. Searching for Relevant Tweets Based on Topic-related User Activities. *J. Web Eng.* **2016**, *15*, 249–276.
28. Papadopoulos, S.; Kompatsiaris, Y.; Vakali, A.; Spyridonos, P. Community detection in social media. *Data Mining and Knowledge Discovery* **2012**, *24*, 515–554.
29. Deitrick, W.; Hu, W. Mutually Enhancing Community Detection and Sentiment Analysis on Twitter Networks. *Journal of Data Analysis and Information Processing* **2013**, *01*, 19–29.
30. Gao, B.; Berendt, B.; Clarke, D.; De Wolf, R.; Peetz, T.; Pierson, J.; Sayaf, R. Interactive grouping of friends in OSN: Towards online context management. Data Mining Workshops (ICDMW), 2012 IEEE 12th International Conference on. IEEE, 2012, pp. 555–562.
31. Orman, G.K.; Labatut, V.; Cherifi, H. Qualitative comparison of community detection algorithms. International conference on digital information and communication technology and its applications. Springer, 2011, pp. 265–279.
32. Tan, C.; Lee, L.; Tang, J.; Jiang, L.; Zhou, M.; Li, P. User-level Sentiment Analysis Incorporating Social Networks. Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining; ACM: New York, NY, USA, 2011; KDD '11, pp. 1397–1405.
33. Hu, X.; Tang, L.; Tang, J.; Liu, H. Exploiting Social Relations for Sentiment Analysis in Microblogging. Proceedings of the Sixth ACM International Conference on Web Search and Data Mining; ACM: New York, NY, USA, 2013; WSDM '13, pp. 537–546.
34. Xiaomei, Z.; Jing, Y.; Jianpei, Z.; Hongyu, H. Microblog sentiment analysis with weak dependency connections. *Knowledge-Based Systems* **2018**, *142*, 170–180.

35. Wick, M.L.; Rohanimanesh, K.; Bellare, K.; Culotta, A.; McCallum, A. SampleRank: Training Factor Graphs with Atomic Gradients. ICML, 2011, Vol. 5, p. 1. 00052.
36. Blondel, V.D.; Guillaume, J.L.; Lambiotte, R.; Lefebvre, E. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment* **2008**, *2008*, P10008.
37. Shamma, D.A.; Kennedy, L.; Churchill, E.F. Tweet the Debates: Understanding Community Annotation of Uncollected Sources. Proceedings of the First SIGMM Workshop on Social Media; ACM: New York, NY, USA, 2009; WSM '09, pp. 3–10.
38. Speriosu, M.; Sudan, N.; Upadhyay, S.; Baldridge, J. Twitter Polarity Classification with Label Propagation over Lexical Links and the Follower Graph. Proceedings of the Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2011, pp. 53–56.
39. Kwak, H.; Lee, C.; Park, H.; Moon, S. What is Twitter, a Social Network or a News Media? Proceedings of the 19th International Conference on World Wide Web; ACM: New York, NY, USA, 2010; WWW '10, pp. 591–600.
40. Araque, O.; Zhu, G.; Iglesias, C.A. A semantic similarity-based perspective of affect lexicons for sentiment analysis. *Knowledge-Based Systems* **2019**, *165*, 346–359.
41. Demšar, J. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine learning research* **2006**, *7*, 1–30. 07790.
42. West, R.; Paskov, H.S.; Leskovec, J.; Potts, C. Exploiting Social Network Structure for Person-to-Person Sentiment Analysis. *CoRR* **2014**, *abs/1409.2450*.

© 2020 by the authors. Submitted to *Appl. Sci.* for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).

3.3.3 A Model of Radicalization Growth using Agent-based Social Simulation

Title	A Model of Radicalization Growth using Agent-based Social Simulation
Authors	Méndez, Tasio and Sánchez-Rada, J. Fernando and Iglesias, Carlos A. and Cummings, Paul
Proceedings	Proceedings of EMAS 2018
CORE Ranking	CORE-B (CORE 2018)
ISBN	
Year	2018
Keywords	Agent-based Social Simulation, Radicalization, terrorism
Pages	
Abstract	This work presents an agent based model of radicalization growth based on social theories. The model aims at improving the understanding of the influence of social links on radicalism spread. The model consists of two main entities, a Network Model and an Agent Model. The Network Model updates the agent relationships based on proximity and homophily, it simulates information diffusion and updates the agents' beliefs. The model has been evaluated and implemented in Python with the agent-based social simulator Soil. In addition, it has been evaluated using a sensitivity analysis.

A Model of Radicalization Growth using Agent-based Social Simulation

Tasio Méndez[1], J. F. Sánchez-Rada[1], Carlos A. Iglesias[1], and Paul Cummings[2]

[1] Intelligent Systems Group, Universidad Politécnica de Madrid, Spain
{tasio.mendez, jf.sanchez, carlosangel.iglesias}@upm.es
http://www.gsi.dit.upm.es
[2] Krasnow Institute, GMU Computational Social, Fairfax, VA
{pcummin2}@gmu.edu

Abstract. This work presents an agent based model of radicalization growth based on social theories. The model aims at improving the understanding of the influence of social links on radicalism spread. The model consists of two main entities, a Network Model and an Agent Model. The Network Model updates the agent relationships based on proximity and homophily, it simulates information diffusion and updates the agents' beliefs. The model has been evaluated and implemented in Python with the agent-based social simulator Soil. In addition, it has been evaluated using a sensitivity analysis.

Keywords: Radicalization · Terrorism · Agent-based social simulation.

1 Introduction

Research works on political terrorism began in the early 1970s. These works were focused on collecting empirical data and analyzing it for public policy purposes. Terrorist activity was usually attributed to personality disorders or "irrational" thinking [1]. However, later research paint a richer picture, and suggest that there are many additional factors that should be considered.

Many scholars, government analysts and politicians point out that since the mid 1990s terrorism has changed. This "new" form of terrorism is is often motivated by religious beliefs and it is more fanatical, deadly, and pervasive. It also differs in terms of goals, methods and organization [1, 2]. However, this the drivers of current terrorism involve not only political or religious interests but also include fanaticism. Consequently, terrorism is the result of a complex process of radicalization. i.e., a progressive adoption of extreme political, social or religious ideals.

Nevertheless, this process does not always lead to violence acts such as terrorism [3]. It is of vital importance to understand the properties of radicalization in order to anticipate said violence. The main challenge with regard to understanding how these organizations work is that information is not always available. And, when it is available, it is often incomplete or inaccurate.

One common approach to face terrorism is trying to understand its roots, motivation and practices. In particular, it is of vital importance nowadays to understand how terrorist organizations recruit new members and isolate them. Moreover, terrorist organizations have effectively used social media and social networks to expand their networks through real-time information exchange.

As society and new forms of communications evolve, terrorists are developing new forms of organization for their purposes. Organizations can thus flatten out their pyramid of authority and control. The resulting structure can take different forms, from a dense network to a group of more or less autonomous, dispersed entities, linked by communications and perhaps nothing more than a common purpose [4]. Thus, terrorist organizations can be modelled as Social Networks (SNs) where vertices represents members of the organization and links represent communication between members.

Regardless of their structure, terrorist organizations are by definition SNs, and can be modelled as such. Hence, a research based on Agent-based Social Simulation (ABSS) could be a good starting point for understanding the information flow within the network.

This paper proposes an agent-based model of a terrorist organization growth which has been implemented in Soil [5], an agent-based social simulator designed for modelling social networks.

This remainder of the paper is structured as follows. Sect. 2 introduces the ABSS Soil, paying special attention to its modelling approach as well as specific features developed for modeling problems with a geographical component, as it happens in the radicalization process. Sect. 3 introduces the agent-based model of radicalization. Sect. 4 describes the implementation of the model using Soil, and provides an overview of the simulation results, including a sensitivity analysis of the simulation results to evaluate the developed model. Finally, some conclusions and insights are presented in Sect. 5.

2 ABSS Soil

Soil [5] is a modern ABSS for modelling and simulation of SNs. It has been applied to a number of scenarios, ranging from rumour propagation to emotion propagation and information diffusion. Each simulation consists of a set of agents, which typically represent humans, and a network that represents social links between agents.

Agents are characterized by their state and the behaviours they can carry out in every simulation step, usually depending on user state. Each behaviour defines the actions carried out and how agent state evolves, depending on external factors or social factors. Those external or social factors are controlled by environment agents, which are not assigned to any network node.

The main reason for using this simulator is that it is one of the few ABSS platforms that support social network analysis [5]. Two other alternatives were considered: Hashkat and Krowdix.

HashKat [6] is a C++ ABSS platform specifically designed for the study and simulation of social networks. It includes facilities for network growth and information diffusion, based on a kinetic Monte Carlo model. It exports information to be processed by machine learning libraries such as NetworkX [7] or R's iGraph [8] and network visualization with Gephi [9]. The simulator is highly performant, but has two major drawbacks. Firstly, simulations are expressed in a descriptive language. Agents are created by specifying a set of highly configurable parameters. As a result, adding behaviours beyond those already included in the platform involves adding new capabilities to the framework. Secondly, and most importantly, modifications to these behaviours are very tied to the architecture of the platform, rather than being isolated for every type of agent. This makes customization costly, especially for someone without a C++ background.

On the other hand, Krowdix [10] is built on Java ABSS. It uses JUNG [11] for network functions and JFreeChart [12] for visualization. The simulation model considers users, their relationships, user groups and interchanged contents. Its main drawback is that it is not open source.

Conversely, Soil is open source and built using Python and benefits from all the Python ecosystem. Regarding the alternatives, Krowdix project is not longer active, while Hashkat provides many facilities for modifying the settings of the provided agent models, but makes hard the integration of new models. In contrast, Soil has being conceived for experimenting and developing easily new simulation models in Python. This has the advantage of Python's increased popularity, its very gradual learning curve, readability, clear syntax and availability of libraries for network processing and machine learning. The network features of Soil are based on NetworkX, which is the defacto standard library for Social Network Analysis (SNA) of small to medium networks. NetworkX provides functionalities for manipulating and representing graph models, generators of classical and popular graph models, including generators for geometric graphs, and graph algorithms for analyzing graph properties. In addition, NetworkX is interoperable with a great number of graph formats, including GEXF, GML, GraphML and JSON among others.

2.1 Architecture

As previously stated, simulations in Soil consist of agents and a network that represents social links between agents. Agents are characterized by their state (e.g. infected) and the behaviours they can carry out in every simulation step, which usually depend on the user state. Each behaviour defines the actions carried out (e.g. tweeting, following a user, etc.) and how the agent state evolves depending on external factors (e.g. news about a topic) or social factors (e.g. opinion of their friends). The likelihood or frequency of each action is typically configurable by either globally or agent-level variables.

This simulation model has been implemented in the architecture shown in Fig. 1 and consists of four main components.

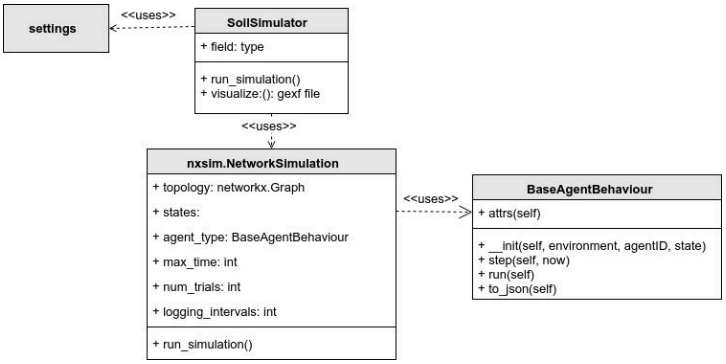


Fig. 1. Simulation components

The *NetworkSimulation* class is in charge of the network simulator engine. It provides forward-time simulation of events in a network based on nxsim [3] and Simpy [13]. Based on configuration parameters, a graph is generated with NetworkX and an agent class is populated to each network node. The main parameters are the network type, number of nodes, maximum simulation time, number of simulations and timeout between each simulation step.

The *BaseAgentBehaviour* class is the basic agent behaviour that should be extended for each social network simulation model. It provides a basic functionality for generation of a JSON file with the status of the agents for its analysis with machine libraries such as Scikit-Learn [14].

The *SoilSimulator* class is in charge of running the simulation pipeline defined in Sect. 2.2, which consists in running the simulation and generating a visualization file in Graph Exchange XML Format (GEXF) which can be visualized with Gephi. In addition, interactive analysis can be done with IPython web interface.

Settings groups the general settings for simulations and the settings of the different models available in Soil's simulation model library.

2.2 Simulation workflow

An overview of the system's flow is shown in Fig. 2. The simulation workflow consists of three steps: configuration, simulation and visualization.

In the first step, the main parameters of the simulation are configured in the JSON or YAML settings file. The main parameters are: network graph type, number of agents, agent types and weights, maximum time of simulation and time step length. In addition, the parameters of the behaviour model should

[3] https://pypi.python.org/pypi/nxsim

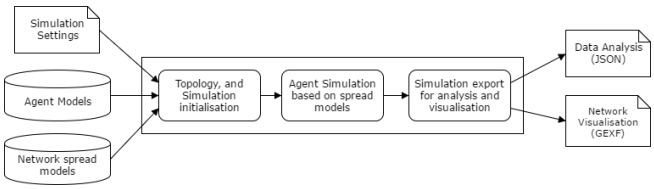


Fig. 2. Social simulator's workflow

be configured (e.g. initial states or probability of an agent action). Agent behaviours should be selected from the provided library or developed extending the *BaseAgentBehaviour* class.

Once the simulation is configured, the next step is the simulation, that can be done step by step or a number of steps. The class *BaseAgentBehaviour* stores the status of every agent in every simulation step into a JSON file to be exported once the simulation is finished. This allows us to automatize the process of generating the .gexf file.

Finally, users can carry out further analysis with the JSON file as well as visualize the evolution the simulation with the generated .gexf file with Gephi.

3 Radical Simulation Model

3.1 Problem

As previously discussed, in the last years, the way people communicate has changed, becoming more relevant social networks, where everyone can exchange messages, images and videos. Terrorist organizations also have moved forward by setting up radio stations, TV channels or Internet websites. These activities allow them to increase their strength, their funds and better recruit new people.

Since terrorist organizations can be modeled as social networks we can study how information is shared and how people become members of groups or even new relationships. Within the proposed model (Sec. 3.2), terrorist groups will be represented as graphs where vertices represent members and edges represent communication between those members.

However, radicalism is not only sustained by flow information. Multiple causes, rather than a single cause should be considered, including social and spacial relations which evolve over time. Estimating their evolution is important for management, command and control structures, as well as for intelligence analysis research purposes. By knowing future social and spacial distributions, analysts can identify emergent leaders, hot spots, and organizational vulnerabilities [15].

In order to approach to the radicalism spread, a spatial distribution is used based on Geometric Graph Generators [16], which provides geographical positions to agents, being able to manage real environments.

The physical space aims to produce more insightful results when considering the spread of terrorism [17]. Properties of space and place are vital components of terrorist training, planning, and activities.

Besides, based on the principle of homophily, as a contact between similar people occurs at a higher rate than among dissimilar people, it is more likely to have contact with those who are closer to us in geographic location than those who are distant [18]. It is theorized that, in general, close proximity in geographic space strongly influences closeness in social space [17].

As it was discussed above, the proposed model will try to approach to the fact of the rise of radicalism within a specified geographic area considering real geographical connections between members.

3.2 Model development

Three levels of analysis are widely accepted for the radicalization process [19]: *micro-level* (i.e. the individual level involving feelings of grievance, marginalization, etc.), *meso-level* (i.e. the social environment surrounding radicals and the population and lead to the formation of radical groups), and *macro-level* (i.e. impact of government policies, religion, media, including radicalization of the public opinion and political parties).

The model here proposed is focused on analyzing the macro-level, including limited aspects of the micro-level (such as the vulnerability level).

Several aspects have been considered for modeling the radicalism growth at the meso-level. First, the model considers the impact of *havens* [20] and *training areas* [21]. Havens, also known as sanctuaries, provide radical groups the possibility to obtain long term funding and serve the purposed of solidifying group cohesion. Terrorist training camps aim at providing indoctrination and teaching for terrorism and are distributed around the world. They foster group identity formation and group cohesion, and require geographical isolation and easy access to weapons.

The modelling of the radicalism spread involves population and places as it was discussed above. People can play two roles: (1) population as the people that can be radicalized and (2) terrorist that spread their message to locals and try to recruit civilians to join the terrorist network.

Based on a previous model proposed by Cummings [17], terrorists have little opportunities for effective training, planning, and other logistic necessities. Those

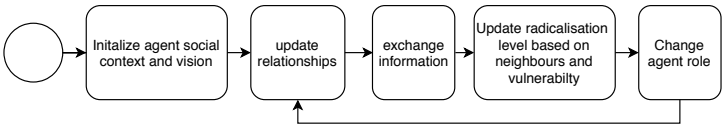


Fig. 3. General workflow of the simulation

areas are modelled by (1) training environments, which increase the influence to the nodes that are attached to them, and (2) havens where people is save. The nodes that are joined to havens get less influenced if the havens is not radicalized, but it could get radicalized and its behaviour will change.

For implementing the environment described, we will use four different models that interact with each other.

- Spread model in charge of the information flow which determine the state of population. Each node contains a threshold where once reached, the node is marked as informed and it will pass from a civilian state to a radical state.
- Network model in charge of controlling spatial and social relations between population.
- Havens model which will modify nodes vulnerability depending on haven state as it is going to be explained below.
- Training areas model which will decrease neighbouring nodes vulnerability.

The network consists on N nodes that have two coordinates, as since Geometric Graph Generators [16] are used, that position each node on a map. The edge between two nodes, indicates direct bidirectional communication between both of them.

All agents are assumed to have similar parameters but are heterogeneous in their representation. Within the spread model, each node develops its own belief about whether the information is valid by calculating weighted mean belief B_i from it neighbors, and combining that with its initial belief B_0, which is normalized between 0 and 1 [22]. In addition, in every step two agents will exchange information given a probability of interaction.

The mean belief is calculated given its own vulnerability and the neighbours influence as well as the information spread intensity (α) which is also normalized and consider how much information is exchanged in every step of the simulation.

$$B_e = \sum_{i=0}^{n} \frac{B_i \; D_i}{\sum_{j=0}^{n} D_j} \tag{1}$$

The node influence D_i parameter has been included in Eq. 1 – where n is the number of neighbours of the node – as the change in behavior that one person causes in another as a result of an interaction [23] measured as degree centrality that is defined as the number of adjacencies upon a node, which is the sum of each row in the adjacency matrix representing the network. It can be interpreted within social networks as a measure of immediate influence – the ability to infect others directly or in one time period [24]. This SNA function returns values that are normalized by dividing by the maximum possible degree in a simple graph $N - 1$ where N is the number of nodes in G.

$$B_n = B_e \; \alpha + B_0 \; (1 - \alpha) \quad ; \quad 0 \leq \alpha \leq 1 \tag{2}$$

As it was explained above, in Eq. 2 the parameter to indicate the information spread intensity is included. When its value is 0%, no information is exchanged and when it increases, the knowledge diffusion grows.

$$B_i = B_n \; N_v + B_0 \; (1 - N_v) \quad ; \quad 0 \leq N_v \leq 1 \tag{3}$$

The node vulnerability (N_v) parameter is included in Eq. 3 as the extent to which individuals conform or adopt variable attributes such as opinions from their attached nodes. In other words, if $N_v = 1$, the node will be fully influenced by their connected nodes, where a value of $N_v = 0$, would mean it would not be influenced by connected nodes, so no change in the network is expected. Thus, individuals who are merely sympathetic may be influenced to more extreme opinions by their friends after they join the terrorist network.

Once the mean belief developed by the agent reach the threshold, it is marked as informed and it will change its state from civilian to radical. Every agent in radical state will be only influenced by radical agents since the radical experience no restraining influence from non-radicals [25]. Furthermore, once an agent is in the radical state, the information spread intensity will began to value 100%, as once you are radical the most information you get from another radical agents.

With the purpose of determining the most important nodes within the terrorist network, they are marked as *leaders* based on the SNA function: betweenness centrality [22], that is defined of a node v as the sum of the fraction of all-pairs shortest paths that pass through v.

As node vulnerability (N_v) was explained above, training areas and havens will modify this attribute depending on their status. Training areas will decrease the parameter from its neighbours, where a value of 1 for training area influence will make all its neighbours fully vulnerable. However, a value of 1 for haven influence will make invulnerable all its neighbours when the state of the haven is not radical. Nevertheless, once the haven is marked as radical, its behaviour will be similar to training areas.

Finally, the network model in charge of controlling spacial and social relations takes into account that agents have the opportunity to interact with other agents. They select an agent to interact with according to a probability of interaction – different from the one mentioned above – based on two parameters: (1) social distance and (2) spatial proximity.

On one side, social distance (SD) take into account the fact that if two agents must cross many social links, then the probability should be low and vice versa. It compute it by finding the shortest path between to agents and then dividing one by the number of links in that path.

$$SD_{i,j} = \frac{1}{|A \; A_{i,j}|} \tag{4}$$

where $|A \; A_{i,j}|$ is the shortest path from i to j. When computing the social distance, each agent can only reach all those nodes that are withing its sphere of influence parameter. An agent can recognize and distinguish the closeness of other agents withing the sphere of influence, but it can't differentiate the closeness when the interacting agent is outside the perimeter.

On the other side, spatial proximity (SP) takes into account that two agents at the same location are more likely to talk than being in different locations.

Some might argue that SP is not significant in the Internet age. However, in the terrorism domain, attending the same training area or the same location is a critical interaction indicator [15].

As Geometric Graph Generators returns coordinates normalized between 0 and 1, the probability of being at the same location will be computed as the inverse of the distance between two agents.

$$SP_{i,j} = (1 - |d_{i,j}|) \tag{5}$$

where $|d_{i,j}|$ is the distance between the nodes. Like in SD the probability is bounded by a sphere of influence parameter, in SP the probability will be bounded by a vision range parameter. All agents outside this perimeter will be unreachable by the current agent.

Table 1. Simulation input parameters.

Model	**Name**	**Implication**
Terrorist Spread	information_spread_intensity	The amount of information exchanged in every step of the simulation.
	terrorist_additional_influence	Additional influence added to agents whom status is radical.
	min_vulnerability	The minimum vulnerability that an agent could have (*default 0*).
	max_vulnerability	The maximum vulnerability that an agent could have. The allocation of this parameter follows a continuous uniform distribution. The maximum value that this parameter can take is the unit.
	prob_interaction	The probability that two agents exchange information in one step.
Training Area	training_influence	The influence that a training area applies to its neighbours.
Haven	haven_influence	The influence that a haven applies to its neighbours.
Terrorist Network	sphere_influence	The maximum number of social links that an agent can cross for a new interaction.
	vision_range	The range on the spatial-route network specifying the maximum distance an agent can move for a new interaction.
	weight_social_distance	The weight of social distance (SD) to calculate the interaction probability.
	weight_link_distance	The weight of spatial proximity (SP) to calculate the interaction probability.

Once defined both parameters, we can compute the probability of interaction that it will be calculated as following.

$$P_{i,j}^{Interaction} = \omega_1 \ SD_{i,j} + \omega_2 \ SP_{i,j} \tag{6}$$

where ω_1 and ω_2 are the weights of SD and SP respectively with the purpose of customizing the environment.

None of the parameters will limit the probability of interaction. Thus, the candidate agents will be the sum of all the agents that are inside the perimeter of the sphere of influence or the vision range.

Thereby, an agent can establish a new way of communication with its candidate agents, so the probability of interaction is calculated between each agent and its candidate agents.

As it was explained, the aim of the model is trying to approach to the fact of the radicalism spread withing a specified geographic area. For that reason, in Table 1 all parameters of the simulation are detailed for representing a scenario as real as possible. Aside from all the parameters explained, the network can be modelled using one of the random network generation methods from NetworkX. It is also possible to control the ratio of each type of agent.

4 Experimental results

The model has been implemented using the Soil Simulator as it was discussed above. The scenario represents a specified geographic area that can be customized with the purpose of approaching a real scenario.

Every agent exchange information several times during the simulation and every portion of time is known as *step*. One one hand, in every step an agent belonging to the Network Model will update its relationships based on the input parameters. After this action, the control is passed to the Spread Model that will be in charge of how the information will flow in that step. The current agent will be influenced by its neighbours depending on their internal parameters values.

On the other hand, if the current agent is a haven or a training area, the step will consist on modifying the internal parameters of their neighbours as it was explained in the previous section.

With the purpose of making the simulations more interactive, a web application has been developed using D3.js [26] for visualizing the results. As we can notice in Fig. 4 the simulation returns a graph that is presented in the main area of the web application. The graph can be positioned in a map, and it could be represented depending on the step, being able to see it evolve over time. Furthermore, the interface allows users filtering the results or changing the simulation parameters.

The application not only allows the user to visualize the results, it also provides statistics and the option of running more simulations changing the input parameters as it is displayed in Fig. 5. The web application also allows users to

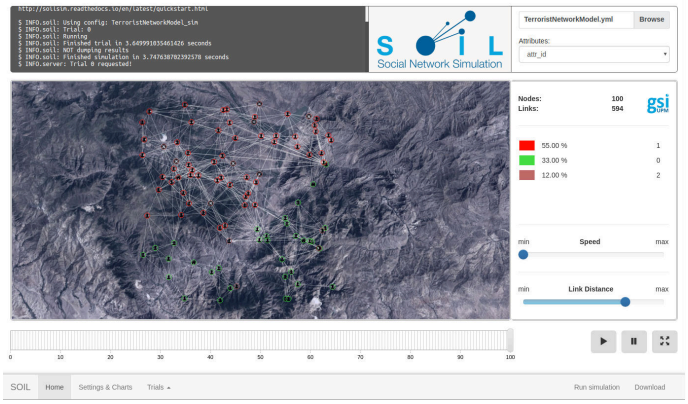


Fig. 4. Visualization of the simulation

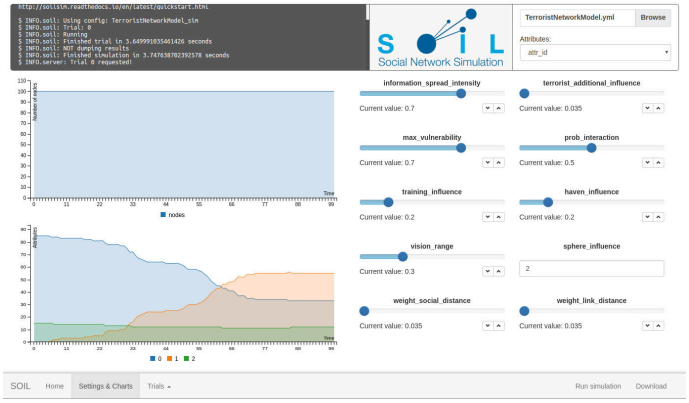
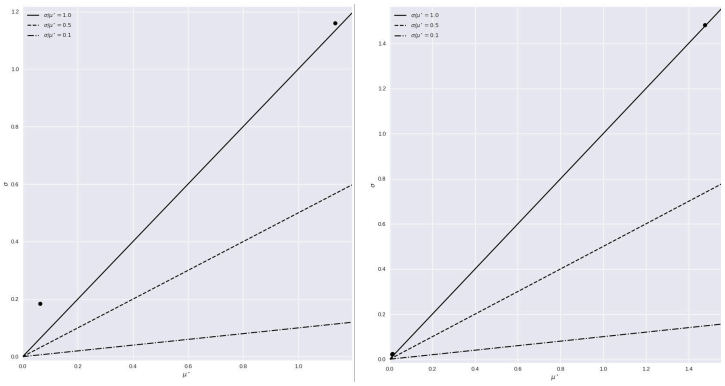


Fig. 5. Visualization of the simulation

export the results of the simulation in different formats such as GEXF [27] or JSONGraph[4] to be analyzed with any other tool.

The model has been evaluated using two different sensitivity analysis methods. The first one is a local approach known as One-at-Time (OAT) approach, that studies small input perturbations on the model output. To bring about

[4] http://netflix.github.io/falcor/documentation/jsongraph.html



(a) Scale Free output (b) Small World output

Fig. 6. Morris method results representation for radical population output for 200 trajectories

this method, 1.000 simulations have been launched with different input values and have been analyzed using the Seaborn [28] library available for Python for exploring and understanding the results.

The other method applied is the Morris method [29] that is referred to as "global sensitivity analysis" that in contrast to local sensitivity analysis, it considers the whole variation range of the inputs [30]. This method is computed using the SALib [31] library for Python.

The primary model outputs of interest for the sensitivity analysis are the radical population understood as the number of agents that have become radical from those who were not radical at the beginning and the mean radicalism within the network.

Both outputs will be measured taking into account different types of simulations. On one side, the network model will be studied assuming that the spread model inherit the another. On the other side, three different topologies (small world, scale free and random clustered) will be analyzed.

In Table 2 the Morris indices are detailed for the network model and mean radicalism output order by μ^*. A total of 200 trajectories were built for the model which results in 1.800 samples. Fig. 7 plots results on the graph (μ^*, σ) and identifies the probability of interaction, the maximum vulnerability and the information spread intensity as the strongest influence on the mean radicalism within the network.

The analysis have been made using a random clustered topology that is created based on proximity between nodes for 100 nodes, and with same number of radical agents at the beginning.

Table 2. Morris indices for network model and mean radicalism output.

Parameter	μ	μ^*	σ
prob_interaction	0.320 631	0.367 384	0.517 95
max_vulnerability	0.243 827	0.349 831	0.413 981
information_spread_intensity	0.252 602	0.324 202	0.379 572
terrorist_additional_influence	0.036 039	0.128 335	0.206 991
weight_social_distance	−0.004 388	0.110 129	0.186 007
vision_range	0.019 502	0.109 09	0.180 97
sphere_influence	0.006 756	0.107 522	0.173 183
weight_link_distance	0.007 996	0.101 815	0.179 93

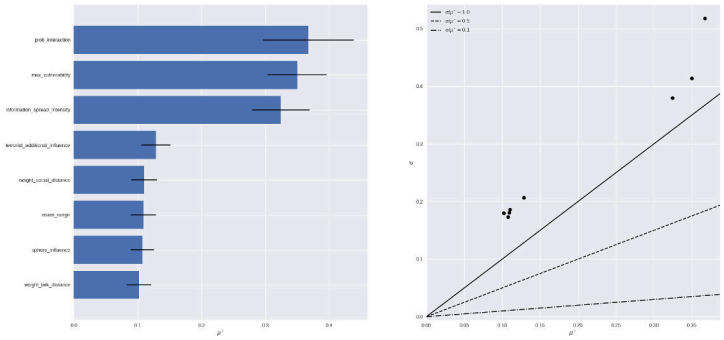


Fig. 7. Morris method results representation for network model and mean radicalism output for 200 trajectories

However, taking into account the population radicalized in a simulation as we can notice in Table 3 and Fig. 8 are similar, but the maximum vulnerability and the information spread intensity is in this case more influential than the probability of interaction.

Morris indices for the three different topologies have similarities as the weight of the radical agents for the distribution through the network is the most influential parameter for both outputs as it can be noticed in Fig. 6 for Scale Free and Small World topologies. In addition, the model output linearly depends on the weight of the agents. Nevertheless, the size of the network have no influence on the two model outputs.

The methods presented attempt to validate certain factors such as types of network connections and the presence of certain kinds of meeting sites which facilitate radicalization while other plausible factors such as community size have little effect. Network types can play an important part in understanding how radicalism spreads, and can be equally important when trying to destabilize or destroy a network.

Table 3. Morris indices for network model and radicalized population output.

Parameter	μ	μ^*	σ
max_vulnerability	0.466 355	0.484 857	0.596 371
information_spread_intensity	0.392 325	0.402 566	0.541 922
prob_interaction	0.268 707	0.331 403	0.568 499
terrorist_additional_influence	0.092 038	0.186 473	0.415 794
weight_link_distance	−0.012 333	0.181 102	0.401 011
vision_range	−0.001 680	0.176 981	0.380 602
sphere_influence	0.005 437	0.169 812	0.358 775
weight_social_distance	0.003 899	0.165 475	0.375 792

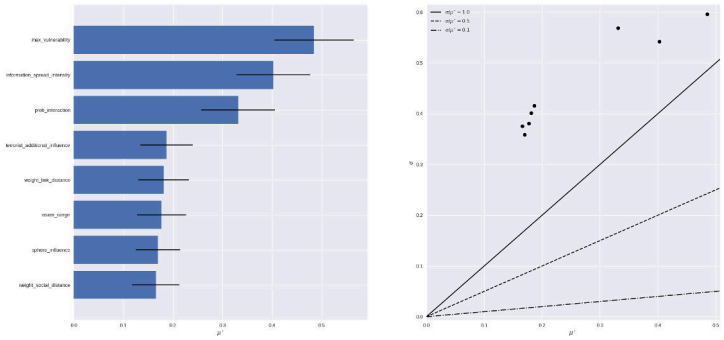


Fig. 8. Morris method results representation for network model and radicalized population output for 200 trajectories

5 Conclusions and Future work

Understanding radicalization roots is a first step for being able to define and apply suitable counter-terrorism measures. There are many challenges for analyzing terrorism networks, given the lack of public datasets and the sensibility of this information. Nonetheless, the application of agent based social simulation is an effective technique for modeling non linear adaptive systems, and they enable analyzing and validating social theories of the radicalization process.

In this work we present a model and a tool for agent-based modeling of radical terrorist networks. We have propose building the agent-based model around two main concepts, the Network Model and the Agent Model. While the first is in charge of managing agent relationships, the second defines the specific behaviour of every agent. This approach has been applied for modeling terrorist growth. The proposed model is focused on analyzing the impact of the information exchange and environmental radicalization in the radicalization process.The

evaluation and analysis of the simulation results provides insight regarding the importance of the simulation parameters, including the network characteristics.

Future work should include a broader and deeper perspective of absolute and relative deprivation and how each can influence the spread of radicalism.

Acknowledgements

This work is supported by the Spanish Ministry of Economy and Competitiveness under the R&D projects SEMOLA (TEC2015-68284-R), by the Regional Government of Madrid through the project MOSI-AGIL-CM (grant P2013/ICE-3019, co-funded by EU Structural Funds FSE and FEDER); by the European Union through the project Trivalent (Grant Agreement no: 740934) and by the Ministry of Education, Culture and Sport through the mobility research stay grant PRX17/00417.

References

1. Martha Crenshaw. The psychology of terrorism: An agenda for the 21st century. *Political psychology*, 21(2):405–420, 2000.
2. Alexander Spencer. Questioning the concept of 'new terrorism'. *Peace, Conflict and Development*, pages 1–33, 2006.
3. Oliver Gruebner, Martin Sykora, Sarah R Lowe, Ketan Shankardass, Ludovic Trinquart, Tom Jackson, SV Subramanian, and Sandro Galea. Mental health surveillance after the terrorist attacks in Paris. *The Lancet*, 387(10034):2195–2196, 2016.
4. David Tucker. What is new about the new terrorism and how dangerous is it? *Terrorism and Political Violence*, 13(3):1–14, 2001.
5. Jesús M. Sánchez, Carlos A. Iglesias, and J. Fernando Sánchez-Rada. Soil: An Agent-Based Social Simulator in Python for Modelling and Simulation of Social Networks. In Bajo J. Vale Z. Demazeau Y., Davidsson P., editor, *Advances in Practical Applications of Cyber-Physical Multi-Agent Systems: The PAAMS Collection*, volume 10349 of *LNAI*, pages 234–245. PAAMS 2017, Springer Verlag, June 2017.
6. Kevin Ryczko, Adam Domurad, Nicholas Buhagiar, and Isaac Tamblyn. Hashkat: large-scale simulations of online social networks. *Social Network Analysis and Mining*, 7(1):4, 2017.
7. Aric Hagberg, Pieter Swart, and Daniel S Chult. Exploring network structure, dynamics, and function using NetworkX. Technical report, Los Alamos National Lab.(LANL), Los Alamos, NM (United States), 2008.
8. Gabor Csardi and Tamas Nepusz. The igraph software package for complex network research. *InterJournal, Complex Systems*, 1695(5):1–9, 2006.
9. Mathieu Bastian, Sebastien Heymann, Mathieu Jacomy, et al. Gephi: an open source software for exploring and manipulating networks. *Icwsm*, 8:361–362, 2009.
10. Diego Blanco-Moreno, Rubén Fuentes-Fernández, and Juan Pavón. Simulation of online social networks with Krowdix. In *Computational Aspects of Social Networks (CASoN), 2011 International Conference on*, pages 13–18. IEEE, 2011.
11. Joshua O'Madadhain, Danyel Fisher, Padhraic Smyth, Scott White, and Yan-Biao Boey. Analysis and visualization of network data using JUNG. *Journal of Statistical Software*, 10(2):1–35, 2005.

12. David Gilbert. The jFreeChart class library. *Developer Guide. Object Refinery*, 7, 2002.
13. Norm Matloff. Introduction to discrete-event simulation and the simpy language. *Davis, CA. Dept of Computer Science. University of California at Davis. Retrieved on August*, 2:2009, 2008.
14. Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of machine learning research*, 12(Oct):2825–2830, 2011.
15. Il-Chul Moon and Kathleen M Carley. Modeling and simulating terrorist networks in social and geospatial dimensions. *IEEE Intelligent Systems*, 22(5), 2007.
16. Mathew Penrose. *Random geometric graphs*. Oxford university press, 2003.
17. Paul Cummings and Chalinda Weerasinghe. Modeling the characteristics of radical ideological growth using an agent based model methodology. In *MODSIM World*, 2017.
18. Miller McPherson, Lynn Smith-Lovin, and James M Cook. Birds of a feather: Homophily in social networks. *Annual review of sociology*, 27(1):415–444, 2001.
19. Rositsa Dzhekova, N Stoynova, A Kojouharov, M Mancheva, D Anagnostou, and E Tsenkov. Understanding Radicalisation. Review of Literature. *Center for the Study of Democracy, Sofia*, 2016.
20. Ari Jean-Baptiste. *Terrorist Safe Havens: Towards an Understanding of What They Accomplish for Terrorist Organizations*. PhD thesis, University of Kansas, 2010.
21. James JF Forest. Terrorist Training Centers Around the World: A Brief Review. *The Making of a Terrorist: Recruitment, Training and Root Causes*, 2, 2005.
22. Paul Cummings. Modeling the characteristics of radical ideological growth using an agent basedmodel methodology. Master Thesis, George Mason University, 2017.
23. Lisa Rashotte. Social influence. *The Blackwell encyclopedia of sociology*, 2007.
24. Stephen P Borgatti. Centrality and network flow. *Social networks*, 27(1):55–71, 2005.
25. Michael Genkin and Alexander Gutfraind. How do terrorist cells self-assemble: Insights from an agent-based model of radicalization. Technical report, SSRN, July 2011.
26. Nick Qi Zhu. *Data visualization with D3.js cookbook*. Packt Publishing Ltd, 2013.
27. GEXF Working Group and others. GEXF file format, 2015.
28. Kevin Sheppard. Introduction to Python for econometrics, statistics and data analysis. *Self-published, University of Oxford, version*, 2, 2012.
29. Max D Morris. Factorial sampling plans for preliminary computational experiments. *Technometrics*, 33(2):161–174, 1991.
30. Bertrand Iooss and Paul Lemaître. A review on global sensitivity analysis methods. In *Uncertainty management in simulation-optimization of complex systems*, pages 101–122. Springer, 2015.
31. Jon Herman and Will Usher. SALib: an open-source Python library for sensitivity analysis. *The Journal of Open Source Software*, 2(9), 2017.

3.3.4 Modeling Social Influence in Social Networks with SOIL, a Python Agent-Based Social Simulator

Title	Modeling Social Influence in Social Networks with SOIL, a Python Agent-Based Social Simulator
Authors	Merino, Eduardo and Sánchez, Jesús M. and García Martín, David and Sánchez-Rada, J. Fernando and Iglesias, Carlos A.
Proceedings	Advances in Practical Applications of Cyber-Physical Multi-Agent Systems: The PAAMS Collection
ISBN	978-3-319-59929-8
Volume	10349
Year	2017
Keywords	agent based social simulation, networkx, social network, soil
Pages	337–341
Online	https://link.springer.com/chapter/10.1007/978-3-319-59930-4_33
Abstract	The application of Agent-based Social Simulation (ABSS) for modeling social networks requires specific facilities for modeling, simulation and visualization of network structures. Moreover, ABSS can benefit from interactive shell facilities that can assist the model development process. We have addressed these problems through the development of a tool called SOIL, which provides a Python ABSS specifically designed for social networks. In this paper we present how this tool is applied to simulate viral marketing processes in a social network, and to evaluate the model with real data.

Modeling Social Influence in Social Networks with SOIL, a Python Agent-Based Social Simulator

Eduardo Merino, Jesús M. Sánchez, David García,
J. Fernando Sánchez-Rada, and Carlos A. Iglesias(✉)

Intelligent Systems Group, DIT, E.T.S. de Ingenieros de Telecomunicación,
Universidad Politécnica de Madrid, 28040 Madrid, Spain
{eduardo.merinom,jesusmanuel.sanchez.martinez,
david.garcia.martin}@alumnos.upm.es, {jfernando,cif}@dit.upm.es
http://www.gsi.dit.upm.es

Abstract. The application of Agent-based Social Simulation (ABSS) for modeling social networks requires specific facilities for modeling, simulation and visualization of network structures. Moreover, ABSS can benefit from interactive shell facilities that can assist the model development process. We have addressed these problems through the development of a tool called SOIL, which provides a Python ABSS specifically designed for social networks. In this paper we present how this tool is applied to simulate viral marketing processes in a social network, and to evaluate the model with real data.

Keywords: Social network · SOIL · Python · Viral marketing · Brand reputation · Rumor propagation

1 Introduction

Social networks have become relevant in our professional and personal relationships. Thus, social network analysis and simulation can be effective for understanding and exploiting homophily and social influence processes in social networks. Marketing techniques are usually applied to exploit social influence in social networks, in applications such as viral or word-of-mouth marketing, rumor spreading and online reputation management. This paper complements the demo presented at PAAMS 2017 on the use of the Python-based ABSS SOIL tool for social network modeling and analysis, which is illustrated with a number of developed models.

2 Main Purpose

SOIL aims at providing a research environment for ABSS in Python, with a strong focus on interoperability with existing libraries. It integrates with the

© Springer International Publishing AG 2017
Y. Demazeau et al. (Eds.): PAAMS 2017, LNAI 10349, pp. 337–341, 2017.
DOI: 10.1007/978-3-319-59930-4_33

popular network processing library NetworkX[1] and with network visualization tools such as Gephi[2].

3 Demonstration

In this paper we present a case study that models the social influence of users in the social network Twitter. In particular, we study the role of social influence in rumor propagation and brand monitoring. In both applications, a diffusion message (rumor or brand advertisement) is propagated in the social network with the aim of infecting users. Users are considered infected when they accept or embrace the content of the message. The model presented is $M_{2.2}$ [4]. Twitter users are modeled as agents which can be in three states: *neutral*, if they are not affected by the message; *infected*, if they accept the message; *vaccinated*, if they have not been infected yet and believe in the antirumor or are infected by a message from a different brand; and *cured*, if they have been infected, but now believe the antirumor or are infected by a different brand. Additionally, the model includes a specific kind of users, called beacons, which detect the propagation of the message and try to combat it. Beacons are modeled after authorities that prevent rumor diffusion and competing influencers in social media. Agents include two additional states, beacon-off and beacon-on to represent beacons before and after detecting a rumor in a close node (neighbor).

The spread model starts with an initial number of infected users. In every simulation step, the state of each user may change though a series of interactions, each of which happens with a different probability. Infected users try to infect their neutral neighbors. Neutral agents may also become vaccinated with a given probability based on external factors (i.e. news). Vaccinated users attempt to cure or vaccinate their neighbors. Lastly, beacon agents spread anti-rumors to their neighbors, and follow these neighbors' contacts.

Table 1. Datasets of Twitter rumors and brand monitoring

Dataset	Number of tweets	Purpose	Period	Reference
Ford	348	Brand monitoring	13 months	[1]
Toyota	582	Brand monitoring	14 months	[1]
Obama	4975	Rumor propagation	8 days	[3]
Palin	4423	Rumor propagation	10 days	[3]

We have validated this diffusion model on four datasets (Table 1). The first two datasets (Ford and Toyota) are subsets of the Replab dataset [1], which focuses on monitoring the reputation of companies and individuals in Twitter.

[1] https://networkx.github.io/.
[2] https://gephi.org/.

Each tweet is classified as related (or unrelated) to an entity, the polarity for the entity's reputation (positive, negative or neutral), and the priority of the topic cluster the tweet belongs to (alert, midly important, unimportant). We have filtered the dataset and selected two automotive brands, Ford and Toyota, which can simulate how two brands advertise themselves on social media. In this case, the advertisement message is propagated and succeeds if the brand gets a good reputation. The last two datasets (Obama and Palin) are rumor datasets [3] that deal with identifying the spread of misinformation in social networks, such as Obama being a muslin or Palin's divorce. The dataset is labeled as endorses (propagate the rumor), denies (deny the rumor), questions (doubt about rumor credibility) or unrelated (not related to the rumor).

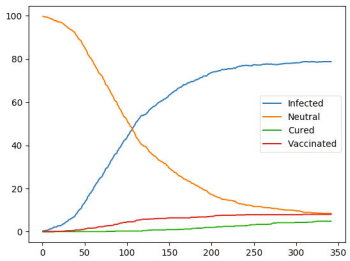


Fig. 1. Agent evolution

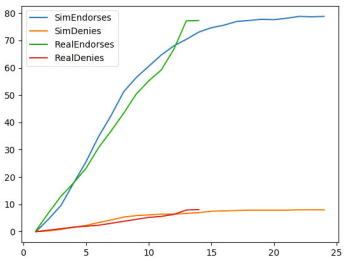


Fig. 2. Realism evaluation

The demonstration may be run in an IPython interactive shell, where simulation parameters can be defined. After running the simulation, the results are stored as Python objects, which can be inspected and visualized. For example, Fig. 1 shows the temporal evolution of agent states. The x axis represents the days and the y axis the number of simulated agents. In addition, the platform includes facilities for evaluating the realism of the simulation. For this purpose, we compare the daily number of endorsers and deniers in the dataset and the simulation. Figure 2 shows a comparison for the dataset of Toyota as a monthly evolution of the ratio of users that accept the diffusion message (endorsers) or reject it (deniers).

In addition, the platform generates a Graph Exchange XML Format (GEXF) file that can be used for analyzing the simulation with network analysis tools such as Gehpi. In particular, the visualization can be animated to show the temporal evolution of the spread model. Figure 3 shows a screenshot of the animation, where the colors denote infected (red), vaccinated (blue), cured (green) and beacon-off (yellow). Another interesting experiment is validating the realism of the simulation. An alternate view of the network is shown in Fig. 4.

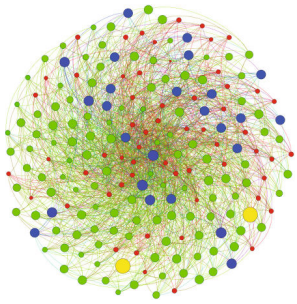


Fig. 3. Network visualization in Gephi (Color figure online)

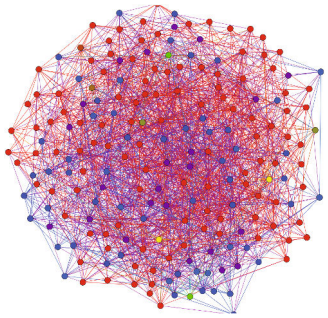


Fig. 4. Alternate network visualization

4 Conclusions

This demonstration shows the application of a Python ABSS specifically designed for social network modeling and its application to information diffusion in social networks. The models in this paper had an existing implementation written in Java [4], combining MASON [2] and the graph library GraphStream[3]. Porting them to SOIL was straightforward and resulted in much simpler comprehensible code. The main benefits from using SOIL derive from using a simple yet extensible interface and the Python programming language. As a result, it is very easy to extend agent behavior while leveraging the existing ecosystem to integrate machine learning algorithms or semantic interfaces, to name a few. Moreover, the use of an interactive shell such as IPython[4].

Acknowledgements. This work is supported by the Spanish Ministry of Economy and Competitiveness under the R&D projects SEMOLA (TEC2015-68284-R) and EmoSpaces (RTC-2016–5053-7), by the Regional Government of Madrid through the project MOSI-AGIL-CM (grant P2013/ICE-3019, co-funded by EU Structural Funds FSE and FEDER), and by the European Union through the project MixedEmotions (Grant Agreement no: 141111). The authors want to thank Vahed Qazvinian for making available the rumor datasets for our research.

References

1. Amigó, E., Carrillo de Albornoz, J., Chugur, I., Corujo, A., Gonzalo, J., Martín, T., Meij, E., Rijke, M., Spina, D.: Overview of RepLab 2013: Evaluating online reputation monitoring systems. In: Forner, P., Müller, H., Paredes, R., Rosso, P., Stein, B. (eds.) CLEF 2013. LNCS, vol. 8138, pp. 333–352. Springer, Heidelberg (2013). doi:10.1007/978-3-642-40802-1_31
2. Luke, S.: MASON: A multiagent simulation environment. Simulation **81**, 517–527 (2005)

[3] http://graphstream-project.org/.
[4] https://ipython.org/.

3. Qazvinian, V., Rosengren, E., Radev, D.R., Mei, Q.: Rumor has it: Identifying misinformation in microblogs. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 1589–1599. Association for Computational Linguistics (2011)
4. Serrano, E., Iglesias, C.A.: Validating viral marketing strategies in twitter via agent-based social simulation. Expert Syst. Appl. **50**(1), 140–150 (2016)

3.3.5 Soil: An Agent-Based Social Simulator in Python for Modelling and Simulation of Social Networks

Title	Soil: An Agent-Based Social Simulator in Python for Modelling and Simulation of Social Networks
Authors	Sánchez, Jesús M. and Iglesias, Carlos A. and Sánchez-Rada, J. Fernando
Proceedings	Advances in Practical Applications of Cyber-Physical Multi-Agent Systems: The PAAMS Collection
ISBN	978-3-319-59929-8
Volume	10349
Year	2017
Keywords	agent based social simulation, python, social networks, soil
Pages	234–245
Online	https://link.springer.com/chapter/10.1007/978-3-319-59930-4_19
Abstract	Social networks have a great impact in our lives. While they started to improve and aid communication, nowadays they are used both in professional and personal spheres, and their popularity has made them attractive for developing a number of business models. Agent-based Social Simulation (ABSS) is one of the techniques that has been used for analysing and simulating social networks with the aim of understanding and even forecasting their dynamics. Nevertheless, most available ABSS platforms do not provide specific facilities for modelling, simulating and visualising social networks. This article aims at bridging this gap by introducing an ABSS platform specifically designed for modelling social networks. The main contributions of this paper are: (1) a review and characterisation of existing ABSS platforms; (2) the design of an ABSS platform for social network modelling and simulation; and (3) the development of a number of behaviour models for evaluating the platform for information, rumours and emotion propagation. Finally, the article is complemented by a free and open source simulator.

Soil: An Agent-Based Social Simulator in Python for Modelling and Simulation of Social Networks

Jesús M. Sánchez, Carlos A. Iglesias( ), and J. Fernando Sánchez-Rada

Intelligent Systems Group, DIT, E.T.S. de Ingenieros de Telecomunicación,
Universidad Politécnica de Madrid, 28040 Madrid, Spain
jesusmanuel.sanchez.martinez@alumnos.upm.es,
{cif,jfernando}@dit.upm.es
http://www.gsi.dit.upm.es

Abstract. Social networks have a great impact in our lives. While they started to improve and aid communication, nowadays they are used both in professional and personal spheres, and their popularity has made them attractive for developing a number of business models. Agent-based Social Simulation (ABSS) is one of the techniques that has been used for analysing and simulating social networks with the aim of understanding and even forecasting their dynamics. Nevertheless, most available ABSS platforms do not provide specific facilities for modelling, simulating and visualising social networks. This article aims at bridging this gap by introducing an ABSS platform specifically designed for modelling social networks. The main contributions of this paper are: (1) a review and characterisation of existing ABSS platforms; (2) the design of an ABSS platform for social network modelling and simulation; and (3) the development of a number of behaviour models for evaluating the platform for information, rumours and emotion propagation. Finally, the article is complemented by a free and open source simulator.

1 Introduction

Social Networks (SNs) have a great impact in our lives. While they started to improve and aid communication, nowadays they are used both in professional and personal spheres, affecting different aspects ranging economic [11] to health outcomes [22].

The emergence of social computing [45] has raised the interest in the design, analysis and forecasting of social systems. To this end, Social Computing is a cross-disciplinary field with theoretical underpinnings including both computational and social sciences, as well as research from areas such as social psychology, human computer interaction, Social Network Analysis (SNA), anthropology, sociology, organization theory, and computing theory.

One of the fields where ABSS has been applied is the analysis and simulation of social networks, in applications such as viral marketing [40], innovation diffusion [20], rumour propagation [23]. In fact, some authors [33] propose that the use of social media in agent based simulations can leverage the input data

© Springer International Publishing AG 2017
Y. Demazeau et al. (Eds.): PAAMS 2017, LNAI 10349, pp. 234–245, 2017.
DOI: 10.1007/978-3-319-59930-4_19

problem in ABSS, since capturing data from individuals is an expensive and difficult task in longitudinal studies.

Nevertheless, there is a lack of ABSS platforms that provide support for social network modelling. Thus, we aim at bridging this gap by designing and developing an ABSS in Python specifically designed for social networks which benefits from the wide number of available Python libraries for network analysis and machine learning.

The remainder of the article is organised as follows. First, we review existing ABSS platforms to justify why they are not suitable for our problem in Sect. 2, as well as applications of ABSS to social network analysis. Based on this, we present a set of requirements for the desired platform in Sect. 3. Then, the proposed model, architecture and simulation workflow are presented in Sect. 4. The platform has been evaluated through the development of a library of models which is described in Sect. 5. We conclude with Sect. 6 and provide an outlook of future work.

2 Review of ABSS Platforms for Modelling SNs

In recent years numerous ABSS have been developed, as shown by Railsback et al. [34] and Nikolay et al. [31]. Based on this latter work that reviews 55 ABSS platforms, we have reviewed ABSS platforms to evaluate their suitability for modelling social networks, attending to the following aspects: (i) type of platform (general purpose or domain specific), (ii) programming language, (iii) expertise in its application to SNs, (iv) whether the framework provides SNA facilities and (v) whether the license is Open Source (OS). Table 1 summarizes the platforms and the reviewed aspects.

From the initial list provided in [31] we have filtered out platforms that are under a commercial license (e.g. cougaar), not actively developed (e.g. ABLE), focused on training (e.g. AgentSheets), or otherwise not directly focused on simulation (e.g. ECJ or Jade). The resulting set of platforms is Common-Pool Resources and Multi-Agent Systems (Cormas) [7], Madkit [14], Mason [24], NetLogo [35], Repast [32], SeSam [16] and Swarm [27]. Based on our literature research, we have added some additional platforms: UbikSim [9], EscapeSim [41], HashKat [38], Mesa [28], Krowdix [6] and Multi-Agent Scalable Runtime platform for Simulation (MASeRaTi) [2].

Cormas [7] is a general ABSS platform dedicated to natural and common resource management. There is a work [36] that models a social network of innovation diffusion in the medical domain. Madkit [14] is a general multiagent platform which relies on organization concepts and includes simulation facilities. Kodia et al. [21] describe a model where investors are tied by relationships such as friendship, trust and privacy. Mason [24] is a popular multiagent simulation. There is an extension *socialnets* that provides simple network statistics and a bridge to the Java Graph library Jung.[1] Some authors [40] have used Mason for

[1] http://jung.sourceforge.net/.

Table 1. Review of ABSS platforms

Name	Domain	Language	SNs	SNA	OS
Cormas	Generic	VisualWorks	✓	✗	✓
NetLogo	Generic	NetLogo, Scala & Java	✓	✗	✓
Swarm	Generic	Objective-C, Java	✓	✗	✓
MadKit	Generic	Java	✓	✗	✓
MASON	Generic	Java	✓	✗	✓
Repast	Generic	Java	✓	✓	✓
SeSam	Generic	Java	✗	✗	✓
MASeRaTi	Generic	Java	✗	✗	✓
Mesa	Generic	Python	✗	✗	✓
UbikSim	AmI	Java	✗	✗	✓
EscapeSim	Evacuation	Java	✗	✗	✓
HashKat	Social networks	C++	✓	✓	✓
Krowdix	Social networks	Java	✓	✓	✗
Soil	Social networks	Python	✓	✓	✓

modelling viral marketing in Twitter. To this end, authors usually complement Mason with other libraries and tools e.g. GraphStream[2] for synthetic network generation and dynamic network visualisation, iGraph[3] for centrality and network measures, and Gephi[4] for detailed analysis of the network. NetLogo [35] is a multiagent programming and simulation environment. It includes facilities for network representations although not for network analysis. An outdated extension to NetLogo is described in [5], where the network analysis and visualisation tool Pajek[5] is integrated. In addition, there are some available models of social networks (e.g. Social circles [15]), but they do not provide facilities for analysing or building new models. Repast [32] is an agent based simulation platform that provides a large library of simulation models. Repast has been extended for SNA [19]. The library Repast Social Network Analysis (ReSoNetA) adds network functionality to RepastJ. It provides a number of network metrics (centrality, prestige and authority) based on the graph Java library Jung as well as visualisation facilities. This library exploits Repast's built-in facilities for network modelling. In addition, other works such as van Maanen [25] have used Repast for modelling social influence in Twitter. SeSam [16] provides a generic environment for agent based simulations but it has not been applied for social network modelling. Swarm [27] is a well known agent-based simulator that has been applied to social network problems such as open source project dynamics [27].

[2] http://graphstream-project.org/.
[3] http://igraph.org/.
[4] https://gephi.org/.
[5] http://mrvar.fdv.uni-lj.si/pajek/.

While the previous ABSS platforms were designed for its application to a wide variety of domains, other platforms, such as UbikSim [9] and EscapeSim [41] has been specifically designed for a particular domain, such as Ambient Intelligence (AmI) and evacuation.

HashKat [38] is a C++ ABSS platform specifically designed for the study and simulation of social networks. It includes facilities for network growth and information diffusion, based on a kinetic Monte Carlo model. It exports information to be processed by machine learning libraries such as NetworkX[6] or R's iGraph and network visualisation with Gephi.

Mesa [28] is an ABSS platform that aims at providing a Python alternative to traditional Netlogo, MASON or Repast. It enables in-browser visualisation and takes advantage of Python ecosystem. Krowdix [6] is a Java ABSS for social networks but it is not open source. It uses JUNG for network functions and JFreeChart[7] for visualisation. The simulation model considers users, their relationships, user groups and interchanged contents. It has been applied to Twitter and Facebook. MASeRaTi [2] is a distributed and scalable ABSS that uses the Belief-Desire-Intention (BDI) framework lightjason [3], that extends the agent-oriented programming language AgentSpeak.

To summarise, except for HashKat and Krodix, ABSS platforms do not provide support for the analysis of social networks, although some platforms have already been used for this purpose. Moreover, most ABSS platforms are programmed in Java. MASeRaTi follows a different approach where agents can be programmed based on a BDI model. Main challenges for applying existing platforms to social networks come from their underlying models, frequently tied to spatial models.

3 ABSS Requirements for Social Networks

Based on the previously presented review of ABSS platforms and their application to SN analysis and simulation, we have identified the requirements listed below, which are structured in network and agent model.

Network model. The network level groups all the functionalities related to the structural aspects of the social network. The following requirements have been identified:

– *Generation of synthetic graphs.* Even though accessing real social network graphs is critical, real datasets have a number of disadvantages [39]. First, sharing large social graphs is challenging, since they should be anonymised and there are limitations in the way they can be shared (for example, only tweet ids can be shared in Twitter, which requires collecting the dataset with API restrictions and difficulties in reproducing the original dataset since some tweets could be no longer available). Second, the availability of a small number of social graphs can limit the statistical confidence in the experimentation

[6] https://networkx.github.io/.
[7] http://www.jfree.org/jfreechart/.

results. Finally, obtaining real datasets suitable for the desired experimentation can be difficult and require a great effort. Thus, synthetic graph generated by measurement-calibrated graph models [39] so that graph models are fitted to a real social graph, and the simulation are realistic. The platform should provide implementation of classical social graph models [39] (e.g. Barabasi-Albert model [4], Random Walk [44], etc.) and should be extensible to innovative models.
- *Graph traversing and visualisation.* The platform should provide functionalities for traversing social graphs and visualising social structure, in order to be applied to diffusion models [13].
- *SNA functionalities.* Several functionalities should be available for the analysis of the social graph, such as calculation of social metrics (e.g. centrality, betweenness, etc.) as well as algorithms for community detection.
- *Export and import of network model.* There should be facilities for importing and exporting social graphs, based on popular formats such as Graph Modelling Language (GML) [18], GraphML [8] and Graph Exchange XML Format (GEXF) [12].

Agent model. The agent level models the agent characteristics, their state, how agent state evolves in every simulation step. Following the modelling steps proposed by Macal and North [26], we outline the requirements for social network modelling. Platform should allow users to: (i) define agent type definition and attributes (e.g. sentiment, frequency of tweeting, number of followers, etc.); (ii) define interactions with the environment, that represent external factors to agent decision, such as news or market evolution; and (iii) specify methods to update agent state based on their interaction with other agents and the environment. This include the capability to update the agent social network (i.e. creation or modification of social links).

Non functional requirements. Regarding non functional requirements, several aspects have been considered. First of all, the *programming language* is an important decision. In order to provide a homogeneous programming environment, network and machine learning libraries should be available. Both Java and Python fulfill these requirements, as we have introduced previously. As previously outlined, it is very important that ABSS provide *interactive experimentation facilities* that enable researchers to run and define their experiments. In this regard, most platforms ABSS platforms such as Mason or Repast provide configurable and extensible *configuration facilities* [43]. *Scalability* has been recently addressed by a number of researchers [1,2]. The ability to distribute agents across machines or big data processing infrastructures can be required for the simulation of large scale social networks. Finally, *extensibility and reusability* of simulation models should be encouraged [37], so that researchers can benefit from a library of tested simulation models that can be used, extended and adapted to model new behaviours.

4 Soil Platform

4.1 Design Decisions

The first design decision is the selection of Python [30], given its increased popularity, its very gradual learning curve, readability, clear syntax and availability of libraries for network processing and machine learning. In addition, we consider the interactive analysis of the IPython interface[8] very beneficial for simulation. From the reviewed platforms, only one platform is available in Python, Mesa, but it does not provide network facilities yet and is still in constant evolution. Hence, we evaluated different options to extend Mesa for this scenario. Another alternative was to extend nxsim[9], a Python library that provides a basic ABSS framework, based on Simpy [29]. We eventually chose nxsim due to its simplicity and robustness.

Regarding the network model, we have opted for NetworkX, which is the defacto standard library for SNA analysis of small to medium networks. For massive networks, the transition to NetworkKit [42] is straight forward. NetworkX provides functionalities for manipulating and representing graphs, generators of classical and popular graph models, and graph algorithms for analysing graph properties. In addition, NetworkX is interoperable with a great number of graph formats, including GML, GraphML JSON and GEXF.

For network visualization, we have selected Gephi, an open-sourced software for network and graph interactive analysis. Gephi is able to render in 3D and real-time large and complex networks. In addition, both NetworkX and Gephi support the format GraphML, so a graph generated with NetworkX can be explored with Gephi in every simulation step. Finally, configurability will be achieved with configuration files.

4.2 Simulation Model for Social Networks

We propose a simulation model of SNs consisting of users represented by agents and a network that represents the social links between users. Agent are characterised by their state (e.g. infected) and the behaviours they can carry out in every simulation step, usually depending on the user state. Each behaviour defines the actions carried out (e.g. tweeting, following a user, etc.) and how the agent state evolves, depending on external factors (e.g. news about a topic) or social factors (e.g. opinion of their friends). Probabilities defined in the configuration control the frequency of actions in every behaviour.

This simulation model has been implemented in the architecture shown in Fig. 1 and consists of four main components.

The *NetworkSimulation* class is in charge of the network simulator engine. It provides forward-time simulation of events in a network based on nxsim and Simpy. Based on configuration parameters, a graph is generated with NetworkX

[8] https://ipython.org/.
[9] https://pypi.python.org/pypi/nxsim.

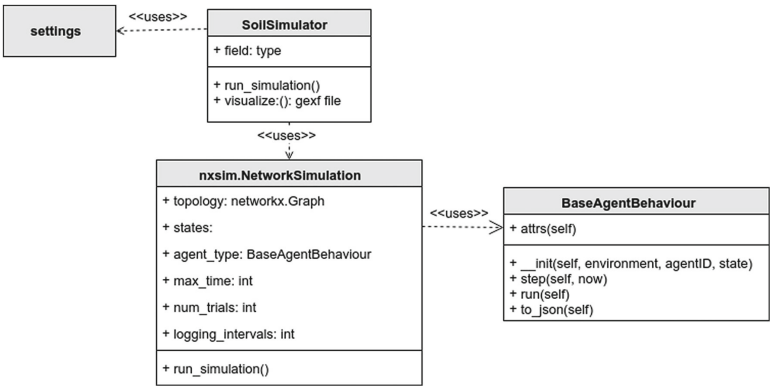


Fig. 1. Simulation components

and an agent class is populated to each network node. The main parameters are the network type, number of nodes, maximum simulation time, number of simulations and timeout between each simulation step.

The *BaseAgentBehaviour* class is the basic agent behaviour that should be extended for each social network simulation model. It provides a basic functionality for generation of a JSON file with the status of the agents for its analysis with machine libraries such as Scikit-Learn.

The *SoilSimulator* class is in charge of running the simulation pipeline defined in Sect. 4.3, which consists in running the simulation and generating a visualisation file in GEXF which can be visualised with Gephi. In addition, interactive analysis can be done through IPython notebooks.

Settings groups the general settings for simulations and the settings of the different models available in Soil's simulation model library.

4.3 Simulation Workflow

An overview of the system's flow is shown in Fig. 2. The simulation workflow consists of three steps: configuration, simulation and visualization.

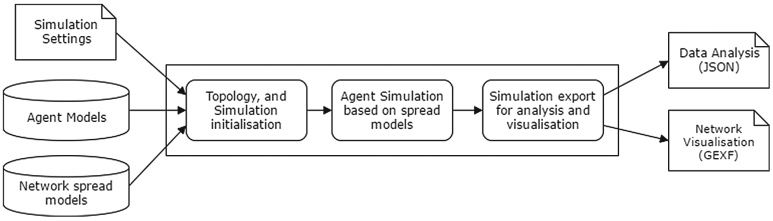


Fig. 2. Social simulator's workflow

In the first step, the main parameters of the simulation are configured in the *settings.py* file. The main parameters are: network graph type, number of agents, agent type, maximum time of simulation and time step length. In addition, the parameters of the behaviour model should be configured (e.g. initial states or probability of an agent action). Agent behaviours should be selected from the provided library or developed extending the *BaseAgentBehaviour* class.

Once the simulation is configured, the next step is the simulation, that can be done step by step or a number of steps. The class *BaseAgentBehaviour* stores the status of every agent in every simulation step into a JSON file to be exported once the simulation is finished. This allows us to automatise the process of generating the .gexf file.

Finally, users can carry out further analysis with the JSON file as well as visualize the evolution the simulation with the generated .gexf file with the tool Gephi, as shown in Fig. 5.

5 Test Cases

We have evaluated Soil in the development of a number of simulation models. In these experiments, we have used the Barabasi-Albert network generation model [4].

The models included in the library deal with viral marketing in Twitter [40], infection (SISa [17]), sentiment correlation the social network Weibo [10], Bass model [35] and Independent Correlation Model [35] of information diffusion in social networks.

In order to illustrate the functionalities of Soil, we review the Viral Marketing model [40], which is based on rumour propagation models. In it, agents have four potential states: neutral, infected, vaccinated and cured. This model includes the fact that infected users who made a mistake believing in the rumour will not be

```
class ControlModelM2(BaseBehaviour):
# init states
def step(self, now):
if self.state['id'] == 0:  #Neutral
self.neutral_behaviour()
elif self.state['id'] == 1:  #Infected
self.infected_behaviour()
...
def infected_behaviour(self):
# Infected
neutral_neighbors = self.get_neighboring_agents(state_id=0)
for neighbor in neutral_neighbors:
if random.random() < self.prob_infect:
neighbor.state['id'] = 1  # Infected
```

Fig. 3. Code snippet of an infected behaviour

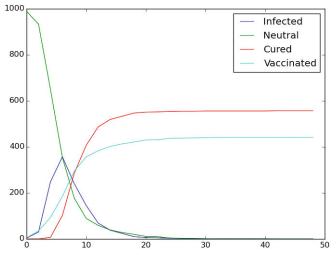


Fig. 4. Agent evolution

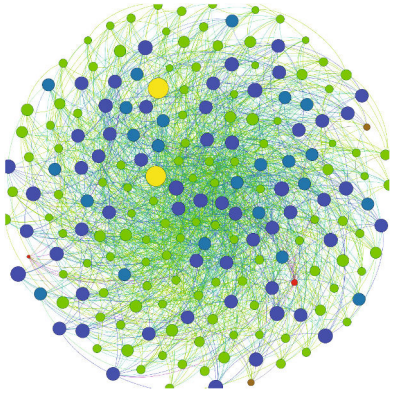


Fig. 5. Network visualization

in favour of spreading their mistakes through the network. A example of how behaviours are programmed is shown in Fig. 3. This behaviour shows that an infected agent first selects its neutral neighbours and infects them with a given probability. Figures 4 and 5 show the evolution of agent states and network visualisation, respectively.

6 Conclusions and Outlook

While generic ABSS provide a suitable framework, we think that further research on ABSS platforms for specific domains is needed. In this paper we have reviewed the existing frameworks and the requirements for modelling and simulation of social networks.

Soil is a modern ABSS for social networks developed in Python that benefits from the Python ecosystem. It has been applied to a number of social network simulation models, ranging from rumour propagation to emotion propagation and information diffusion. Additionally, it is fully open source, cross-platform and produces outputs compatible with SNA packages and network visualisation tools. The platform has been designed for research purposes, and has focused on simplicity of developing new simulation models. Soil allows the generation of dynamic networks and its animation thanks to the use of Gephi. In spite of the growing development of the Python ecosystem, there are still some functionalities, such as Exponential Random Graph Model (ERGMs) which are better supported in other environments such as R with the statnet[10] package, which provides a wide range of functionality for the statistical analysis of social networks. In particular, these models are very interesting for fitting models given a network data set. As future work, we aim at evaluating and integrating implementations such as ergm[11].

[10] http://statnetproject.org.
[11] https://github.com/jcatw/ergm.

Lastly, Soil is work in progress. We aim at improving the experimentation and visualisation facilities provided by the platform, and improve the platform through its application in more use cases and through the collaboration with other research groups.

Acknowledgements. This work is supported by the Spanish Ministry of Economy and Competitiveness under the R&D projects SEMOLA (TEC2015-68284-R) and EmoSpaces (RTC-2016-5053-7), by the Regional Government of Madrid through the project MOSI-AGIL-CM (grant P2013/ICE-3019, co-funded by EU Structural Funds FSE and FEDER), and by the European Union through the project MixedEmotions (Grant Agreement no: 141111).

References

1. Aaby, B.G., et al.: Efficient simulation of agent-based models on multi-GPU and multi-core clusters. In: Proceedings of the 3rd International ICST Conference on Simulation Tools and Techniques, SIMUTools 2010, Torremolinos, Malaga, Spain. ICST (2010)
2. Ahlbrecht, T., Dix, J., Köster, M., Kraus, P., Müller, J.P.: A scalable runtime platform for multiagent-based simulation. In: Dalpiaz, F., Dix, J., Riemsdijk, M.B. (eds.) EMAS 2014. LNCS, vol. 8758, pp. 81–102. Springer, Cham (2014). doi:10.1007/978-3-319-14484-9_5
3. Aschermann, M., et al.: LightJason: a BDI framework inspired by Jason. Technical report IfI-16-04, Depart. Department of Computer Science, TU Clausthal, Germany (2014)
4. Barabáasi, A.-L., Albert, R.: Emergence of scaling in random networks. Science **286**(5439), 509–512 (1999)
5. Berryman, M.J., Angus, S.D.: Tutorials on agent-based modelling with NetLogo and network analysis with Pajek. In: Complex Physical, Biophysical and Econophysical Systems, vol. 1. World Scientific, Hackensack (2010)
6. Blanco-Moreno, D., Cárdenas, M., Fuentes-Fernández, R., Pavón, J.: Krowdix: agent-based simulation of online social networks. In: Bazzan, A.L.C., Pichara, K. (eds.) IBERAMIA 2014. LNCS, vol. 8864, pp. 587–598. Springer, Cham (2014). doi:10.1007/978-3-319-12027-0_47
7. Bommel, P., et al.: Cormas, an agent-based simulation platform for coupling human decisions with computerized dynamics. In: ISAGA 2015: Hybrid Simulation and Gaming in the Network Society (2015)
8. Brandes, U., et al.: Graph markup language (GraphML). In: Handbook of Graph Drawing and Visualization 20007 (2013)
9. Campuzano, F., Garcia-Valverde, T., Garcia-Sola, A., Botia, J.A.: Flexible simulation of ubiquitous computing environments. In: Novais, P., Preuveneers, D., Corchado, J.M. (eds.) Ambient Intelligence - Software and Applications. AINSC, vol. 92, pp. 189–196. Springer, Heidelberg (2011)
10. Fan, R., et al.: Anger is more influential than joy: sentiment correlation in Weibo. In: CoRR abs/1309.2402 (2013)
11. Granovetter, M.: The impact of social structure on economic outcomes. J. Econ. Perspect. **19**(1), 33–50 (2005)
12. Group, G.W.: GEXF file format. GEXF Working Group (2009)

13. Guille, A., et al.: Information diffusion in online social networks: a survey. ACM SIGMOD Rec. **42**(2), 17–28 (2013)
14. Gutknecht, O., Ferber, J.: The MadKit agent platform architecture. In: Wagner, T., Rana, O.F. (eds.) AGENTS 2000. LNCS, vol. 1887, pp. 48–55. Springer, Heidelberg (2001). doi:10.1007/3-540-47772-1_5
15. Hamill, L., Gilbert, N.: Social circles: a simple structure for agent-based social network models. J. Artif. Soc. Soc. Simul. **12**(2), 3 (2009)
16. Herrler, R., Fehler, M.: SeSAm: implementation of agent based simulation using visual programming. In: Components (2006)
17. Hill, A.L., et al.: Emotions as infectious diseases in a large social network: the SISa model. Proc. Roy. Soc. Lond. B Biol. Sci. **277**(1701), 3827–3835 (2010)
18. Himsolt, M.: GML: a portable graph file format. Technical report. Universität Passau (1997)
19. Holzhauer, S.: Developing a Social Network Analysis and Visualization Module for Repast Models, vol. 4. Kassel University Press GmbH, Kassel (2010)
20. Kiesling, E., et al.: Agent-based simulation of innovation diffusion: a review. CEJOR **20**(2), 183–230 (2012)
21. Kodia, Z., Said, L.B., Ghedira, K.: Stylized facts study through a multi-agent based simulation of an artificial stock market. In: Li Calzi, M., Milone, L., Pellizzari, P. (eds.) Progress in Artificial Economics. Lecture Notes in Economics and Mathematical Systems, vol. 645, pp. 27–38. Springer, Heidelberg (2010). doi:10.1007/978-3-642-13947-5_3
22. Korda, H., Itani, Z.: Harnessing social media for health promotion and behavior change. Health Promot. Pract. **14**(1), 15–23 (2013)
23. Liu, D., Chen, X.: Rumor propagation in online social networks like twitter - a simulation study. In: 2011 Third International Conference on Multimedia Information Networking and Security, November 2011
24. Luke, S.: MASON: a multiagent simulation environment. Simulation **81**, 517–527 (2005)
25. van Maanen, P.-P., van der Vecht, B.: An agent-based approach to modeling online social influence. In: Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining. ACM (2013)
26. Macal, C.M., North, M.J.: Tutorial on agent-based modeling and simulation. In: Simulation Conference, 2005 Proceedings of the Winter. IEEE (2005)
27. Madey, G., et al.: Agent-based modeling of open source using Swarm. In: AMCIS 2002 Proceedings (2002)
28. Masad, D., Kazil, J.: MESA: an agent-based modeling framework. In: Proceedings of the 14th Python in Science Conference (SCIPY 2015) (2015)
29. Matloff, N.: Introduction to discrete-event simulation and the Simpy language. In: Davis, CA. Dept of Computer Science. University of California at Davis. Retrieved 2 Aug 2008
30. McKinney, W.: Python for Data Analysis. O'Reilly, Sebastopol (2012)
31. Nikolai, C., Madey, G.: Tools of the trade: a survey of various agend based modeling platforms. J. Artif. Soc. Soc. Simul. **12**(2), 2 (2009)
32. Ozik, J., Collier, N., Combs, T., Macal, C.M., North, M.: Repast simphony statecharts. J. Artif. Soc. Soc. Simul. **18**(3), 11 (2015). http://jasss.soc.surrey.ac.uk/18/3/11.html
33. Padilla, J.J., et al.: Leveraging social media data in agentbased simulations. In: Proceedings of the 2014 Annual Simulation Symposium. Society for Computer Simulation International (2014)

34. Railsback, S.F., et al.: Agent-based simulation platforms: review and development recommendations. Simulation **82**(9), 609–623 (2006)
35. Rand, W., Wilensky, U.: An Introduction to Agent-Based Modeling: Modeling Natural, Social, and Engineered Complex Systems with NetLogo. MIT Press, Cambridge (2015)
36. Ratna, N.N., et al.: Diffusion and social networks: revisiting medical innovation with agents. In: Qudrat-Ullah, H., Spector, J.M., Davidsen, P.I. (eds.) Complex Decision Making, pp. 247–265. Springer, Heidelberg (2008)
37. Robinson, S., et al.: Simulation model reuse: definitions, benefits and obstacles. Simul. Model. Pract. Theory **12**(7), 479–494 (2004)
38. Ryczko, K., et al.: Hashkat: large-scale simulations of online social networks. In: arXiv preprint arXiv (2016)
39. Sala, A., et al.: Measurement-calibrated graph models for social network experiments. In: Proceedings of the 19th International Conference on World Wide Web. ACM (2010)
40. Serrano, E., Iglesias, C.A.: Validating viral marketing strategies in Twitter via agent-based social simulation. Expert Syst. Appl. **50**(1), 140–150 (2016)
41. Serrano, E., et al.: Towards a holistic framework for the evaluation of emergency plans in indoor environments. Sensors **14**(3), 4513–4535 (2014)
42. Staudt, C., et al.: NetworKit: an interactive tool suite for high-performance network analysis. In: CoRR abs/1403.3005 (2014)
43. Szufel, P., et al.: Controlling simulation experiment design for agent-based models using tree representation of parameter space. Found. Comput. Decis. Sci. **38**(4), 277–298 (2013)
44. Vázquez, A.: Growing network with local rules: preferential attachment, clustering hierarchy, and degree correlations. Phys. Rev. E **67**(5), 056104 (2003)
45. Wang, F.Y., et al.: Social computing: from social informatics to social intelligence, March 2007

CHAPTER 4

General Discussion, Conclusions and Future Research

This chapter provides a general discussion of the work in this thesis, including an overview of the solutions proposed, an analysis of the results, conclusions, and future lines of research.

4.1 Overview

In Section 1.1 we introduced four areas for improvement in sentiment analysis, as well as the advantages of each of them (Figure 1.1). We also identified a series of challenges in each area. The ultimate goal of this thesis was to help sentiment analysis grow in those areas by tackling these challenges. Overall, the main contributions are: 1) Onyx, a vocabulary for emotions, models of emotions; 2) vocabulary and schemas for language resources and sentiment analysis services; 3) an architecture for sentiment analysis services and its reference implementation; and 4) a model of social context (i.e., context of content and users in a social network). Figure 4.1 shows how each these and other contributions fit into the areas of improvement, and the remaining of the section discusses how these contributions tackle the initial challenges.

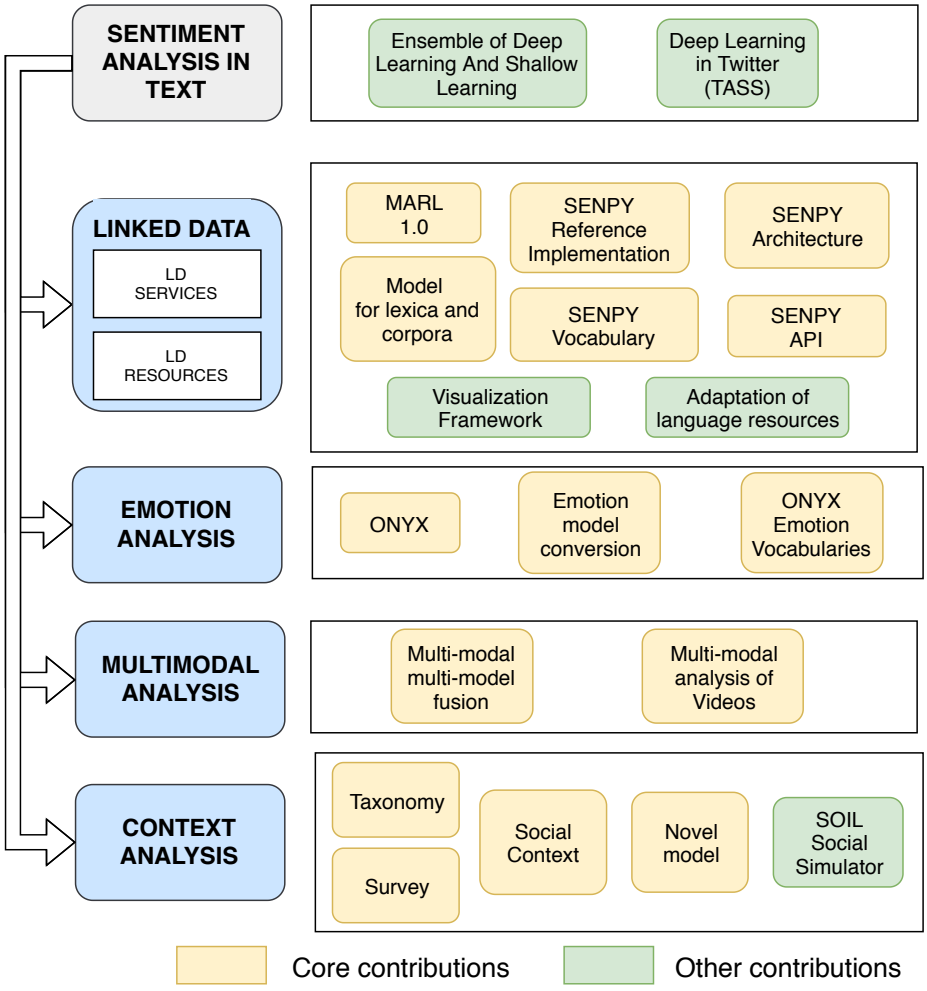


Figure 4.1: Summary of contributions, grouped by type of analysis.

The challenges we identified were the following: 1) lack of interoperability, i.e., heterogeneity of formats and schemas; 2) underrepresentation of emotion analysis; 3) difficulty to integrate with other types of analysis (multimodality); and 4) disregard for contextual information.

To bring interoperability, we proposed a set of vocabularies and schemas for sentiment analysis services and language resources. We also designed a generic architecture for sentiment analysis services (Senpy), and developed a reference implementation. The architecture is described in the paper **"Senpy: A Pragmatic Linked Sentiment Analysis Framework"** (Section 3.2.4), and the reference implementation in **"Senpy: A framework for semantic sentiment and emotion analysis services"** (Section 3.2.1). On the language resources side, this made it possible to create a multilingual portal for language resources, which could be automatically converted and enriched using linked data sources. On the services side, it has enabled the creation of multitude of new services, and the introduction of novel features such as automated evaluation and pipelining of results.

To foster research on emotion analysis, we published a vocabulary for emotions and emotion analysis, and we adapted several emotion models from other projects (i.e., EmotionML and WordNet-Affect) to be used together with the main vocabulary. This has enabled the creation of semantic emotion analysis services, and it paved the way for multi-modal analysis. The result is described in **"Onyx: Describing Emotions on the Web of Data"** (Section 3.1.1) and **"Onyx: A Linked Data Approach to Emotion Representation"** (Section 3.1.2).

Regarding multimodality, analyses in different modalities can now use the same URIs for their content, which allows for multimodal fusion. This is achieved by using a new URI scheme that can be used to unify URIs. Moreover, since different modalities tend to use different emotion models, we leveraged Onyx to perform automatic model conversion. The approach is summarized in **"Multimodal Multimodel Emotion Analysis as Linked Data"** (Section 3.2.2).

To help develop new interoperable services, we also provided an architecture for advanced sentiment and emotion analysis services in **"Senpy: A Pragmatic Linked Sentiment Analysis Framework"** (Section 3.2.4). The architecture takes into account features such as automatic evaluation of different services. All these features have been integrated in the reference implementation of the architecture, described in **"Senpy: A framework for semantic sentiment and emotion analysis services"** (Section 3.2.1).

Lastly, we helped unify the terminology and ease the creation of new models that exploit social context for sentiment and emotion analysis. We did so by providing a formal definition

of Social Context, as well as a methodology to compare different approaches. We also used this methodology to summarize and compare the state of the art in the field. The results are published in **"Social context in sentiment analysis: Formal definition, overview of current trends and framework for comparison"** (Section 3.3.1).

To conclude, a different way to analyze these contributions is through the fields of knowledge to which they belong. If we do so, we find three main fields that are thematically different, yet interconnected: 1) definition of linked data vocabularies; 2) development and use of linked data technologies; and 3) exploitation of social context. For the purposes of this thesis, those fields are focused on three types of results: 1) language resources; 2) sentiment analysis services; 3) analysis models. This is all illustrated in Figure 4.2, where each contribution is linked to those fields of knowledge, types of results and existing technologies or approaches. This view is particularly interesting because it shows how interconnected the contributions are. For instance, the It also shows the fact that certain contributions belong to more than one field. For instance, the Senpy vocabulary is very tied to the definition and implementation of analysis services.

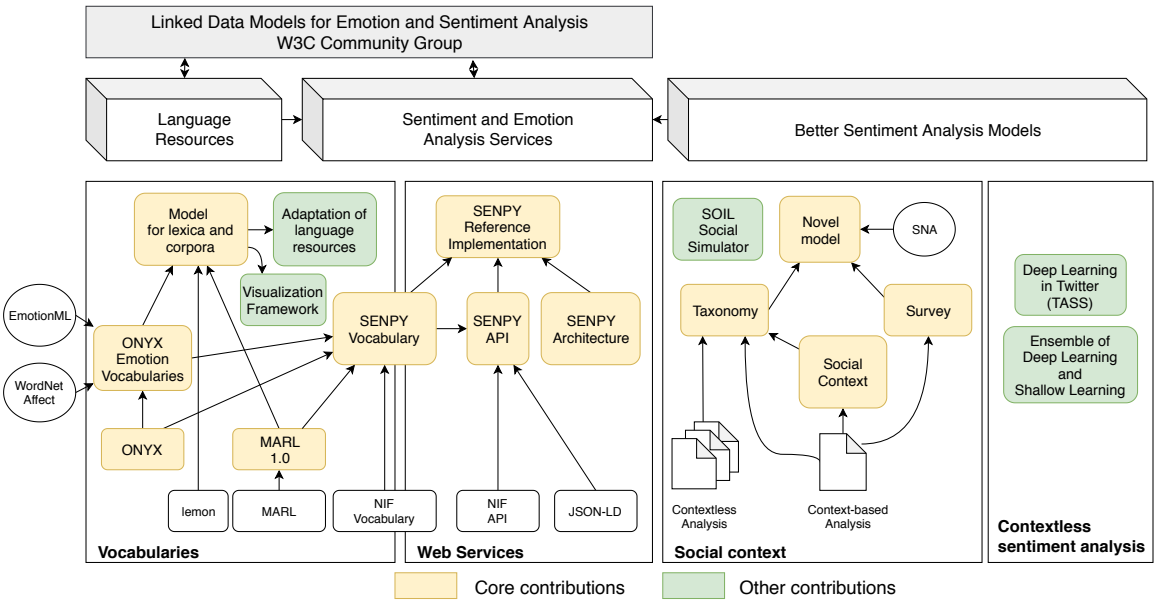


Figure 4.2: Summary of contributions in each field, and their relation to the existing body of knowledge.

4.2 Scientific results

This section discusses the scientific results and their relationship to the initial objectives. Each subsection contains a list of publications that contributed in some way to achieving one of the objectives. For the sake of brevity, the table only contains titles, impact (if

available) and summary of contributions. More details for each publication can be found on the referenced page, including abstract and full text.

4.2.1 Objective 1: Definition of a vocabulary for emotions

The first work in this thesis was to extend the Marl vocabulary (Westerski, Carlos A. Iglesias, and Tapia, 2011). Marl is a vocabulary for opinions in the web of data. Its original focus was on the opinion, with sentiment analysis as one of its use cases. It did not cover the concept of a sentiment analysis activity, or the relationship between the analysis and the results. Both of those ideas are very important when modelling sentiment analysis services and their output. To the best of knowledge, there were no vocabularies that modeled a sentiment analysis process. The relationship between processes and their output had been thoroughly modeled by the provenance ontology (PROV-O).

Instead of creating an independent vocabulary with the new classes, it was decided to extend Marl and leverage its popularity in the sentiment analysis community. Hence, the concept of Sentiment Analysis was introduced, as a PROV-O activity, and Marl's Opinion also subclassed PROV-O Entity. With this extension, Marl was aligned with the PROV-O ontology. The change to the ontology was minimal, but it laid the foundation to what later became a community group recommendation. The result is available online [1]. The extension of Marl has been key in the development of the Objective-3 and Objective-4.

Once the modification to Marl was successful, we addressed the lack of a widespread vocabulary for emotions. The goal was to allow for the same level of expressiveness for both sentiment and emotion analysis. When compared to sentiment analysis, emotion analysis suffers from two main drawbacks. First of all, it is not as popular. Second of all, there are several competing models of emotions in psychology. Some of these models are more popular than others, but there is no real consensus in the community. In our search for a vocabulary, we evaluated different alternatives. The most popular of these alternatives is Emotion-ML, a mark-up language for emotions. However, Emotion-ML does not provide a semantic vocabulary, so it could not be used in linked data applications. Based on the experience with the extension of Marl, we decided to create a new vocabulary for emotions, just as Marl did for opinions. This new vocabulary, which we named Onyx, was also aligned with PROV-O. It can be used to annotate any entity with emotion, although the main targets are the results from emotion analysis services and all the types of language resources involved (e.g. corpora and lexicons). This vocabulary can connect results from different providers and applications, even when different models of emotions are used. At its core, the

[1] `https://www.gsi.upm.es/ontologies/marl`

ontology has three main classes: *Emotion*, *EmotionAnalysis* and *EmotionSet*. In a standard emotion analysis, these three classes are related as follows: an *EmotionAnalysis* is run on a source (which is generally text, e.g. a status update), the result is represented as one or more *EmotionSet* instances that contain one or more *Emotion* instances each. To remedy the lack of consensus on what model of emotions to use, Onyx follows a similar approach to Emotion-ML: it provides a meta-model for emotions, and different models can be defined separately. The election of existing emotion models, or the creation of new ones, is left to the user. In Emotion-ML these independent models are called vocabularies (not to be confused with semantic vocabularies such as Onyx or Marl), and the Emotion-ML group has provided the definition of the most commonly used models. To make Onyx more usable, and to encourage model re-use, we also adapted these existing vocabularies from Emotion-ML, and created equivalent Onyx EmotionModel instances with the appropriate classes and/or dimensional properties. We also converted WordNet-Affect's a-labels (affective labels for concepts) to an Onyx emotion model, and a taxonomy of emotions, using the SKOS ontology. The models based on Emotion-ML vocabularies and on WordNet-Affect have been published online. Following the best practices in the semantic web, new users are encouraged to use these models intead of defining their own. The **adaptation of Emotion-ML vocabularies and WordNet-Affect labels to Onyx** was included in the Onyx publication. Onyx was originally published as a conference paper, **"Onyx: Describing Emotions on the Web of Data"** (Section 3.1.1), and later on it was published as an journal article in Information & Management: **"Onyx: A Linked Data Approach to Emotion Representation"** (Section 3.1.2). The creation of Onyx and its vocabularies fulfilled the first objective in the thesis, **Objective-1**.

Furthermore, Onyx and Marl have since been used to annotate lexica and other language resources with sentiment and emotion. e.g., the paper **"Generating Linked-Data based Domain-Specific Sentiment Lexicons from Legacy Language and Semantic Resources"** (Section 3.1.8). It has also been used extensively in the Senpy framework, in emotion analysis services compatible with Senpy (especially as part of EuroSentiment and MixedEmotions), and in the creation of semantically annotated language resources.

Table 4.1: Publications related to Objective 1

Page	Title	Impact	Contribution
91	Linguistic Linked Data for Sentiment Analysis		First use of Marl in language resources, and introduction of Onyx.

Table 4.1: Publications related to Objective 1

30	Onyx: Describing Emotions on the Web of Data		A first version of Onyx, the model for Emotion representation, and examples of how it could be used to annotate language resources, a brief comparison with EmotionML notation, and a proof of concept of service that uses the model. (**Objective-1**)
43	Onyx: A Linked Data Approach to Emotion Representation	Q1 (2.391)	A complete version of Onyx (the emotion vocabulary), with examples of use in both language resources and emotion analysis services. It also exemplifies novel applications enabled by representing emotions using semantic technologies: emotion mapping using SPIN. (**Objective-1**)
-	Linked Data Models for Sentiment and Emotion Analysis in Social Networks	Book chapter	It consolidates and promotes concepts from our earlier publications.

4.2.2 Objective 2: Definition of a model to annotate language resources and to be used in analysis services

Once the appropriate models for sentiment and emotion existed, the third task was to define a model for language resources, and a base API for sentiment and emotion analysis services. In particular, we consider two kinds of resources: lexicons, which are roughly dictionaries that map lexical elements (e.g., word) to emotions or sentiment, and corpora, or collections of annotated entries (e.g., tweets). The nature of corpora makes their annotation very similar to service results. Hence, we focused on modelling lexicons first. The API would include both the schema and representation for input and output, as well as the set of endpoints, HTTP verbs and arguments to accept. The former belongs to the **vocabularies** side of this thesis, whereas the second is in the **services** side.

The model for lexicons would need to integrate already existing and popular vocabularies. In particular, we decided to base our model on lemon McCrae, Spohr, and Cimiano, 2011a, a model for modeling lexicon and machine-readable dictionaries and linked to the Semantic Web and the Linked Data cloud. Lemon supports the linking of a computational lexical resource with the semantic information defined in an ontology Lemon defines a set of basic aspects of lexical entries, including its morpho-syntactic variants and normalizations. Lexical entries can be linked to semantic information through lexical sense objects. In addition, lemon has a number of modules that allow for modeling different aspects of a lexicon. The list of currently defined modules includes: linguistic description, phrase structure, morphology, syntax and mapping, and variation. To create a sentiment or emotion lexicon, or to add sentiment and emotion annotations to an existing lexicon, the only requisite is that each lemon lexical entry in the resource needs to include a sentiment (`marl:hasOpinion`) or

emotion (`onyx:hasEmotionSet`) annotation. Listing 4.1 shows the combination of lemon and Onyx can be used to annotate lexical resources with emotion. The case of opinions is similar, yet simpler, since the model of opinions is less complex than that of emotions.

Listing 4.1: Example of a sentiment lexicon entry

```
:susto a lemon:Lexicalentry ;
 lemon:canonicalForm [ lemon:writtenRep "susto"@es ] ;
 lemon:sense [ lemon:reference wn:synset-fear-noun-1 ;
 onyx:usesEmotionModel emoml:pad ;
 onyx:hasEmotionSet [
  onyx:hasEmotion [
    emoml:pad_dominance 4.12 ;
    emoml:pad_arousal 5.77 ;
    emoml:pad_pleasure 3.19 ;
  ]
 ] ;
 lexinfo:partOfSpeech lexinfo:noun .
```

To choose a model for sentiment services and corpora, we examined the state of the art and the most popular services for sentiment analysis. At that time, NIF (the NLP Interchange Format), was the most suitable alternative for NLP services. NIF is a combination of a vocabulary for NLP analysis responses, and an API for such services. Not only was NIF the best alternative, but it also aligned very well with our needs and vision at the time. Unfortunately, NIF had two shortcomings for its adoption in sentiment and emotion analysis.

First of all, neither the vocabulary nor the API take into consideration sentiment and emotion analysis. On the vocabulary side, it would be necessary to decide and document how to combine NIF with vocabularies such as Onyx and Marl. We documented this process, and generated a set of examples to be followed in different scenarios. For instance, Listing 4.2 shows an excerpt of an emotion-annotated corpus. Emotion analysis services would produce very similar outputs, as illustrated in Listing 4.3.

Listing 4.2: Example of emotion annotations in a corpus

```
<http://semeval2014.org/myrestaurant#char=0,80>
 rdf:type nif:RDF5147String , nif:Context;
 nif:beginIndex "0";
 nif:endIndex "80";
 nif:sourceURL <http://tripadvisor.com/myrestaurant.txt>;
 nif:isString "I loved their fajitas and their pico de gallo is not bad and the
     service is correct.";
 onyx:hasEmotionSet <http:///semeval2014.org/myrestaurant/emotion/1>.

<http://semeval2014.org/myrestaurant/emotion/1>
```

```
  rdf:type onyx:EmotionSet;
  prov:generated <http://mixedemotions.eu/analysis/2>;
  onyx:describesObject dbp:Restaurant;
  onyx:describesFeature dbp:Food;
  onyx:hasEmotion [:Emo1, :Emo2].

:Emo1 a onyx:Emotion;
  onyx:hasEmotionCategory emoml:happiness;
  onyx:hasEmotionIntensity 0.7.

:Emo2 a onyx: Emotion;
  onyx:hasEmotionCategory emoml:disgust;
  onyx:hasEmotionIntensity 0.1.

<http://mixedemotions.eu/analysis/2>
  rdf:type onyx:EmotionAnalysis;
  onyx:usesEmotionModel emoml:big6;
  onyx:algorithm "dictionary-based";
  prov:used le:restaurant_en;
  prov:wasAssociatedWith <http://dbpedia.org/resource/UPM>.
```

Listing 4.3: Example of an emotion service output

```
# Service Call: curl --data-urlencode input="My iPad is an awesome device"
# -d informat=text -prefix="http://mixedemotions.eu/example/ipad\#"
# -emodel="wna" "http://www.gsi.dit.upm.es/sa-nif-ws.py"

Service Output:
<http://mixedemotions.eu/example/ipad#char=0,28>
  rdf:type nif:RDF5147String ;
  nif:beginIndex "0" ;
  nif:endIndex "28" ;
  nif:sourceURL < http://mixedemotions.eu/example/ipad>;
  onyx:hasEmotionSet :emotionSet1.

:customAnalysis
  a onyx:EmotionAnalysis;
  onyx:algorithm "SVM";
  onyx:usesEmotionModel wna:WNAModel.

:emotionSet1
  a onyx:EmotionSet;
  prov:wasGenerated :customAnalysis;
  sioc:has_creator [
     sioc:UserAccount <http://www.gsi.dit.upm.es/jfernando>. ];
  onyx:hasEmotion [ :emotion1; emotion2 ]
  onyx:emotionText: "My iPad is an awesome device".

:emotion1
  a onyx:Emotion
  onyx:hasEmotionCategory wna:dislike;
  onyx:hasEmotionIntensity 0.7.

:emotion2
  a onyx:Emotion
  onyx:hasEmotionCategory wna:despair;
  onyx:hasEmotionIntensity 0.1.

<http://mixedemotions.eu/example/ipad#3,6>
  nif:anchorOf "iPad";
  itsrdf:taIdentRef: <http://dbpedia.org/iPad>.
```

On the services side, we would need to extend the API to take the specific needs of sentiment and emotion analysis services into account. We did so, with an extended API that takes emotion and emotion conversion into account. This extension is shown in Table 4.2, which includes all the parameters in the API at the time of writing this document. The elements not originally included in NIF are highlighted.

Furthermore, NIF is a semantic model, and at the time we evaluated it, it only took traditional semantic representation formats into account. i.e., n-triples, XML RDF, etc. This

Table 4.2: The extended NIF-based sentiment and emotion analysis API, which includes parameters to control emotion conversion.

parameter	**description**
input(i)	serialized data (i.e. the text or other formats, depends on informat)
informat (f)	format in which the input is provided: turtle, text (default) or **json-ld**
outformat (o)	format in which the output is serialized: turtle (default), text or **json-ld**
prefix (p)	prefix used to create and parse URIs
minpolarity (min)	minimum polarity value of the sentiment analysis
maxpolarity (max)	maximum polarity value of the sentiment analysis
language (l)	language of the sentiment or emotion analysis
domain (d)	domain of the sentiment or emotion analysis
algorithm (a)	plugin that should be used for this analysis
emotionModel (emodel, e)	emotion model in which the output is serialized (e.g. WordNet-Affect, PAD, etc.)
conversionType	type of emotion conversion. Currently accepted values: 1) full, results contain both the converted emotions and the original emotions, alongside; 2) nested, converted emotions should appear at the top level, and link to the original ones; 3) filtered, results should only contain the converted emotions.

can be a high entrance barrier for developers and users without experience with semantic technologies. We wanted the format for semantic services to also be familiar for these developers, without sacrificing the advantages of semantic technologies. This was the motivation behind the creation of JSON-LD, a subset of JSON with semantics. JSON-LD bridges the gap between pragmatism and semantic correctness. It is a representation format that is familiar to most developers, but fully integrated in the RDF ecosystem. NIF listed JSON-LD as a possible future adition to its formats, but it was not yet included. Nevertheless, we developed a model that resulted in a user-friendly JSON-LD schema in the responses, and we included JSON-LD as the default option in the API, to foster its use by the general public.

Moreover, we meant to cover multi-modal sentiment and emotion analysis. This requires yet another extension to the NIF-based model, to include URI schemes that are compatible with multimedia. Our proposed representation format was published as a conference paper, **"A Linked Data Model for Multimodal Sentiment and Emotion Analysis"** (Section 3.1.4). We collaborated with partners that have expertise in sentiment analysis in audio and video, and worked on the fusion of different modalities. This resulted in a further extension to the model that includes representation for emotion conversion, and fusion of different modalities, as shown in **"Multimodal Multimodel Emotion Analysis as Linked Data"** (Section 3.2.2). These two publications cover **Objective-2**.

Table 4.3: Publications related to Objective 2

Page	Title	Impact	Contribution
86	EUROSENTIMENT: Linked Data Sentiment Analysis		Complete vocabulary for language resources and services, based on MARL and ONYX (**Objective-2**)
113	Generating Linked-Data based Domain-Specific Sentiment Lexicons from Legacy Language and Semantic Resources		Application of the vocabulary for sentiment in language resources to convert legacy resources, which showcases the power of the Linked Data approach.
76	A Linked Data Model for Multimodal Sentiment and Emotion Analysis		The introduction of new URI schemes that allow NIF contexts to link to multimedia fragments.
-	Linked Data Models for Sentiment and Emotion Analysis in Social Networks	Book chapter	It consolidates and promotes concepts from our earlier publications.
68	Towards a Common Linked Data Model for Sentiment and Emotion Analysis		First joint publication of the W3C Community Group.
100	A Linked Data Approach to Sentiment and Emotion Analysis of Twitter in the Financial Domain		Application of the proposed vocabularies and services in a use case.

4.2.3 Objective 3: Definition of a reference architecture for sentiment and emotion analysis services

At this point, we had defined a NIF-based API for services, and a model for both resources and services that combined NIF, Marl, Onyx and the PROV-O ontology, as well as a specific model to represent results from a service. However, our evaluation of the alternatives to develop new services revealed that most researchers would not share their models as a service, and those who did were developing their own solutions from scratch. A third group of researchers were sharing their classifiers not as services, but as extensions to tools such as GATE. The heterogeneity of solutions, and the barrier imposed by having to develop a custom solution, are very detrimental to research in the field, especially for newcomers and for the comparison of different models. This led us to the the definition of a framework for semantic sentiment analysis services (senpy), and its reference implementation. The main motivation behind this framework was to help standardize the APIs and tools used in the sentiment and emotion analysis community, which would foster the creation of more services, especially public ones. The framework presents a modular view of a service, which contains the basic modules in a complete service analysis (e.g., parameter validation), and a series of modules that are necessary to leverage the potential of semantic technologies (Figure 4.3). It also uses the API and models previously defined, which are vital to the interoperability of different services, and to enable most of the features that differentiate semantic services from the rest. e.g., automatic transformations such as emotion model conversion, and service-agnostic evaluation. The architecture was published as a conference paper **"Senpy: A Pragmatic Linked Sentiment Analysis Framework"** (Section 3.2.4) (**Objective-3**), and used later on in the reference implementation.

Table 4.4: Publications related to Objective 3

Page	Title	Impact	Contribution
86	EUROSENTIMENT: Linked Data Sentiment Analysis		Complete vocabulary for language resources and services, based on MARL and ONYX (**Objective-2**)
145	Senpy: A Pragmatic Linked Sentiment Analysis Framework		Definition of the architecture for sentiment and emotion analysis services (**Objective-3**)
125	Multimodal Multimodel Emotion Analysis as Linked Data		Automatic model conversion and proof of concept of multimodal fusion.
132	MixedEmotions: An Open-Source Toolbox for Multi-Modal Emotion Analysis	Q1 (4.292)	Summary of MixedEmotions's results, including the common API and several analysis services in different modalities (audio, video and text) that are interoperable because of it.

Table 4.4: Publications related to Objective 3

118	Senpy: A framework for semantic sentiment and emotion analysis services	Q1 (5.101)	Reference implementation of the architecture, which includes advanced features such as automatic evaluation on a selection of datasets and automatic model conversion (**Objective-4**)

4.2.4 Objective 4: Development of a reference implementation of the architecture

To illustrate the capabilities and feasibility of the framework, we developed a reference implementation. This implementation was targeted to both NLP researchers that developed new classifiers, as well as for consumers of such classifiers, either locally or through web services. Hence, we developed it using a modular architecture, based on plugins, which could host several classifiers (i.e., plugins), and that abstracted away the details of semantics and vocabularies from developers as much as possible. The reference implementation also includes adapters for external services (e.g., meaning cloud and sentiment140) and well known tools (e.g., Vader). Figure 4.4 shows the plugin-based architecture of the reference implementation. The reference implementation was published in the Original Software track of Knowledge-Based Systems **"Senpy: A framework for semantic sentiment and emotion analysis services"** (Section 3.2.1), thus fulfilling **Objective-4**. Senpy[2] has since been used in two European projects (EuroSentiment and MixedEmotions) and in more than 7 bachelor and master theses.

Table 4.5: Publications related to Objective 4

Page	Title	Impact	Contribution
145	Senpy: A Pragmatic Linked Sentiment Analysis Framework		Definition of the architecture for sentiment and emotion analysis services (**Objective-3**)
125	Multimodal Multimodel Emotion Analysis as Linked Data		Automatic model conversion and proof of concept of multimodal fusion.
132	MixedEmotions: An Open-Source Toolbox for Multi-Modal Emotion Analysis	Q1 (4.292)	Summary of MixedEmotions's results, including the common API and several analysis services in different modalities (audio, video and text) that are interoperable because of it.
118	Senpy: A framework for semantic sentiment and emotion analysis services	Q1 (5.101)	Reference implementation of the architecture, which includes advanced features such as automatic evaluation on a selection of datasets and automatic model conversion (**Objective-4**)

[2] `https://github.com/gsi-upm/senpy`

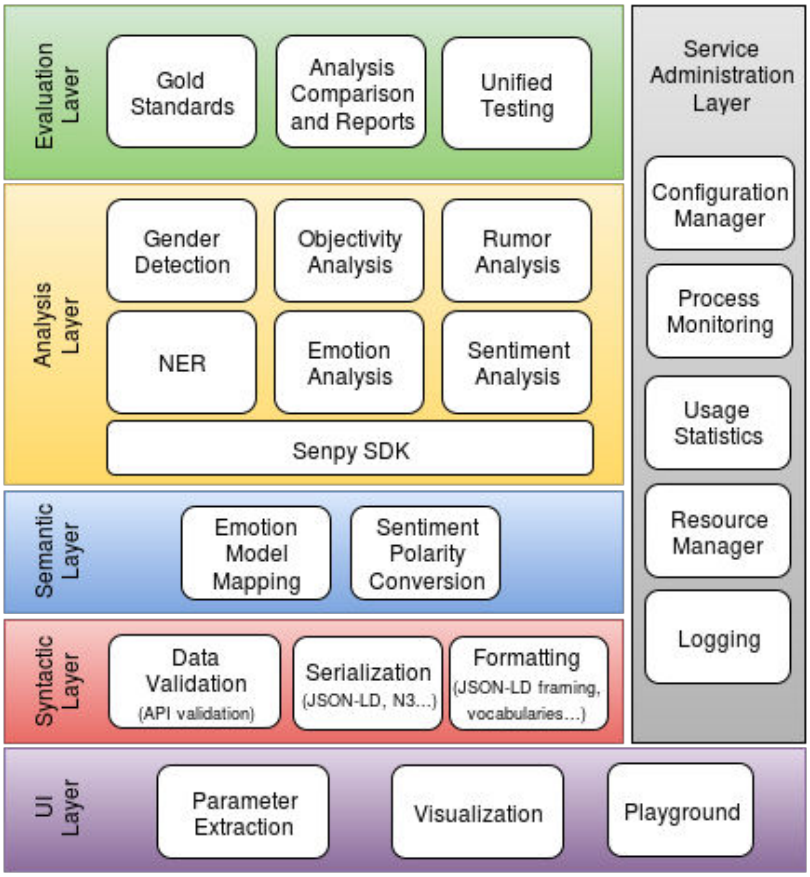


Figure 4.3: Generic architecture for sentiment and emotion analysis services.

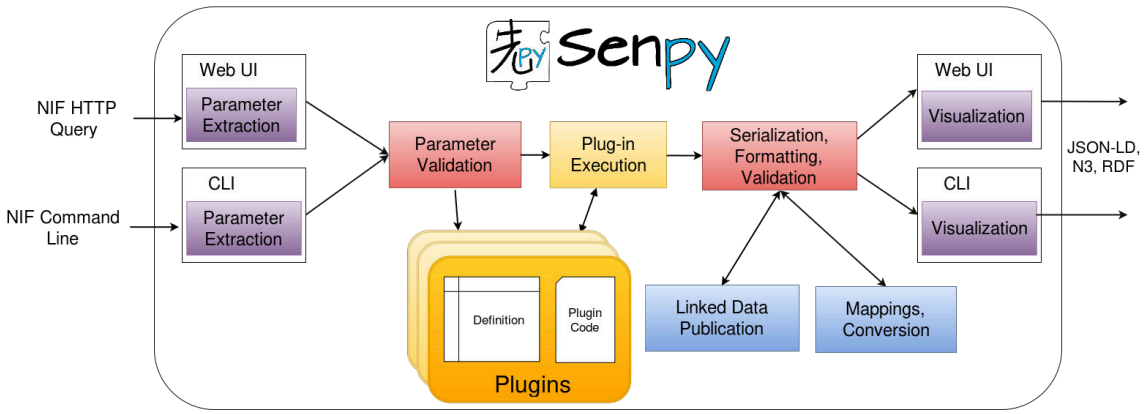


Figure 4.4: Architecture of Senpy's reference implementation.

4.2.5 Objective 5: Modelling the types of contextual information and social theories

The last part of the thesis was dedicated to studying the role of contextual information in sentiment and emotion analysis. After surveying the works in the area, it was evident that there was no consensus on terminology or methodology for this type of approach. This made it very hard to evaluate different works. Hence, we first proposed a formal definition of this type of contextual information, which we call social context. The definition includes the main entities (users and content), as well as the links between them (relations and interactions), and it can be modelled as a graph with different types of nodes and edges. The formal definition of context provides a common language to describe the information used in each work, which makes it easier to describe and understand the information used in each approach. There was still the issue that there are countless ways to gather and combine elements in social context. To address that, and to be able to characterize and compare different approaches, we proposed a methodology for comparison and a taxonomy of approaches based on the elements of social context used. The methodology includes different elements that may differ in each approach, and the taxonomy includes four main categories inspired the social sciences and economics: contextless, micro, meso and macro. The formal definition, methodology and comparison of different approaches was published as a journal paper **"Social context in sentiment analysis: Formal definition, overview of current trends and framework for comparison"** (Section 3.3.1), thus meeting **Objective-5**.

At a high level, the definition of social context is the following:

$$SocialContext = \langle C, U, R, I \rangle \tag{4.1}$$

Where: U is the set of content generated; C is the set of users; I is the set of interactions between users, and of users with content; R is the set of relations between users, between pieces of content, and between users and content. Users may interact (i) with other users (I^u), or with content (I^c). Relations (R) can link any two elements: two users (R^u), a user with content (R^{uc}), or two pieces of content (R^c).

This definition is illustrated in Figure 4.5, which provides a graphical representation of the possible links between entities of the two available types.

One of the novel aspects of the methodology for comparison of approaches is the introduction of different levels of analysis, based on the complexity or scope of their context. Our proposal is inspired by the micro, meso and macro levels of analysis typically used in social sciences Bolíbar, 2016. The two differences are: 1) a level of analysis is added to account for

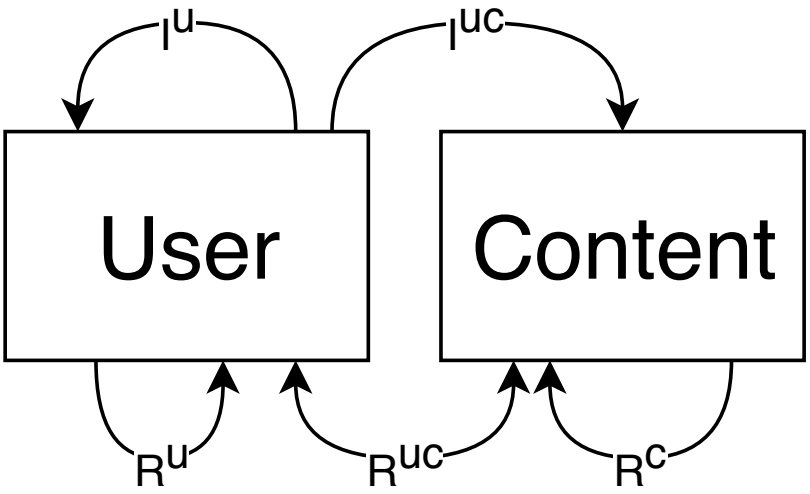


Figure 4.5: Model of Social Context, including: content (C), users (U), relations (R^c, R^u and R^{uc}), and interactions (I^u and I^{uc}).

analysis without social context, and 2) the meso level is further divided into three sub-levels ($meso_r$, $meso_i$, and $meso_e$), to better capture the nuances at the meso level. The result is shown in Fig. 4.6. The specific levels are the following:

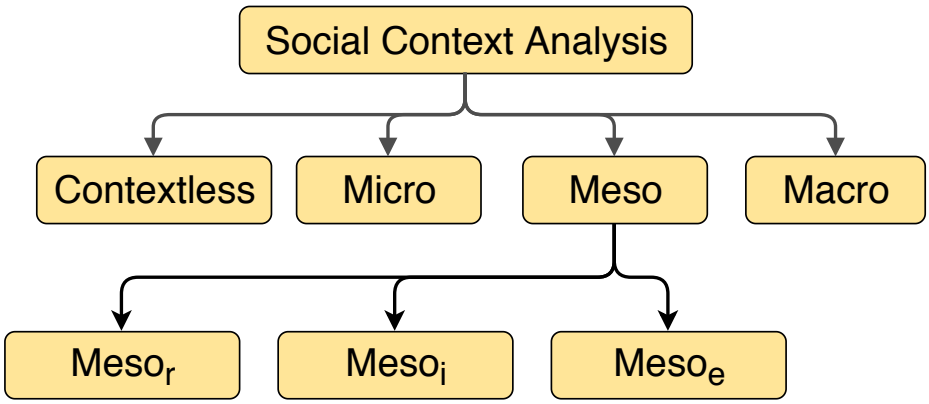


Figure 4.6: Taxonomy of approaches, and the elements of Social Context involved.

- Contextless: The approaches in this category do not use social context, and they rely solely on textual features.
- Micro: These approaches exploit the relation of content to its author(s), and may include other content by the same author. For instance, they may use the sentiment of previous posts (Aisopos et al., 2012) or other personal information such as gender and age to use a language model that better fits the user (Volkova, Wilson, and Yarowsky, 2013).

- Meso-relations ($Meso_r$): In this category, the elements from the micro category are used together with relations between users. This new information can be used to create a network of users. The slow-changing nature of relations makes the network very stable. The network can be used in two ways. First, to calculate user and content metrics, which can later be used as features in a classifier. e.g., a useful metric could be the ratio of positive neighboring users (Aisopos et al., 2012). Second, the network can be actively used in the classification, with approaches such as label propagation (Speriosu et al., 2011).

- Meso-interactions ($Meso_i$): This category also models and utilizes interactions. Interactions can be used in conjunction with relations to create a single network or be treated individually to obtain several independent networks. The resulting network is much richer than the previous category, but also subject to change. In contrast to relations, interactions are more varied and numerous. To prevent interactions from becoming noisy, they are typically filtered. For instance, two users may only be connected only when there have been a certain number of interactions between them.

- Meso-enriched ($Meso_e$): A natural step further from $Meso_i$, this category uses additional information inferred from the social network. A common technique in this area is community detection. Community partitions may inform a classifier, influence the features used for each instance (Tommasel and Godoy, 2018), or be used to process groups of users differently (Deitrick and W. Hu, 2013). Other examples would be metrics such as modularity and betweenness, which can be thought of as proxies for importance or influence. Some works have successfully explored the relationship between these metrics and user behavior, in order to model users. However, these results are seldom used in classification tasks.

- Macro: At this level, information from other sources outside the social network is incorporated. For instance, Li et al., 2012 use public opposition of political candidates in combination with social theories to improve sentiment classification. Another example of external information is facts such as the population of a country, or current government, which can be combined with geo-location information in social media content. A more complex example would be events in the real world or in other types of media, such as television, which can be analyzed in combination with social media activity (Heo et al., 2016).

The six levels of approaches are listed in increasing order of detail, measured as the number of elements social context may include.

Although the analysis of the state of the art revealed a tendency towards more complex models (i.e., $meso_e$), there are still few models that fully exploit SNA for sentiment analysis. At the same time, we were interested in incorporating community detection, which could help find weak relationships between users who are otherwise not connected in the network. This motivated us to work on a novel model for sentiment classification that exploits a combination of social context and social network analysis. In addition to including community detection, our model is also particular in that it can classify both users and content. The results have been published in an Open Access paper titled **"CRANK: A Hybrid Model for User and Content Sentiment Classification Using Social Context and Community Detection"** (Section 3.3.2).

Lastly, in order to investigate the effect of social theories in social context, and to test different classification algorithms, we developed an agent-based social simulator (Soil[3]), described in **"Soil: An Agent-Based Social Simulator in Python for Modelling and Simulation of Social Networks"** (Section 3.3.5). We also developed several agent models for the simulator, including different propagation behaviors. These behaviors apply to propagation phenomena such as rumor spreading, and emotion contagion. The simulator has been used in several works **"Modeling Social Influence in Social Networks with SOIL, a Python Agent-Based Social Simulator"** (Section 3.3.4), in diverse topics such as radicalization (Méndez et al., 2018; Méndez et al., 2019). It should also be very useful in future research on social context, either in the generation of synthetic datasets, or in the evaluation of new models.

Table 4.6: Publications related to Objective 5

Page	Title	Impact	Contribution
154	Social context in sentiment analysis: Formal definition, overview of current trends and framework for comparison	Q1 (10.716)	Formalism for the study of Social Context, including a definition of Social Context, a methodology to compare different approaches, and a survey of the state of the art using that methodology. (**Objective-5**)
238	Soil: An Agent-Based Social Simulator in Python for Modelling and Simulation of Social Networks	CORE-C	An agent-based social simulator that can be used to study the application of social theories (e.g., emotion contagion) on sentiment analysis. Several agent behaviors have been included, such as the SISa propagation model.
232	Modeling Social Influence in Social Networks with SOIL, a Python Agent-Based Social Simulator		A demo of Soil in a specific scenario.
215	A Model of Radicalization Growth using Agent-based Social Simulation	CORE-B	Application of Soil to study the outcome of different conditions on radical behavior. New agent behaviors for radicalists and radicalist cells were introduced.

[3] https://github.com/gsi-upm/soil

Table 4.6: Publications related to Objective 5

-	Analyzing Radicalism Spread Using Agent-Based Social Simulation	Book chapter	An extended version of the previous paper, in the form of a book chapter.
193	CRANK: A Hybrid Model for User and Content Sentiment Classification Using Social Context and Community Detection	Q2 (2.217)	A novel model for user and content classification using social context and SNA

4.2.6 Other

Lastly, there have been several contributions that are also aligned with the main lines of the thesis, although they do not directly contribute to the main objectives. These publications are in their majority focused on new sentiment or emotion classifiers, where we have collaborated with other researchers to evaluate the combination of deep learning with shallow (traditional) learning. We have also worked on a modular architecture to extract, analyze and visualizate information from social media and big data sources. The main part of the architecture is a visualization toolkit that can display information from semantic (SPARQL, JSON-LD) and non-semantic sources (elasticsearch). The toolkit includes several components to browse content annotated with sentiment and emotion. We have also collaborated to integrate emotion annotation and emotion analysis with a task automation service, geared towards making smart offices emotion-aware and more pleasant.

Table 4.7: Publications not directly related to the main objectives

Page	Title	Impact	Contribution
322	An Emotion Aware Task Automation Architecture Based on Semantic Technologies for Smart Offices	Q1 (3.031)	The use of Onyx in a task automation architecture to model the emotions of workers in a smart office.
343	Enhancing Deep Learning Sentiment Analysis with Ensemble Techniques in Social Applications	Q1 (2.981)	Taxonomy of "shallow" and deep learning approaches, and different combinations through ensemble techniques
363	Aspect based Sentiment Analysis of Spanish Tweets		Detection of sentiments on specific aspects, using traditional techniques.
356	Applying Recurrent Neural Networks to Sentiment Analysis of Spanish Tweets		Application of deep learning for sentiment analysis in Spanish.
303	A Big Linked Data Toolkit for Social Media Analysis and Visualization based on W3C Web Components		A modular architecture with a visualization toolkit, which has been used extensively to extract information, analyze it with Senpy services and show the results.

Table 4.7: Publications not directly related to the main objectives

379	EuroLoveMap: Confronting feelings from News	Visualization of the sentiment of European news sources on different topics, per country.

4.3 Applications

This thesis has contributed to the field of sentiment analysis in notable ways that are not covered by the scientific results in the previous section. The list of contributions includes the application of the scientific results in both industry and academia, the promotion of concepts through academic activities, and the development of open source tools that can be of value to the community.

- **The Linked Data Models for Sentiment and Emotion Analysis W3C Community group**. According to its website, the Sentiment Analysis Community Group[4] is a forum to promote sentiment analysis research. It addresses the following topics:

  - Definition of a Linked Data based vocabulary for emotion and sentiment analysis.
  - Requirements beyond text-based analysis, i.e. emotion/sentiment analysis from images, video, social network analysis, etc.
  - Clarifying requirements and the need for consensus as e.g. systems currently use widely varying features for describing polarity values (1-5, -2/-1/0/1/2, positive/neutral/negative, good/very good etc.).
  - Marl and Onyx are vocabularies for emotion and sentiment analysis that can be taken as a starting point for discussion in the CG.

  As any other W3C Community Group, this group cannot publish specifications. Nonetheless, it is expected to publish recommendations and reports on the activity of the group. The group was created by several partners of the EuroSentiment project, who identified the need for the standardizaitton of vocabularies, tools and practices in the field.

  As co-chair of the group, I collaborated in planning the meetings, gathering information for the community group, coordinating the publication of a conference paper, **"Towards a Common Linked Data Model for Sentiment and Emotion Analysis"** (Section 3.1.3), and publishing a recommendation. This recommendation,

[4] `https://www.w3.org/community/sentiment/`

entitled "Guidelines for developing Linked Data Emotion and Sentiment Analysis services"[5], heavily uses the model defined in this thesis. More specifically, it uses the same schema, and the vocabularies Marl and Onyx to represent sentiment and emotion, respectively.

- **Workshop on Emotion and Sentiment Analysis**. The W3C Community Group was also involved in hosting the 2016 edition of the Emotion and Sentiment Analysis workshop, co-located with the LREC conference in Portoroz, Slovenia. As its predecessors, the aim of this workshop was to connect the related fields around sentiment, emotion and social signals, exploring the state of the art in applications and resources. All this, with a special interest on multidisciplinarity, multilingualism and multimodality. The organizing committee comprised several members of the group:

  - J. Fernando Sánchez-Rada - UPM, Spain
  - Carlos A. Iglesias - UPM, Spain
  - Björn Schuller - Imperial College London, UK
  - Gabriela Vulcu - Insight Centre for Data Analytics, NUIG, Ireland
  - Paul Buitelaar - Insight Centre for Data Analytics, NUIG, Ireland
  - Laurence Devillers - LIMSI, France

  The workshop was an opportunity to promote the results from this thesis, as well as the work in the Community Group.

- The **EuroSentiment project** aimed to develop a large shared data pool for language resources meant to be used by sentiment analysis systems, in order to bundle together scattered resources. One goal was to extend the WordNet Domain to sentiment analysis. The project specified a schema for sentiment analysis and normalise the metrics used for sentiment strength. This schema for language resources was a direct result of this thesis (**Objective-2**), and it included Onyx as the vocabulary for emotions (**Objective-1**). The sharing of resources would be supported by a self-sustainable and profitable framework based on a community governance model, offering contributors the possibility of exploiting commercially the resources they provide.

- **MixedEmotions project** (Grant Agreement no: 141111) aimed to develop innovative multilingual multi-modal Big Data analytics applications that will analyze a more complete emotional profile of user behavior using data from mixed input channels: multilingual text data sources, A/V signal input (multilingual speech, audio, video), social

[5] `https://www.gsi.upm.es/otros/ldmesa/`

media (social network, comments), and structured data. Its commercial applications (implemented as pilot projects) would be in Social TV, Brand Reputation Management and Call Centre Operations. Making sense of accumulated user interaction from different data sources, modalities and languages is challenging and has not yet been explored in fullness in an industrial context. In other words, EuroSentiment focused on creating and sharing multi-lingual resources for sentiment and emotion analysis, and MixedEmotions built on that expertise to develop a multi-lingual multi-modal platform with different analysis services that can be used and combined in different scenarios. MixedEmotions made extensive use of the API and vocabularies defined for Senpy (including Onyx), as well as its reference implementation. Namely, all the NLP services in the platform used the common API, and several services were developed using the reference implementation, either natively (python code) or in the form of a wrapper (plugin that interacts with an external service).

- In addition to EUROSENTIMENT and MixedEmotions, Senpy has been used in more than 5 projects at European and national level. The list includes:

  - **TRIVALENT**. TRIVALENT is an EU funded project which aims to a better understanding of root causes of the phenomenon of violent radicalisation in Europe in order to develop appropriate countermeasures, ranging from early detection methodologies to techniques of counter-narrative. Several ad-hoc services to detect radicalism in text were produced, which were integrated with the existing emotion analysis services, thanks to the common API.

  - **EMOSPACES**. EmoSpaces' goal is the development of an IoT platform that determines context awareness with a focus on sentiment and emotion recognition and ambient adaptation. In this platform, Onyx is integrated with EWE, the Evented WEb ontology Coronado, Carlos A Iglesias, and Serrano, 2015. This combination effectively achieves emotion-aware task automation. In other words, different aspects of the environment can be automated depending on the emotion of the occupants. Moreover, several senpy plugins for emotion analysis were used for sentiment and emotion analysis in text.

  - **SoMeDi**. SOMEDI tries to solve the challenge of efficiently generating and utilising social media and digital interaction data enabled intelligence. To do so, several NLP and machine learning services were used, on large datasets of content from social media. The Senpy API was used in the NLP services of this Big Data architecture to achieve service interoperability, and the reference implementation was used to adapt services from different partners (TAIGER, HI Iberia).

- **SEMOLA**. The SEMOLA project aims to research on models, techniques and tools for the development of the empathetic personal agents endowed with a model of emotions. This facilitates the management of users' relations with an intelligent social environment consisting of sensors and ambient intelligence. To this end, the project aims to: i) investigate recognition techniques and spread models for sentiments and emotions in social networks and smart environments (for that purpose, semantic web and linked data technologies, natural language processing, and social simulation will be employed) ; ii) research on customization services based on the context given by ambient intelligence devices in smart environments; and iii) research on agents models capable of generating emotions with a conversational interface, and to serve as a decision support system to assist users of an intelligent environment in their daily lives. The SEMOLA project exploits results from the three main areas of the thesis: it uses Onyx for emotion representation, Senpy for sentiment analysis in text, and the study of the impact of social theories in emotion-related phenomena in social media such as emotion propagation.
- **Financial Twitter Tracker** The main objective of Financial Twitter Tracker (FTT) is to enrich financial content with information from Online Social media such as Twitter. FTT used Senpy for sentiment analysis of financial-related tweets.

- **Use by industry and community**. Senpy has been used by industry partners such as: HI Iberia, Expert System, Paradigma Tecnológico, Taiger, and emotion-research. The main reason these companies have used Senpy to enable service interoperability in environments with several providers, or to publish their own services to third parties. In addition to that, Senpy is popular in the open source community. As of this writing, its GitHub repository[6] has 19 forks and 63 stars, the main docker image on Docker Hub has been downloaded over 5 thousand times, and the PyPI repository averages more than 200 downloads per month.

- **Creation of Open Source plugins for commercial applications**[7].
  - MeaningCloud[8] (Sentiment). MeaningCloud market-leading solutions for text mining and voice of the customer. The meaningcloud plugin provides a wrapper over their commercial service that exposes the Senpy API. The plugin has been used several times to compare novel approaches to the state of the art.

[6] https://github.com/gsi-upm/senpy
[7] https://github.com/gsi-upm/senpy-plugins-community
[8] https://www.meaningcloud.com/

- Sentiment140 (Go, Bhayani, and Huang, 2009) (Sentiment). Sentiment140 is a public service for sentiment analysis in Twitter. The original work the service is based on has been citepd over 2 thousand times, and the service has been used as baseline in multiple works.
- TAIGER (Sentiment). This is a wrapper for two different commercial sentiment analysis services in the company.
- Vader (Sentiment). A sentiment analysis service based on Vader (Hutto and Gilbert, 2014), a rule-based model for sentiment analysis of social media text.
- DepecheMood (Emotion). DepecheMood is a lexicon for emotion analysis based on crowd-sourced annotations of news (Staiano and Guerini, 2014). This plugin analyses emotions using that lexicon.
- WordNet-Affect (Emotion). WordNet-Affect (Strapparava, Valitutti, et al., 2004) is an extension of WordNet Domains, including a subset of synsets suitable to represent affective concepts correlated with affective words. This plugin maps sentences to WordNet synsets, and then uses WordNet-Affect to compute the global emotion of the text.
- ANEW (Emotion). ANEW (Bradley and Lang, 1999) provides a set of normative emotional ratings for a large number of words in the English language. This plugin uses ANEW in an emotion classifier that detects six possible emotions: anger, fear, disgust, joy, sadness and a neutral emotion.
- Emotion-Research (Emotion in Video). Emotion Research[9] provides emotion analysis of video. This plugin is a wrapper over that service.

4.4 Conclusions

In our first group of contributions, we provided a common set of vocabularies and schemata for both services and languages resources. Our rationale was that these vocabularies would allow these communities to cooperate, and eventually provide hybrid solutions. A major barrier to the creation of a vocabulary of emotions has been the lack of a consensus on what models of emotion to use. We addressed this issue by providing a vocabulary with a meta-model for emotions, where users can define their own specific models, and by separately providing a wide variety of models from which to choose. The combination of this emotion vocabulary with other vocabularies such as lemon has allowed us to successfully model language resources and services alike. This has been supported by our experiences on two

[9] https://emotionresearchlab.com/

European projects, several bachelor and master theses, and other uses that we have detected in the community. In particular, the EuroSentiment language resource pool was the main goal of the project and it was made possible thanks to the model for language resources. The portal is covered in **"EUROSENTIMENT: Linked Data Sentiment Analysis"** (Section 3.1.5). The use of a semantic model in that are has proven to be very advantageous. For instance, the work on automatic domain-specific lexicons from lexical resources exemplifies the benefits of semantically generated resources, as covered in **"Generating Linked-Data based Domain-Specific Sentiment Lexicons from Legacy Language and Semantic Resources"** (Section 3.1.8). This automatic resource translation and adaptation would not be possible without the semantic model and its connection to the WordNet ecosystem.

Our second group of contributions is on linked data for sentiment analysis services. Extending NIF to include affects and creating a schema for service results has yielded very positive results. The schema has been used extensively by different external partners since its inception, and the reference implementation has been used both internally (bachelor, master, PhD theses and other use cases), as well as by third parties (e.g., MixedEmotions project). Choosing to default to a more user-friendly format (JSON-LD) has also been a success. It has empowered developers and researchers from different fields to contribute their own services and consume those of others. They could do so without prior exposure to semantic technologies. At the same time, the interoperability of services and several of the features advertised by the reference implementation, such as automatic model conversion, were powered by semantic technologies. On a related note, our collaboration on multi-modal fusion has been particularly encouraging, as it combines four advantages of the semantic definition of resources and services. Firstly, several semantically annotated languages resources were used in text classification. Secondly, different services were consumed using the same APIs. Thirdly, the combination of the results from each modality was achieved thanks to semantic interoperability. Lastly, once the results were combined, the fusion was achieved through automatic model conversion.

The creation of the W3C Community Group on Linked Data Models for Sentiment and Emotion Analysis has been highly positive for the promotion of linked data and semantic values in the NLP community. Among other things, it has led to several online and offline activities to discuss the use of linked data technologies in the area. The group has also gathered their knowledge on tools, resources and projects that are relevant for other researchers. The result can be seen in both the website of the group, and in the joint publication that served as a declaration of intentions of the group.

The reference implementation has been integrated with popular libraries such as scikit-

learn. The integration is bidirectional: converting a scikit-learn classifier into a Senpy service is straightforward, as is using a senpy plugin as a scikit-learn classifier. This is very convenient for developers, especially in combination with the automatic evaluation of plugins.

Lastly, social context seems like a natural step further in sentiment and emotion analysis research. Our analysis of the state of the art from the past decade confirmed that new approaches that exploit social context outperform contextless baselines. The same analyses revealed a lack of unified terminology in the field. This was expected, as is both a new field and an interdisciplinary one. To remedy that, we provided a formal definition of social context that could be applied to all the works in our analysis, as well as a methodology to compare all their approaches. We hope that our proposal helps to clarify the different approaches available both now and in the future, and that it will contribute to foster research in the area. Unfortunately, there are other barriers other than terminology that is impeding research on social context. The main one is that the use of social context requires more information to be available to the researcher, such as a link between the piece of content and its author, and the connection of that author to other members of the network. This information is very seldom available in public datasets. In some situations it can be gathered a posteriori, provided the piece of content is uniquely identified and searchable (i.e., there is a user ID). But most datasets contain solely the piece of text and its labels, so there is no way to gather the context. Sometimes, IDs are not present because the datasets were created for pure text classification and the use of social context had not been anticipated. Some other times, the creators of a dataset knowingly omit IDs or other traceable features to avoid legal and ethical problems. Further discussion is necessary to circumvent these types of issues and obtain more complete datasets.

In conclusion, all our hypotheses have been supported:

Hypothesis-1 "A Linked Data approach would increase interoperability between services and enable advanced capabilities such as automatic evaluation" has been supported overall by the overall results of the MixedEmotions project, which relied heavily on interoperability between services. The results of that project are described in **"MixedEmotions: An Open-Source Toolbox for Multi-Modal Emotion Analysis"** (Section 3.2.3). It is also supported by the fact that different types of developers, both researchers and non-researchers, and with different types of backgrounds, have successfully used the reference implementation of Senpy, and some of those uses were done in the context of international publications, whose main contribution was unrelated to Senpy.

Hypothesis-2 "A semantic vocabulary for emotions would ease multi-modal analysis

and enable using different emotion models" has been supported by our collaborations on multi-modal analysis, multi-modal fusion, and the implementation of automatic model conversion, which has been used by different partners. This is illustrated in **"A Linked Data Model for Multimodal Sentiment and Emotion Analysis"** (Section 3.1.4) and **"Multimodal Multimodel Emotion Analysis as Linked Data"** (Section 3.2.2).

Hypothesis-3 "Sentiment of social media text can be predicted using additional contextual information (e.g., previous history and relations between users)" has been supported by our analysis of the state of the art in the field in the paper **"Social context in sentiment analysis: Formal definition, overview of current trends and framework for comparison"** (Section 3.3.1), which showed an improvement in context-based approaches, as opposed to contextless approaches, albeit with higher variability. Our further investigation in the matter (yet to be published) also supports the idea that models that use community detection (a form of SNA) perform better than both contextless and other types of context-based approaches (i.e., *micro* and *meso*).

4.5 Future Research

Research is an ongoing task. During the course of this thesis, we identified several ways in which our research could be either expanded or continued in different directions.

First of all, the model for emotion conversion in Onyx has shown potential for interoperability and multi-modal fusion, but it has not been fully exploited. Currently, only two types of bi-directional conversions have been defined: Ekman's to VAD and Ekman's to FSRE. Both of these conversions rely on the same implementation of a centroid-based conversion. In the VAD case, the centroids are calculated using well known lexicons for VAD, and WordNet-Affect labels for the same words.

Although the application of this conversion with multi-modal ensemble has shown good results, more conversion strategies should be implemented and properly evaluated. Perhaps a conversion from WordNet-Affect A-labels to VAD would be a good candidate to start. The fact that WordNet-Affect labels form a taxonomy could also be exploited. The conversion could be parameterized to choose the desired level of the WordNet-Affect taxonomy. That would allow end users to control the level of emotion granularity. The level may be automatically selected depending on the confidence of the algorithm in the conversion.

On the modelling of language resources and services, there are several ways to continue improving. The recommendation by the W3C Community Group has been a great step towards a common model in the community, but it is still far from being a community

standard. On the other hand, it remains to be seen whether the community feels the need for an actual standard. Throughout this document we have outlined the drawbacks of ad-hoc implementations, and the benefits of standardizing the APIs, tooling and terminology in the field. Hence, we believe that a standard will be sought after as the community grows. In the meantime, we should continue working on capturing the use cases for linked data for sentiment analysis, and exploiting the capabilities of semantics.

Regarding the use of linked data in sentiment and emotion analysis services, the Senpy architecture is very complete, and there are several aspects that are not fully utilized in the current implementation. For instance, the evaluation layer provides limited reporting capabilities, which are mostly limited to accuracy and F-1 score evaluation (and cross-evaluation) metrics. Although some results are cached, the metrics are provided on demand. It would be interesting to expose these results and the set of gold standards as linked data to other instances.

Another point for consideration is to extend the architecture, which was initially envisioned as a monolith for relatively light and isolated services. As more and more plugins were developed for the reference implementation, keeping all the independent versions up to date became rather tedious. Releasing a new version of the core requires re-launching each server. In some cases, the server needs to either train a classifier or load data into memory. Re-training can be avoided by several mechanisms we provide to persist and load data. However, the start-up time of several services is still long. A more ambitious extension of the architecture could account for distributed analysis systems, where different parts of the architecture are provided by different servers. In such an architecture, the modules responsible for the analysis could be lightweight microservices that implement a remote interface and communicate with a central module that provides the main features. The central module could allow for automatic registration of analysis modules, with optional authentication. This would be transparent to a user of the service, but it would allow us to decouple the development of services from the .

Regarding social context, our work is the first to formalize the concept of social context, and to propose a methodology to compare different approaches. We have done so by analyzing the state of the art, and by borrowing concepts from social sciences. Unfortunately, the field is rather new, which means the taxonomy of approaches could not be too detailed. As more and more works in the field get published, the taxonomy could be refined. Likewise, additional useful abstractions could be incorporated in the definition of social context, to broaden the vocabulary and to help characterize different scenarios and approaches. All this new knowledge could be crystallized in the form of an ontology of social context. New works

could employ that ontology to define their approaches and their datasets. Moreover, the ontology could be used as input to context-aware analysis services, which take some form of social context as input to their algorithm.

One of the main obstacles to social context research is the lack of available datasets. More datasets would enable the creation of more precise models, and further evaluation of existing approaches. The lack can be partially compensated in some scenarios with synthetic datasets. These datasets could be generated using agent-based social simulation, with tools such as Soil.

Lastly, there is an interesting trend in deep learning that consists in capturing a representation of part of the social context information into fixed-length vectors. Those vectors are later used as features in a neural network. This technique is known as embedding, and it is very common for text features (e.g., Word2Vec, GloVe). The rationale behind embedding is that the vectors summarize latent semantic relations between words. The same principle could apply to social context features, including the graph of relations or interactions. In fact, some works are already working on producing network embeddings. It would be very interesting to combine the insights gathered from our study of social context with the power of both type of embeddings. One limitation of such an approach is once again the lack of datasets that either contain or can be extended with social context features. Embedding relies on high volumes of data in order to converge to quality vectors.

Bibliography

Aisopos, Fotis et al. (2012). "Content vs. context for sentiment analysis". In: *Proceedings of the 23rd ACM conference on Hypertext and social media - HT '12*. New York, New York, USA: ACM Press, p. 187. ISBN: 978-1-4503-1335-3. DOI: 10.1145/2309996.2310028. URL: http://dl.acm.org/citation.cfm?doid=2309996.2310028.

Araque, Oscar et al. (June 2017). "Enhancing Deep Learning Sentiment Analysis with Ensemble Techniques in Social Applications". In: *Expert Systems with Applications*. ISSN: 0957-4174. DOI: http://dx.doi.org/10.1016/j.eswa.2017.02.002. URL: http://www.sciencedirect.com/science/article/pii/S0957417417300751.

Ashimura, Kazuyuki et al. (May 2012). *EmotionML vocabularies*. Tech. rep. W3C. URL: http://www.w3.org/TR/2012/NOTE-emotion-voc-20120510/.

Barsade, Sigal G. (2002). "The Ripple Effect: Emotional Contagion and its Influence on Group Behavior". In: *Administrative Science Quarterly* 47.4, pp. 644–675.

Bengio, Yoshua (Nov. 2009). "Learning Deep Architectures for AI". English. In: *Foundations and Trends® in Machine Learning* 2.1, pp. 1–127. ISSN: 1935-8237, 1935-8245.

Bizer, Christian, Tom Heath, and Tim Berners-Lee (2009). "Linked data-the story so far". In: *Semantic Services, Interoperability and Web Applications: Emerging Concepts*, pp. 205–227. DOI: doi:10.4018/jswis.2009081901.

Bizer, Christian, Jens Lehmann, et al. (2009). "DBpedia-A crystallization point for the Web of Data". In: *Web Semantics: science, services and agents on the world wide web* 7.3, pp. 154–165.

Bolíbar, Mireia (Sept. 2016). "Macro, meso, micro: broadening the 'social' of social network analysis with a mixed methods approach". en. In: *Quality & Quantity* 50.5, pp. 2217–2236. ISSN: 0033-5177, 1573-7845.

Bontcheva, Kalina and Dominic Rout (2012). "Making sense of social media streams through semantics: a survey". In: *Semantic Web* 1, pp. 1–31. DOI: 10.3233/SW-130110.

Borth, Damian et al. (2013). "SentiBank: large-scale ontology and classifiers for detecting sentiment and emotions in visual content". In: *Proceedings of the 21st ACM international conference on Multimedia*. MM '13. Barcelona, Spain: ACM, pp. 459–460. ISBN: 978-1-4503-2404-5. URL: http://doi.acm.org/10.1145/2502081.2502268.

Bradley, Margaret M and Peter J Lang (1999). *Affective norms for English words (ANEW): Instruction manual and affective ratings*. Tech. rep. Technical Report C-1, The Center for Research in Psychophysiology, University of Florida.

Buitelaar, Paul, Mihael Arcan, et al. (Sept. 2013). "Linguistic Linked Data for Sentiment Analysis". In: *2nd Workshop on Linked Data in Linguistics (LDL-2013): Representing and linking lexicons, terminologies and other language data. Collocated with the Conference on Generative Approaches to the Lexicon*. Ed. by Christian Chiarcos et al. Pisa, Italy: Association for Computational Linguistics, pp. 1–8. ISBN: 978-1-937284-77-0. URL: https://www.aclweb.org/anthology/W/W13/W13-55.pdf.

Buitelaar, Paul, Philipp Cimiano, et al. (2011). "Ontology lexicalisation: The lemon perspective". In: *Workshop at 9th International Conference on Terminology and Artificial Intelligence (TIA 2011)*, pp. 33–36.

Buitelaar, Paul, Ian D. Wood, et al. (Oct. 2018). "MixedEmotions: An Open-Source Toolbox for Multi-Modal Emotion Analysis". In: *IEEE Transactions on Multimedia*. ISSN: 1520-9210. URL: http://ieeexplore.ieee.org/document/8269329/.

Burkhardt, F. et al. (2013). *W3C Emotion Markup Language (EmotionML) 1.0 proposed recommendation*.

Cambria, Erik (2016). "Affective computing and sentiment analysis". In: *IEEE Intelligent Systems* 31.2, pp. 102–107.

Cambria, Erik, Catherine Havasi, and Amir Hussain (2012). "SenticNet 2: A Semantic and Affective Resource for Opinion Mining and Sentiment Analysis." In: *FLAIRS Conference*, pp. 202–207.

Cambria, Erik, Andrew Livingstone, and Amir Hussain (2012). "The hourglass of emotions". In: *Cognitive Behavioural Systems*. Springer, pp. 144–157.

Chiarcos, Christian (2012a). "Ontologies of Linguistic Annotation: Survey and perspectives." In: *LREC*, pp. 303–310.

— (2012b). "POWLA: Modeling linguistic corpora in OWL/DL". In: *The Semantic Web: Research and Applications*. Springer, pp. 225–239.

— (2013). "Linguistic Linked Open Data (LLOD)–Building the cloud". In: *Joint Workshop on NLP&LOD and SWAIE: Semantic Web, Linked Open Data and Information Extraction*, p. 1.

Chiarcos, Christian, Sebastian Hellmann, and Sebastian Nordhoff (2011a). "Towards a Linguistic Linked Open Data cloud: The Open Linguistics Working Group". In: *TAL* 52.3, pp. 245–275.

— (2011b). "Towards a linguistic linked open data cloud: The Open Linguistics Working Group". In: *TAL* 52.3, pp. 245–275.

— (2012). “Linking linguistic resources: Examples from the open linguistics working group”. In: *Linked Data in Linguistics*. Springer, pp. 201–216.

Chiarcos, Christian, John McCrae, et al. (2013). “Towards open data for linguistics: Linguistic linked data”. In: *New Trends of Research in Ontologies and Lexical Resources*. Springer, pp. 7–25.

Collobert, Ronan et al. (2011). “Natural language processing (almost) from scratch”. In: *The Journal of Machine Learning Research* 12, pp. 2493–2537. URL: http://dl.acm.org/citation.cfm?id=2078186 (visited on 04/15/2015).

Coronado, Miguel, Carlos A Iglesias, and Emilio Serrano (2015). “Modelling rules for automating the Evented WEb by semantic technologies”. In: *Expert Systems with Applications* 42.21, pp. 7979–7990.

Decker, S. et al. (Sept. 2000). “The Semantic Web: the roles of XML and RDF”. In: *IEEE Internet Computing* 4.5. 01101, pp. 63–73. ISSN: 1941-0131.

Deitrick, William and Wei Hu (2013). “Mutually Enhancing Community Detection and Sentiment Analysis on Twitter Networks”. In: *Journal of Data Analysis and Information Processing* 01.3, pp. 19–29. ISSN: 2327-7211, 2327-7203. DOI: 10.4236/jdaip.2013.13004. URL: http://www.scirp.org/journal/PaperDownload.aspx?DOI=10.4236/jdaip.2013.13004 (visited on 06/24/2016).

Ekman, Paul (1999). “Basic emotions”. In: *Handbook of cognition and emotion* 98, pp. 45–60.

Ekman, Paul and Wallace V Friesen (1971). “Constants across cultures in the face and emotion.” In: *J. of personality and social psychology* 17.2, p. 124.

Excellence, HUMAINE Network of (June 2006). *HUMAINE Emotion Annotation and Representation Language (EARL): Proposal*. Tech. rep. HUMAINE Network of Excellence. URL: http://emotion-research.net/projects/humaine/earl/proposal#Dialects.

Fellbaum, Christiane (1998). *WordNet*. Wiley Online Library.

Fontaine, Johnny R. J. et al. (2007). “The World of Emotions is not Two-Dimensional”. en. In: *Psychological Science* 18.12, pp. 1050–1057. ISSN: 0956-7976, 1467-9280. DOI: 10.1111/j.1467-9280.2007.02024.x.

Gao, Bo et al. (2012). “Interactive grouping of friends in OSN: Towards online context management”. In: *Data Mining Workshops (ICDMW), 2012 IEEE 12th International Conference on*. IEEE, pp. 555–562.

García-Pablos, Aitor, Montse Cuadros Oller, and German Rigau Claramunt (2016). “A comparison of domain-based word polarity estimation using different word embeddings”. eng. In: *Proceedings of the Tenth International Conference on Language Resources and Evaluation*. Portoroz, Slovenia.

Gimpel, Kevin et al. (2011). "Part-of-speech Tagging for Twitter: Annotation, Features, and Experiments". In: *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers - Volume 2*. HLT '11. Stroudsburg, PA, USA: Association for Computational Linguistics, pp. 42–47. ISBN: 978-1-932432-88-6.

Go, Alec, Richa Bhayani, and Lei Huang (2009). "Twitter sentiment classification using distant supervision". In: *CS224N Project Report, Stanford* 1, p. 12.

Grassi, Marco (2009). "Developing HEO human emotions ontology". In: *Biometric ID Management and Multimodal Communication*. Springer, pp. 244–251. URL: http://link.springer.com/chapter/10.1007/978-3-642-04391-8_32 (visited on 01/21/2016).

Groth, Paul and Luc Moreau (2013). *Prov-O W3C Recommendation*. Tech. rep. W3C. URL: http://www.w3.org/TR/prov-o/.

Hajian, B. and T. White (Oct. 2011). "Modelling Influence in a Social Network: Metrics and Evaluation". In: *2011 IEEE Third International Conference on Privacy, Security, Risk and Trust and 2011 IEEE Third International Conference on Social Computing*, pp. 497–500.

Hastings, Janna et al. (2011). "Dispositions and Processes in the Emotion Ontology". In: *Proc. ICBO*, pp. 71–78.

Hellmann, Sebastian (2013). "Integrating Natural Language Processing (NLP) and Language Resources using Linked Data". PhD thesis. Universität Leipzig.

Hellmann, Sebastian et al. (2013). "Integrating nlp using linked data". In: *The Semantic Web–ISWC 2013*. Springer, pp. 98–113.

Heo, Yun-Cheol et al. (May 2016). "The emerging viewertariat in South Korea: The Seoul mayoral TV debate on Twitter, Facebook, and blogs". In: *Telematics and Informatics* 33.2. 00014, pp. 570–583. ISSN: 0736-5853.

Hogenboom, Alexander et al. (2015). "Exploiting Emoticons in Polarity Classification of Text." In: *J. Web Eng.* 14.1&2. 00043, pp. 22–40.

Hu, Xia et al. (2013). "Exploiting Social Relations for Sentiment Analysis in Microblogging". In: *Proceedings of the Sixth ACM International Conference on Web Search and Data Mining*. WSDM '13. New York, NY, USA: ACM, pp. 537–546. ISBN: 978-1-4503-1869-3. DOI: 10.1145/2433396.2433465. URL: http://doi.acm.org/10.1145/2433396.2433465 (visited on 03/10/2016).

Hutto, Clayton J and Eric Gilbert (2014). "Vader: A parsimonious rule-based model for sentiment analysis of social media text". In: *Eighth International AAAI Conference on Weblogs and Social Media*.

Ide, Nancy and Jean Veronis (1995). *Text encoding initiative: Background and contexts*. Vol. 29. Springer Science & Business Media. DOI: 10.1007/978-94-011-0325-1.

Jiang, Fei et al. (2015). "Microblog sentiment analysis with emoticon space model". In: *Journal of Computer Science and Technology* 30.5. 00026, pp. 1120–1129.

Kim, Yoon (Oct. 2014). "Convolutional Neural Networks for Sentence Classification". In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, pp. 1746–1751.

Kiritchenko, S., X. Zhu, and S. M. Mohammad (Aug. 2014). "Sentiment Analysis of Short Informal Texts". en-US. In: *Journal of Artificial Intelligence Research* 50, pp. 723–762. ISSN: 1076-9757.

Kramer, Adam DI, Jamie E. Guillory, and Jeffrey T. Hancock (2014). "Experimental evidence of massive-scale emotional contagion through social networks". In: *Proceedings of the National Academy of Sciences*, p. 201320040. URL: http://www.pnas.org/content/early/2014/05/29/1320040111.short (visited on 11/20/2014).

Li, Hao et al. (2012). "Combining Social Cognitive Theories with Linguistic Features for Multi-genre Sentiment Analysis." In: *PACLIC*, pp. 127–136.

Lipton, Zachary C. (2016). "The mythos of model interpretability". In: *arXiv preprint arXiv:1606.03490*.

Marcus, Gary (2018). "Deep learning: A critical appraisal". In: *arXiv preprint arXiv:1801.00631*.

McCrae, John, Dennis Spohr, and Philipp Cimiano (2011a). "Linking Lexical Resources and Ontologies on the Semantic Web with Lemon". In: *The Semantic Web: Research and Applications*. Ed. by Grigoris Antoniou et al. Vol. 6643. Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 245–259. ISBN: 978-3-642-21033-4.

— (2011b). "Linking lexical resources and ontologies on the semantic web with lemon". In: *Extended Semantic Web Conference*. 00210. Springer, pp. 245–259.

Melville, Prem, Wojciech Gryc, and Richard D. Lawrence (2009). "Sentiment Analysis of Blogs by Combining Lexical Knowledge with Text Classification". In: *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '09. New York, NY, USA: ACM, pp. 1275–1284. ISBN: 978-1-60558-495-9.

Méndez, Tasio et al. (July 2018). "A Model of Radicalization Growth using Agent-based Social Simulation". In: *Proceedings of EMAS 2018*. Stockholm, Sweden.

— (Aug. 2019). "Analyzing Radicalism Spread Using Agent-Based Social Simulation". In: *Engineering Multi-Agent Systems 6th International Workshop, EMAS 2018 Stockholm, Sweden, July 14–15, 2018 Revised Selected Papers*. Ed. by Alessandro Ricci Danny Weyns Viviana Mascardi. Vol. 11375. Lecture Notes in Artificial Intelligence. Springer International Publishing, pp. 263–285. ISBN: 0302-9743. URL: https://www.springer.com/gp/book/9783030256920.

Merino, Eduardo et al. (June 2017). "Modeling Social Influence in Social Networks with SOIL, a Python Agent-Based Social Simulator". In: *Advances in Practical Applications of Cyber-Physical Multi-Agent Systems: The PAAMS Collection*. Vol. 10349. Springer-Verlag, pp. 337–341. ISBN: 978-3-319-59929-8. URL: https://link.springer.com/chapter/10.1007/978-3-319-59930-4_33.

Mikolov, Tomas et al. (2013). "Efficient estimation of word representations in vector space". In: *arXiv preprint arXiv:1301.3781*.

Moreau, Luc et al. (2011). "The Open Provenance Model core specification (v1.1)". In: *Future Generation Computer Systems* 27.6, pp. 743–756. ISSN: 0167-739X. URL: http://www.sciencedirect.com/science/article/pii/S0167739X10001275.

Nasukawa, Tetsuya and Jeonghee Yi (2003). "Sentiment Analysis: Capturing Favorability Using Natural Language Processing". In: *Proceedings of the 2Nd International Conference on Knowledge Capture*. K-CAP '03. New York, NY, USA: ACM, pp. 70–77. ISBN: 978-1-58113-583-1.

Noro, Tomoya and Takehiro Tokuda (July 2016). "Searching for Relevant Tweets Based on Topic-related User Activities". In: *J. Web Eng.* 15.3-4, pp. 249–276. ISSN: 1540-9589. (Visited on 12/07/2018).

Novak, Petra Kralj et al. (2015). "Sentiment of emojis". In: *PloS one* 10.12. 00226, e0144296.

Orman, Günce Keziban, Vincent Labatut, and Hocine Cherifi (2011). "Qualitative comparison of community detection algorithms". In: *International conference on digital information and communication technology and its applications*. Springer, pp. 265–279.

Osgood, Charles Egerton, George J. Suci, and Percy H. Tannenbaum (1957). *The Measurement of Meaning*. en. Urbana, Illinois, USA: University of Illinois Press. ISBN: 978-0-252-74539-3.

Otte, Evelien and Ronald Rousseau (Dec. 2002). "Social network analysis: a powerful strategy, also for the information sciences". en. In: *Journal of Information Science* 28.6, pp. 441–453. ISSN: 0165-5515.

Pak, Alexander and Patrick Paroubek (2010). "Twitter as a corpus for sentiment analysis and opinion mining." In: *LREc*. Vol. 10, pp. 1320–1326.

Pamungkas, Endang Wahyu, Valerio Basile, and Viviana Patti (2019). "Stance Classification for Rumour Analysis in Twitter: Exploiting Affective Information and Conversation Structure". In: *CoRR* abs/1901.01911. URL: http://arxiv.org/abs/1901.01911 (visited on 12/04/2019).

Pang, Bo and Lillian Lee (2008a). "Opinion mining and sentiment analysis". In: *Foundations and trends in information retrieval* 2.1-2, pp. 1–135.

— (2008b). *Opinion mining and sentiment analysis*. Vol. 2. URL: http://dl.acm.org/citation.cfm?id=1454712 (visited on 11/20/2014).

Pang, Bo, Lillian Lee, and Shivakumar Vaithyanathan (2002). "Thumbs up?: sentiment classification using machine learning techniques". In: *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*. Association for Computational Linguistics, pp. 79–86. URL: http://dl.acm.org/citation.cfm?id=1118704 (visited on 03/16/2015).

Papadopoulos, Symeon et al. (2012). "Community detection in social media". In: *Data Mining and Knowledge Discovery* 24.3, pp. 515–554.

Phillips, Estelle, Derek Pugh, et al. (2010). *How to get a PhD: A handbook for students and their supervisors*. McGraw-Hill Education (UK).

Plutchik, Robert (1980a). "A general psychoevolutionary theory of emotion". In: *Theories of emotion* 1.3-31, p. 4.

— (1980b). *Emotion: A psychoevolutionary synthesis*. Harper & Row New York.

Polanyi, Livia and Annie Zaenen (2006). "Contextual valence shifters". In: *Computing attitude and affect in text: Theory and applications*. Springer, pp. 1–10.

Posner, Jonathan, James A Russell, and Bradley S Peterson (2005). "The circumplex model of affect: An integrative approach to affective neuroscience, cognitive development, and psychopathology". In: *Development and psychopathology* 17.03, pp. 715–734.

Pozzi, Federico Alberto et al. (2013). "Enhance User-Level Sentiment Analysis on Microblogs with Approval Relations". In: Springer International Publishing, pp. 133–144. URL: http://link.springer.com/10.1007/978-3-319-03524-6_12.

Prinz, Jesse J (2004). *Gut reactions: A perceptual theory of emotion*. Oxford University Press.

Ravi, Kumar and Vadlamani Ravi (Nov. 2015). "A survey on opinion mining and sentiment analysis: Tasks, approaches and applications". In: *Knowledge-Based Systems* 89.Supplement C, pp. 14–46. ISSN: 0950-7051.

Rokach, Lior (Feb. 2010). "Ensemble-based classifiers". en. In: *Artificial Intelligence Review* 33.1-2, pp. 1–39. ISSN: 0269-2821, 1573-7462.

Russell, James A (2003). "Core affect and the psychological construction of emotion." In: *Psychological review* 110.1, p. 145.

Saif, Hassan, Yulan He, and Harith Alani (2012). "Alleviating data sparsity for twitter sentiment analysis". In: *MSM2012 Workshop proceedings*, pp. 2–9.

Sánchez-Rada, J. Fernando, Oscar Araque, and Carlos A. Iglesias (Nov. 2019). "Senpy: A framework for semantic sentiment and emotion analysis services". In: *Knowledge-Based Systems*. URL: https://www.sciencedirect.com/science/article/pii/S0950705119305313.

Sánchez-Rada, J. Fernando and Carlos A. Iglesias (Dec. 2013). "Onyx: Describing Emotions on the Web of Data". In: *Proceedings of the First International Workshop on Emotion*

and Sentiment in Social and Expressive Media: approaches and perspectives from AI (ESSEM 2013). Vol. 1096. Torino, Italy: CEUR-WS, pp. 71–82.

Sánchez-Rada, J. Fernando and Carlos A. Iglesias (Jan. 2016). "Onyx: A Linked Data Approach to Emotion Representation". In: *Information Processing & Management* 52, pp. 99–114. ISSN: 0306-4573. URL: http://www.sciencedirect.com/science/article/pii/S030645731500045X.

— (Dec. 2019). "Social context in sentiment analysis: Formal definition, overview of current trends and framework for comparison". In: *Information Fusion* 52, pp. 344–356. ISSN: 1566-2535. URL: https://www.sciencedirect.com/science/article/pii/S1566253518308704.

— (Jan. 2020). "CRANK: A Hybrid Model for User and Content Sentiment Classification Using Social Context and Community Detection". en. In: *Applied Sciences* 10.5. 00000 Number: 5 Publisher: Multidisciplinary Digital Publishing Institute, p. 1662. DOI: 10.3390/app10051662. URL: https://www.mdpi.com/2076-3417/10/5/1662 (visited on 03/03/2020).

Sánchez-Rada, J. Fernando, Carlos A. Iglesias, Ignacio Corcuera-Platas, et al. (Oct. 2016). "Senpy: A Pragmatic Linked Sentiment Analysis Framework". In: *Proceedings DSAA 2016 Special Track on Emotion and Sentiment in Intelligent Systems and Big Social Data Analysis (SentISData)*. Montreal, Canada: IEEE, pp. 735–742. URL: http://ieeexplore.ieee.org/abstract/document/7796961/.

Sánchez-Rada, J. Fernando, Carlos A. Iglesias, and Ronald Gil (July 2015). "A Linked Data Model for Multimodal Sentiment and Emotion Analysis". In: Beijing, China: Association for Computational Linguistics and Asian Federation of Natural Language Processing, pp. 11–19. URL: http://aclweb.org/anthology/W/W15/W15-4202.pdf.

Sánchez-Rada, J. Fernando, Carlos A. Iglesias, Hesam Sagha, et al. (Oct. 2017). "Multimodal Multimodel Emotion Analysis as Linked Data". In: *Proceedings of ACII 2017*. San Antonio, Texas, USA.

Sánchez-Rada, J. Fernando, Björn Schuller, et al. (May 2016). "Towards a Common Linked Data Model for Sentiment and Emotion Analysis". In: *Proceedings of the LREC 2016 Workshop Emotion and Sentiment Analysis (ESA 2016)*. Ed. by J. Fernando Sánchez-Rada and Björn Schuller. LREC 2016, pp. 48–54. URL: http://www.lrec-conf.org/proceedings/lrec2016/workshops/LREC2016Workshop-ESA_Proceedings.pdf.

Sánchez-Rada, J. Fernando, Gabriela Vulcu, et al. (Oct. 2014). "EUROSENTIMENT: Linked Data Sentiment Analysis". In: *Proceedings of the ISWC 2014 Posters & Demonstrations Track a track within the 13th International Semantic Web Conference (ISWC 2014)*. Ed. by Jacco van Ossenbruggen Matthew Horridge Marco Rospocher. Vol. 1272. Riva

del Garda, Trentino: ISWC, pp. 145–148. URL: http://ceur-ws.org/Vol-1272/paper_116.pdf.

Sánchez, Jesús M., Carlos A. Iglesias, and J. Fernando Sánchez-Rada (June 2017). "Soil: An Agent-Based Social Simulator in Python for Modelling and Simulation of Social Networks". In: *Advances in Practical Applications of Cyber-Physical Multi-Agent Systems: The PAAMS Collection*. Vol. 10349. LNAI. Springer Verlag, pp. 234–245. ISBN: 978-3-319-59929-8. DOI: 10.1007/978-3-319-59930-4_19. URL: https://link.springer.com/chapter/10.1007/978-3-319-59930-4_19.

Schröder, Marc, Laurence Devillers, et al. (2007). "What should a generic emotion markup language be able to represent?" In: *Affective Computing and Intelligent Interaction*. Springer, pp. 440–451.

Schröder, Marc, Hannes Pirker, and Myriam Lamolle (2006). "First suggestions for an emotion annotation and representation language". In: *Proceedings of LREC*. Vol. 6, pp. 88–92.

Sharma, Anuj and Shubhamoy Dey (2012). "A comparative study of feature selection and machine learning techniques for sentiment analysis". In: *Proceedings of the 2012 ACM research in applied computation symposium*. ACM, pp. 1–7.

Sixto, Juan, Aitor Almeida, and Diego López-de-Ipiña (2018). "Analysis of the Structured Information for Subjectivity Detection in Twitter". en. In: *Transactions on Computational Collective Intelligence XXIX*, pp. 163–181.

Speriosu, Michael et al. (2011). "Twitter Polarity Classification with Label Propagation over Lexical Links and the Follower Graph". In: *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, pp. 53–56. ISBN: 978-1-937284-13-8. DOI: 10.1017/CBO9781107415324.004.

Staiano, Jacopo and Marco Guerini (2014). "DepecheMood: a Lexicon for Emotion Analysis from Crowd-Annotated News". In: *arXiv preprint arXiv:1405.1605*. URL: http://arxiv.org/abs/1405.1605 (visited on 03/06/2015).

Strapparava, Carlo and Alessandro Valitutti (2004). "WordNet-Affect: an affective extension of WordNet". In: *Proceedings of LREC*. Vol. 4, pp. 1083–1086.

Strapparava, Carlo, Alessandro Valitutti, et al. (2004). "WordNet Affect: an Affective Extension of WordNet." In: *LREC*. Vol. 4, pp. 1083–1086. URL: http://hnk.ffzg.hr/bibl/lrec2004/pdf/369.pdf (visited on 01/21/2016).

Taboada, Maite et al. (Apr. 2011). "Lexicon-Based Methods for Sentiment Analysis". In: *Computational Linguistics* 37.2, pp. 267–307. ISSN: 0891-2017.

Tan, Chenhao et al. (2011). "User-level Sentiment Analysis Incorporating Social Networks". In: *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '11. New York, NY, USA: ACM, pp. 1397–1405. ISBN:

978-1-4503-0813-7. DOI: 10.1145/2020408.2020614. URL: http://doi.acm.org/10.1145/2020408.2020614 (visited on 03/06/2015).

Tim Berners-Lee (2006). *Linked Data - Design Issues*. 00181. URL: https://www.w3.org/DesignIssues/LinkedData.html (visited on 12/18/2019).

Tommasel, Antonela and Daniela Godoy (Mar. 2018). "A Social-aware online short-text feature selection technique for social media". In: *Information Fusion* 40, pp. 1–17. ISSN: 1566-2535.

Tummarello, Giovanni, Renaud Delbru, and Eyal Oren (2007). "Sindice. com: Weaving the open linked data". In: *The Semantic Web*. Springer, pp. 552–565.

Vandenbussche, Pierre-Yves et al. (2017). "Linked Open Vocabularies (LOV): a gateway to reusable semantic vocabularies on the Web". In: *Semantic Web* 8.3, pp. 437–452.

Volkova, Svitlana, Theresa Wilson, and David Yarowsky (2013). "Exploring demographic language variations to improve multilingual sentiment analysis in social media". In: *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pp. 1815–1827.

Vulcu, Gabriela et al. (May 2014). "Generating Linked-Data based Domain-Specific Sentiment Lexicons from Legacy Language and Semantic Resources". In: *th International Workshop on Emotion, Social Signals, Sentiment & Linked Open Data, co-located with LREC 2014,* Reykjavik, Iceland: LREC2014, pp. 6–9.

Wang, Sida and Christopher D. Manning (2012). "Baselines and Bigrams: Simple, Good Sentiment and Topic Classification". In: *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers - Volume 2*. ACL '12. Stroudsburg, PA, USA: Association for Computational Linguistics, pp. 90–94.

Wei, Wei and Jon Atle Gulla (2010). "Sentiment learning on product reviews via sentiment ontology tree". In: *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. 00134. Association for Computational Linguistics, pp. 404–413.

Westerski, Adam, Carlos A. Iglesias, and Fernando Tapia (Oct. 2011). "Linked Opinions: Describing Sentiments on the Structured Web of Data". In: *Proceedings of the Fourth International Workshop on Social Data on the Web (SDoW2011)*. CEUR, pp. 21–32.

Wilde, E. and M. Duerst (Apr. 2008). *URI Fragment Identifiers for the text/plain Media Type*. Internet Engineering Task Force.

Xia, Rui and Chengqing Zong (2010). "Exploring the Use of Word Relation Features for Sentiment Classification". In: *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*. COLING '10. Stroudsburg, PA, USA: Association for Computational Linguistics, pp. 1336–1344.

Xiaomei, Zou et al. (Feb. 2018). "Microblog sentiment analysis with weak dependency connections". English. In: *Knowledge-Based Systems* 142, pp. 170–180. ISSN: 0950-7051.

Yan, Jiajun et al. (2008). "The Creation of a Chinese Emotion Ontology Based on HowNet." In: *Engineering Letters* 16.1, pp. 166–171. URL: http://uais.lzu.edu.cn/uploads/soft/101219/TheCreationofaChineseEmotionOntologyBasedonHowNet.pdf (visited on 01/21/2016).

List of Figures

List of Tables

Glossary

EARL Emotion Annotation and Representation Language

EMO Emotion Ontology

EmotionML Emotion Markup Language

HEO Human Emotion Ontology

HUMAINE Human-Machine Interaction Network on Emotion

LLOD Linguistic Linked Open Data

NIF NLP Interchange Format

NLP Natural Language Processing

OLiA Ontologies of Linguistic Annotation

OWLG Open Linguistics Working Group

OntoLex Ontology-Lexica Community Group

lemon Lexicon Model for Ontologies

OSN Online Social Network

SNA Social Network Analysis

NLP Natural Language Processing

OWL Web Ontology Language

RDF Resource Description Framework

TEI Text Encoding Initiative

FOAF Friend of a Friend

LOD Linked Open Data

APPENDIX A

Publications

A.1 Summary of publications

The following tables summarizes the publications made throughout this thesis, grouped into four categories: publications on vocabularies (Table A.1, publications on Linked Data services (Table A.2), publications on social context (Table A.3), and publications that are not directly related to this thesis (Table A.4). The following color scheme has been used: gray for conference papers, blue for journal papers, and white for other types of publications (e.g., book chapters).

Table A.1: Publications on semantic vocabularies

Page	Title	Year	Venue	Ranking
91	Linguistic Linked Data for Sentiment Analysis	2013	2nd Workshop on Linked Data in Linguistics (LDL-2013): Representing and linking lexicons, terminologies and other language data. Collocated with the Conference on Generative Approaches to the Lexicon	

Table A.1: Publications on semantic vocabularies

30	Onyx: Describing Emotions on the Web of Data	2013	Proceedings of the First International Workshop on Emotion and Sentiment in Social and Expressive Media: approaches and perspectives from AI (ESSEM 2013)	
86	EUROSENTIMENT: Linked Data Sentiment Analysis	2014	Proceedings of the ISWC 2014 Posters & Demonstrations Track a track within the 13th International Semantic Web Conference (ISWC 2014)	
100	A Linked Data Approach to Sentiment and Emotion Analysis of Twitter in the Financial Domain	2014	Second International Workshop on Finance and Economics on the Semantic Web (FEOSW 2014)	
113	Generating Linked-Data based Domain-Specific Sentiment Lexicons from Legacy Language and Semantic Resources	2014	International Workshop on Emotion, Social Signals, Sentiment & Linked Open Data, co-located with LREC 2014,	
76	A Linked Data Model for Multimodal Sentiment and Emotion Analysis	2015	4th Workshop on Linked Data in Linguistics: Resources and Applications	
68	Towards a Common Linked Data Model for Sentiment and Emotion Analysis	2016	Proceedings of the LREC 2016 Workshop Emotion and Sentiment Analysis (ESA 2016)	
43	Onyx: A Linked Data Approach to Emotion Representation	2016	Information Processing & Management	JCR 2016 Q1 (2.391)

Table A.2: Publications on Linked Data tools for sentiment analysis

Page	Title	Year	Venue	Ranking
145	Senpy: A Pragmatic Linked Sentiment Analysis Framework	2016	Proceedings DSAA 2016 Special Track on Emotion and Sentiment in Intelligent Systems and Big Social Data Analysis (SentISData)	
125	Multimodal Multimodel Emotion Analysis as Linked Data	2017	Proceedings of ACII 2017	
118	Senpy: A framework for semantic sentiment and emotion analysis services	2019	Knowledge-Based Systems	JCR 2018 Q1 (5.101)

Table A.2: Publications on Linked Data tools for sentiment analysis

132	MixedEmotions: An Open-Source Toolbox for Multi-Modal Emotion Analysis	2018	IEEE Transactions on Multimedia	JCR 2018 Q1 (4.292)

Table A.3: Publications on Social context

Page	Title	Year	Venue	Ranking
154	Social context in sentiment analysis: Formal definition, overview of current trends and framework for comparison	2019	Information Fusion	JCR 2018 Q1 (10.716)
-	Analyzing Radicalism Spread Using Agent-Based Social Simulation	2019	Engineering Multi-Agent Systems 6th International Workshop, EMAS 2018 Stockholm, Sweden, July 14–15, 2018 Revised Selected Papers	
215	A Model of Radicalization Growth using Agent-based Social Simulation	2018	Proceedings of EMAS 2018	CORE-B (CORE 2018)
232	Modeling Social Influence in Social Networks with SOIL, a Python Agent-Based Social Simulator	2017	Advances in Practical Applications of Cyber-Physical Multi-Agent Systems: The PAAMS Collection	
238	Soil: An Agent-Based Social Simulator in Python for Modelling and Simulation of Social Networks	2017	Advances in Practical Applications of Cyber-Physical Multi-Agent Systems: The PAAMS Collection	CORE-C
??	CRANK: A Hybrid Model for User and Content Sentiment Classification Using Social Context and Community Detection	2020	Applied Sciences	JCR 2018 Q2 (2.217)

Table A.4: Publications indirectly related to this thesis

Page	Title	Year	Venue	Ranking
303	A Big Linked Data Toolkit for Social Media Analysis and Visualization based on W3C Web Components	2018	On the Move to Meaningful Internet Systems. OTM 2018 Conferences. Part II	

Table A.4: Publications indirectly related to this thesis

322	An Emotion Aware Task Automation Architecture Based on Semantic Technologies for Smart Offices	2018	Sensors	JCR 2018 (3.031)	Q1
343	Enhancing Deep Learning Sentiment Analysis with Ensemble Techniques in Social Applications	2017	Expert Systems with Applications	JCR (2.981)	Q1
355	A modular architecture for intelligent agents in the evented web	2017	Web Intelligence		
356	Applying Recurrent Neural Networks to Sentiment Analysis of Spanish Tweets	2017	TASS 2017: Workshop on Semantic Analysis at SEPLN		
363	Aspect based Sentiment Analysis of Spanish Tweets	2015	Proceedings of TASS 2015: Workshop on Sentiment Analysis at SEPLN co-located with 31st SEPLN Conference (SEPLN 2015)		
370	MAIA: An Event-based Modular Architecture for Intelligent Agents	2014	Proceedings of 2014 IEEE/WIC/ACM International Conference on Intelligent Agent Technology		
379	EuroLoveMap: Confronting feelings from News	2014	Proceedings of Come Hack with OpeNER!" workshop at the 9th Language Resources and Evaluation Conference (LREC'14)		

A.2 Publications indirectly related to the thesis

A.2.1 A Big Linked Data Toolkit for Social Media Analysis and Visualization based on W3C Web Components

Title	A Big Linked Data Toolkit for Social Media Analysis and Visualization based on W3C Web Components
Authors	Sánchez-Rada, J. Fernando and Pascual-Saavedra, Alberto and Conde-Sánchez, Enrique and Iglesias, Carlos A.
Proceedings	On the Move to Meaningful Internet Systems. OTM 2018 Conferences. Part II
ISBN	978-3-030-02671-4
Volume	11230
Year	2018
Keywords	big data, linked data, sefarad, soneti, web components
Pages	498–515
Abstract	Social media generates a massive amount of data at a very fast pace. Objective information such as news, and subjective content such as opinions and emotions are intertwined and readily available. This data is very appealing from both a research and a commercial point of view, for applications such as public polling or marketing purposes. A complete understanding requires a combined view of information from different sources which are usually enriched (e.g .sentiment analysis) and visualized in a dashboard. In this work, we present a toolkit that tackles these issues on different levels: 1) to extract heterogeneous information, it provides independent data extractors and web scrapers; 2) data processing is done with independent semantic analysis services that are easily deployed; 3) a configurable Big Data orchestrator controls the execution of extraction and processing tasks; 4) the end result is presented in a sensible and interactive format with a modular visualization framework based on Web Components that connects to different sources such as SPARQL and ElasticSearch endpoints. Data workflows can be defined by connecting different extractors and analysis services. The different elements of this toolkit interoperate through a linked data principled approach and a set of common ontologies. To illustrate the usefulness of this toolkit, this work describes several use cases in which the toolkit has been successfully applied.

A Big Linked Data Toolkit for Social Media Analysis and Visualization based on W3C Web Components

J. Fernando Sánchez-Rada, Alberto Pascual,
Enrique Conde, and Carlos A. Iglesias

Grupo de Sistemas Inteligentes, Universidad Politécnica de Madrid
jf.sanchez@upm.es, a.pascuals@alumnos.upm.es, carlosangel.iglesias@upm.es
http://www.gsi.dit.upm.es

Abstract. Social media generates a massive amount of data at a very fast pace. Objective information such as news, and subjective content such as opinions and emotions are intertwined and readily available. This data is very appealing from both a research and a commercial point of view, for applications such as public polling or marketing purposes. A complete understanding requires a combined view of information from different sources which are usually enriched (e.g .sentiment analysis) and visualized in a dashboard.
In this work, we present a toolkit that tackles these issues on different levels: 1) to extract heterogeneous information, it provides independent data extractors and web scrapers; 2) data processing is done with independent semantic analysis services that are easily deployed; 3) a configurable Big Data orchestrator controls the execution of extraction and processing tasks; 4) the end result is presented in a sensible and interactive format with a modular visualization framework based on Web Components that connects to different sources such as SPARQL and ElasticSearch endpoints. Data workflows can be defined by connecting different extractors and analysis services. The different elements of this toolkit interoperate through a linked data principled approach and a set of common ontologies. To illustrate the usefulness of this toolkit, this work describes several use cases in which the toolkit has been successfully applied.

Keywords: Linked Data · Web Components · Visualization · Social Media · Big Data .

1 Introduction

We are used to the never-ending stream of data coming at us from social media. Social media has become a way to get informed about the latest facts, faster than traditional media. It is also an outlet for our complaints, celebrations and feelings, in general. This mix of factual and subjective information has drawn the interest of research and business alike. The former, because social media could

be used as a proxy to public opinion, a probe for the sentiment of the people. The latter, because knowing the interests and experience of potential users is the holy grail of marketing and advertisement.

However, making sense of such a big stream of data is costly in several ways. The main areas that need to be covered are: extraction, analysis, storage, visualization and orchestration. All these aspects are further influenced by the typical attributes of Big Data such as large volume, large throughput and heterogeneity. We will cover each of them in more detail.

First of all, in order to analyze this data, it needs to be extracted. The volume of data and metadata available in today's media can be overwhelming. For instance, a simple tweet, which in principle consists of roughly 140 characters, contains dozens of metadata fields such as creation date, number of retweets, links to users mentioned in the text, geolocation, plus tens of fields about the original poster. Some of this data is very useful, whereas some information (e.g. the background color of the author's profile) are seldom used. Furthermore, Twitter is one of the best case scenarios, because it provides a well documented API. Other media require collecting unstructured information, or using more cumbersome techniques.

The extracted data needs to be stored and made available for analysis and visualization. Since the volume of data is potentially very large, the data store needs to keep up with this pace, and provide means to quickly query parts of the data. Modern databases such as ElasticSearch or Cassandra have been designed for such types of loads. However, analysis requires using data from different sources. For the sake of interoperability and simplicity, data from different sources should be structured and queried using the same formats. Hence, using vocabularies and semantic technologies such as RDF and SPARQL would be highly beneficial.

The next area is data analysis. The analysis serves different purposes, such as enriching the data (e.g. sentiment analysis), transforming it (e.g. normalization and filtering) or calculating higher order metrics (e.g. aggregation of results). Unfortunately, different analysis processes usually require different tooling, formatting and APIs, which further complicates matters.

Finally, there is visualization, where the results of the analysis are finally presented to users, in a way that allows them to explore the data. This visualization needs to be adaptable to different applications, integrated with other analysis tools, and performance. In practice, visualization is either done with highly specialized tools such as Kibana [13], with little integration with other products, or custom-tailored to each specific application, which hinders reusability.

And, lastly, all these steps need to be repeated for every application. This is, once again, typically done on an ad-hoc basis, and every step in the process is manually programmed or configured via specialized tools.

This work presents a toolkit that that deals with these issues on different levels: 1) to extract heterogeneous information, it provides independent data extractors and web scrapers; 2) data processing is done with independent semantic analysis services that are easily deployed; 3) a configurable Big Data orchestra-

tor controls the execution of extraction and processing tasks; 4) the end result is presented in a sensible and interactive format with a modular visualization framework based on Web Components that connects to different sources such as SPARQL and ElasticSearch endpoints. Data workflows can be defined by connecting different extractors and analysis services. The different elements of this toolkit interoperate through a Linked Data principled approach and a set of common ontologies. The combination of Linked Data principles and Big Data is often referred to as Big Linked Data. The toolkit has been successfully used in several use cases, in different domains, which indicates that it is useful in real scenarios.

The remaining sections are structured as follows: Section 2 presents technologies and concepts that this work is based on; Section 3 explains the architecture of the toolkit, and its different modules; Section 4 illustrates the use of this toolkit in different use cases; Lastly, Section 5 presents our conclusions and future lines of work.

2 Enabling Technologies

2.1 W3C Web Components

W3C Web components are a set of web platform APIs that allow the creation of new custom, reusable, encapsulated HTML tags to use in web pages and web apps. This Web Components idea comes from the union of four main standards: custom HTML elements, HTML imports, templates and shadow DOMs.

- *Custom Elements:* Custom Elements [47] let the user define his own element types with custom tag names. JavaScript code is associated with the custom tags and uses them as an standard tag. Custom elements specification is being incorporated into the W3C HTML specification and will be supported natively in HTML5.3
- *HTML imports:* HTML Imports [26] let users include and reuse HTML documents in other HTML documents, as 'script' tags let include external JavaScript in pages.
- *Templates:* Templates [9] define a new 'template' element which describes a standard DOM-based approach for client-side template. Templates allow developers to declare fragments of markup which are parsed as HTML.
- *Shadow DOM:* Shadow DOM [17] is a new DOM feature that helps users build components. Shadow DOMs can be seen as a scoped sub-tree inside your element.

In order to make compatible these W3C Web components with modern browsers, a number frameworks have emerged to foster their use.

Polymer is one of these emerging frameworks for constructing Web Components that was developed by Google[1]. Polymer simplifies building customized

[1] https://www.polymer-project.org/

and reusable HTML components. In addition, Polymer has been designed to be flexible, fast and close. It uses the best specifications of the web platform in a direct way to simply custom elements creation.

2.2 Emotion and Sentiment Models and Vocabularies

Linked Data and open vocabularies play a key role in this work. The semantic model used enables both the use of interchangeable services and the integration of results from different services. It focuses on Natural Language Processing (NLP) service definition, the result of such services, sentiments and emotions. Following a Linked Data approach, the model used is based on the following existing vocabularies:

- NLP Interchange Format (NIF) 2.0 [15] defines a semantic format for improving interoperability among natural language processing services. To this end, texts are converted to RDF literals and an URI is generated so that annotations can be defined for that text in a linked data way. NIF offers different URI Schemes to identify text fragments inside a string, e.g. a scheme based on RFC5147 [49], and a custom scheme based on context.
- Marl [46], a vocabulary designed to annotate and describe subjective opinions expressed on the web or in information systems.
- Onyx [31], which is built on the same principles as Marl to annotate and describe emotions, and provides interoperability with Emotion Markup Language (EmotionML) [37].
- Schema.org [12] provides entities and relationships for the elements that are outside the realm of the social media itself. For instance, it can be used to annotate product reviews.
- FOAF [11] provides the description of relationships and interactions between people.
- SIOC [4] is used to annotate blog posts, online forums and similar media.
- PROV-O [24] provides provenance information, linking the final results that can be visualized with the original data extracted, the processes that transformed the data, and the agents that took part in the transformation.

Additionally, NIF [15] provides an API for NLP services. This API has been extended for multimodal emotion analysis in previous works [33, 34] . This extension also enables the automatic conversion between different emotion models.

2.3 Senpy

Senpy [32] is a framework for sentiment and emotion analysis services. Services built with Senpy are interchangeable and easy to use because they share a common API and Examples. It also simplifies service development.

All services built using Senpy share a common interface, based on the NIF API [14] and public ontologies. This allows users to use them (almost) interchangeably. Senpy takes care of:

– Interfacing with the user: parameter validation, error handling.
– Formatting: JSON-LD [39], Turtle/n-triples input and output, or simple text input
– Linked Data: Senpy results are semantically annotated, using a series of well established vocabularies.
– User interface: a web UI where users can explore your service and test different settings
– A client to interact with the service. Currently only available in Python.

Senpy services are made up of individual modules (plugins) that perform a specific type of analysis (e.g. sentiment analysis). Plugins are developed independently. Senpy ships with a plugin auto-discovery mechanism to detect plugins locally. There are a number of plugins for different types of analysis (sentiment, emotion, etc.), as well as plugins that wrap external services such as Sentiment140, MeaningCloud and IBM Watson[2].

3 Architecture

This work presents a modular toolkit for processing Big Linked Data encouraging scalability and reusability. The high level architecture of this toolkit, which we call Soneti, is depicted in Figure 1. It integrates existing open source tools with other built specifically for the toolkit. The main modules are orchestration, data ingestion, processing and analysis, storage and visualization and management, which are described below.

Orchestration. The *orchestration module* (Sect. 3.1) is responsible of managing the interaction of the rest modules by automating complex data pipelines and handling failures. This module enables *reusability* at thedata pipeline level, In addition, it enables *scalability*, since every task of the workflow can be executed in a Big Data platform, such as a Hadoop job [48], a Spark job [50] or a Hive query [43], to name a few. Finally, this module helps to recover from failures gracefully and rerun only the uncompleted task dependencies in the case of a failure.

The *Data Ingestion module* (Sect. 3.2) involves obtaining data from the structured and unstructured data sources and transforming these data into linked data formats, using scraping techniques and APIs, respectively. The use of linked data enables *reusability* of ingestion modules as well as *interoperabilty* and provides a uniform schema for processing data.

The *Processing and Analysis module* (Sect. 3.3) collects the different analysis tasks that enrich the incoming data, such as entity detection and linking, sentiment analysis or personality classification. Analysis is based on the NIF recommendation, which has been extended for multimodal data sources. Each

[2] Sentiment140, MeaningCloud and IBM Watson are online sentiment analysis services available at http://www.sentiment140.com/, https://www.meaningcloud.com/ and https://www.ibm.com/watson/services/natural-language-understanding/, respectively.

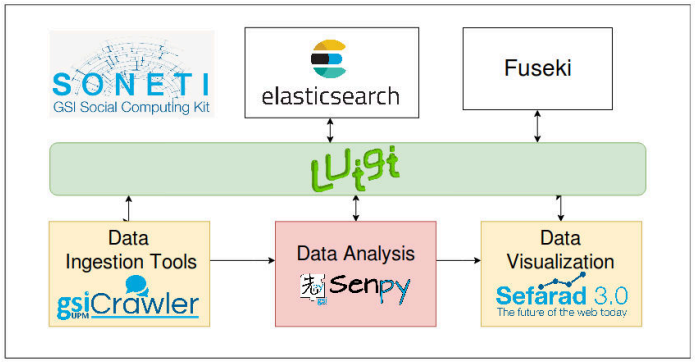


Fig. 1: High-level architecture.

analysis task has been modelled as a plugin of Senpy, presented in Sect. 2.3. In this way, analysis modules can be easily *reused*.

The *Storage module* (Sect 3.4) is responsible for storing data in a nonSQL database. We have selected ElasticSearch [10], since it provides scalability, text search as well as a RESTful server based on a Query DSL language. For our purposes, JSON-LD [39] is used, with the aim of preserving linked data expressivity in a format compatible with the Elasticsearch ecosystem.

The *Visualization and Querying module* (Sect. 3.5) enables building dashboards as well as executing semantic queries. Visualisation is based on W3C Web Components. A library of interconnected Web Component based widgets have been developed to enable faceted search. In addition, one widget has been developed for providing semantic SPARQL queries to a SPARQL endpoint Apache Fuseki [19]. Fuseki is provisioned by the data pipeline.

Figure 2 provides a more detailed view of the architecture, focused on the Visualization and Querying module, to explain its connection to the rest of the modules. The following subsections describe each module in more detail.

3.1 Orchestration

Workflow management systems are usually required for managing the complex and demanding pipelines in Big Data environments. There are a number of open source tools for workflow management, such as Knime [45], Luigi [40], SciLuigi [23], Styx [41], Pinterest's pinball [29] or Airbnb's Airflow [20]. The interested reader can find a detailed comparison in [30, 23].

We have selected as workflow orchestrator the open source software Luigi [40], developed by Spotify. It allows the definition and execution of complex dependency graphs of tasks and handles possible errors during execution. In addition,

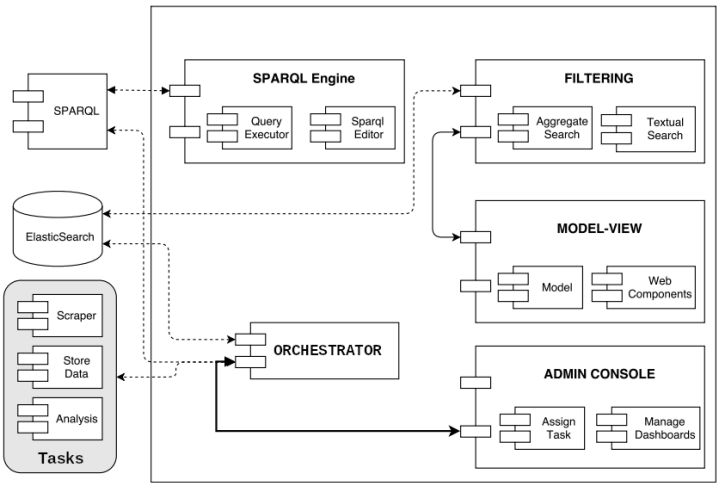


Fig. 2: Detailed architecture of the visualization components.

Luigi provides a web interface to check pipeline dependencies as well as a visual overview of tasks execution.

Luigi is released as a Python module, which provides an homogeneous language since machine learning and natural language processing tasks are also developed in this language.

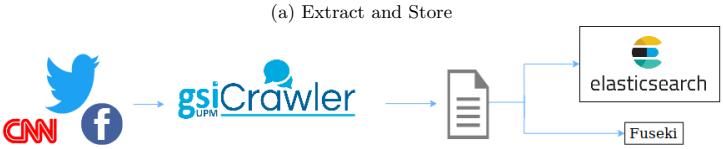
A pipeline is a series of interdependent tasks that are executed in order. Each task is defined by its input (its dependencies), its computation, and its output.

Some examples of the most common pipelines we have reused in a number of projects are shown in Figure 3:

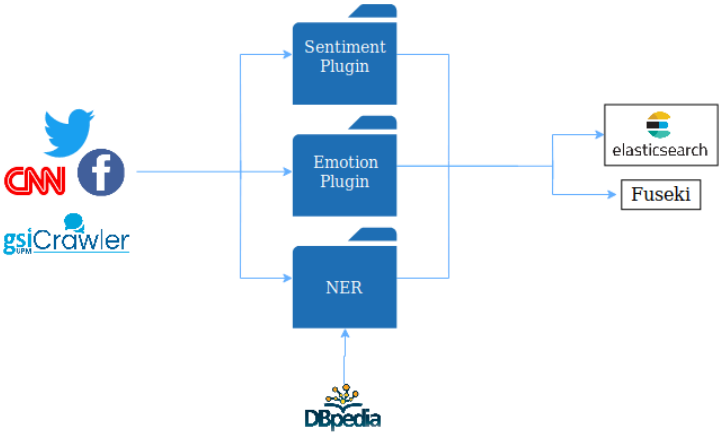
- *Extract and Store:* this workflow extracts data from a number of sources and store them in a JSON-LD format, as shown in Fiure 3a.
- *Extract and Store in a noSQL database and an LD-Server:* this worflow extends the previous worflow by storing in parallel in a noSQL database and a SPARQL endpoint the extracted triples, as depicted in Figure 3b.
- *Extract, Analyze and Store workflow*: this workflow analyze the data before storing it, as shown in Figure 3c. The analysis consists in a data pipeline where each analyzer adds semantic metadata, being the analyzers Senpy plugins. Some examples of these analyzers are sentiment and emotion detection as well as entity recognition and linking.

The processed data in most workflows is stored in one or multiple datastores and formats. Figure 4 illustrates the type of semantic annotations that would

(a) Extract and Store



(b) Extract and Store in a NoSQL database and an LD-Serve



(c) Extract, Analyze and Store

Fig. 3: Examples of different Luigi Workflows.

be generated by a combinatio of three different services (Sentiment Analysis, Emotion Analysis and Named Entity Recognition Analysis).

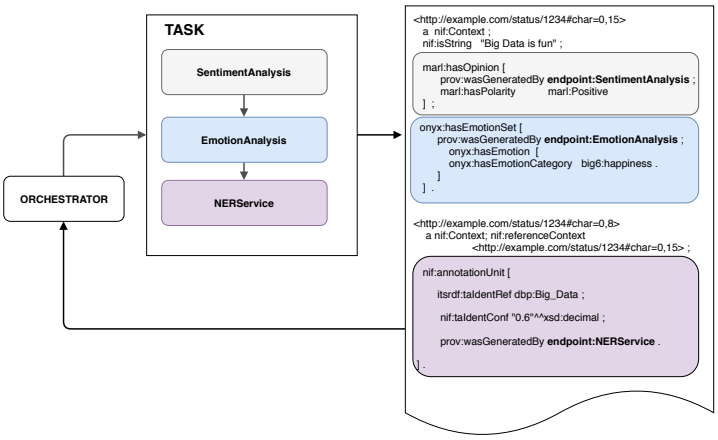


Fig. 4: Enrichment Pipeline results in turtle format. The annotations generated by three independent services have been combined thanks to the Linked Data principles and NIF URI schemes.

3.2 Data Ingestion

The objective of this module is extracting the information from external sources, map it to linked data formats for process and storage.

A tool, so called GSICrawler [3] has been developed to extract information for structured and unstructured sources. The architecture of this tool consists of a set of modules providing a uniform API, which enables its orchestration. GSICrawler contains scraping modules that are based on Scrapy [21], and other modules that connect to external APIs. At the time of writing, there are modules for extracting data from Twitter, Facebook, Reddit, TripAdvisor, Amazon, RSS Feeds, and a number of specific places, including some journals in PDF format. The tool is Open Source and publicly available [4].

Table 1 describes the method available in the GSICrawler API, whereas Table 2 contains the basic parameters for the `/tasks` endpoint. More parameters are available, depending on the type of analysis performed.

[3] GSICrawler's documentation: https://gsicrawler.readthedocs.io
[4] https://github.com/gsi-upm/gsicrawler

Table 1: API endpoints to access tasks and jobs in GSICrawler

endpoint	description
GET /tasks/	Get a list of available tasks in JSON-LD format.
GET /jobs/	Get a list of jobs. It can be limited to pending/running jobs by specifying `?pending=True`
POST /jobs/	Start a new job, from an available task and a set of parameters

Table 2: Basic parameters for a new job. Other parameters may be needed or available, depending on the task.

parameter	description
task_id	Identifier of the task. Example: pdf-crawl.
output	Where to store the results. Available options: `none`, `file`, `elasticsearch`.
retries	If the task fails, retry at most this many times. Optional.
delay	If specified, the job will be run in `delay` seconds instead of immediately. Optional.
timeout	Time in seconds to wait for the results. If the timeout is reached, consider the task failed. Optional.

3.3 Data Processing and Analysis

There is an array of processing tasks that are relevant for social media analysis. The most common are text-based processes such as sentiment and emotion analysis, named entity recognition, or spam detection. These types of processes are covered both from an API point of view (Section 2.3) and from a modelling point of view (Section 2.2), with NIF and its extensions.

Having a common API for analysis services, as covered in Section 2.3, avoids coupling other parts of the system to the idiosyncrasies of the specific services used. As a result, services that provide equivalent types of annotation (e.g. two sentiment services) are interchangeable as far as the rest of the system is concerned. The obvious downside is that, in order to reach this level of decoupling, external services need to be adapted either natively or through the use of additional layers such as proxies and wrappers. Fortunately, the number of services is much lower than the number of applications using them, which makes adapting services much more efficient than having to adapt systems to include other services.

Using a common semantic model for results and annotations means that other modules in the system, especially the visualization module, do not need to rely on specific schemata or formats for every service or type of service. They can focus only on representing the information itself. Semantic standards such as RDF also ensure that applications can be agnostic of the specific serialization format used (e.g. JSON or XML). This independence is exploited in other modules of the toolkit. For example, more than one type of datastore can be used as storage modules, each of them with their own formats. An ElasticSearch database (JSON-based) may co-exist with a Fuseki (RDF-based) datastore, provided the

annotations are correct and the appropriate conversion mechanisms (e.g. framing in the case of JSON-LD) are in place.

All services that are compatible with Senpy's API and format can be used with the toolkit proposed in this paper. An updated list is available at Senpy's documentation and, it includes services for sentiment analysis, emotion analysis, NER, age and gender detection, radicalization detection, spam detection, etc.

3.4 Storage

Our toolkit takes two types of storage into consideration: SPARQL endpoints and traditional datastores with a REST API. In practice, we have employed Fuseki's SPARQL endpoint [19], and ElasticSearch's REST API [10].

One of the main reasons to support other datastores is the need for Big Data analysis. In particular, we focused on ElasticSearch. Elasticsearch is a search server based on Lucene. It provides a distributed, full-text search engine with an HTTP web interface and schema-free JSON documents. ElasticSearch has been widely used in Big Data applications due to its performance and scalability. ElasticSearch nodes can be distributed, it divides indices into shards, each of which can have zero or more replicas. Each node hosts one or more shards, and acts as a coordinator to delegate operations to the correct shard(s).

To retain semantics, we use a subset (or dialect) of JSON, JSON-LD [39], which adds semantic annotation to plain JSON objects.

3.5 Visualization and Querying

One of the main goals of the toolkit is to provide a component-based UI framework that can be used to quickly develop custom data visualizations that lead to insights.

Re-usability and composability were two of our main requirements for the framework. For this reason, we chose to base the visualization module on W3C Web Components. Web Components is an increasingly popular set of standards that enable the development of reusable components. Using Web Components adds a layer of complexity, especially when it is combined with the usual visualization libraries (e.g. D3.js [5]). Fortunately, the additional effort is outweighed by the growing community behind Web Components and the increasing number of compatible libraries.

Nevertheless, several aspects were not fully covered by the current standard. In particular, we wanted to provide faceted search combined with text search and web component communication. To do so, componenents need to communicate with each other, which requires a set of conventions on top of Web Components.

There are several alternatives for web component communication [42], such as custom events between components and publish-subscribe pattern [22]. After considering these alternatives, we chose to follow a Model-View-Controller

[5] https://d3js.org/

(MVC) architecture (Figure 2). In MVC, a single element is in charge of connecting to the data sources, filtering the results, and exposing it to other components, which can them present it.

Since this visualization should also be interactive, visualization components also contain their own set of filters. When interacting with these components, a user may modify the filters, and the component will communicate the change of filters back to the filtering component. This allows storing which elements have been selected and thus making more complex queries to data sources (e.g. ElasticSearch). The communication between components is achieved through observers and computed properties, which allow changes to be seamlessly propagated to all components.

The result of combining Web Components with these conventions to organize data is Sefarad [6], an Open Source code [7] framework which is the core of the visualization module in the toolkit. Visualizations in Sefarad are composed of individual dashboards, which are web pages oriented to display all groups of related information (e.g. visualization of the activity of a brand in social media). In turn, these dashboards are further divided into widgets (e.g. charts and lists), which are connected to present a coherent and interactive view.

Dashboards serve the purpose of integrating a collection of widgets and connecting them to the data sources (e.g. Fuseki). Hence, dashboards are custom-tailored to specific applications, and are not as reusable as widgets. There are two main types of dashboards. On one hand, there are dashboards that provide a simple interface with interactive widgets, filters and textual search. This type of dashboards is aimed towards inexperienced users. Hence, their actions are guided with pre-defined queries and suggestions. On the other hand, we find dashboards that cater to more advanced users, who can explore the dataset through more complex queries using a SPARQL editor. These results can be viewed in raw format, using pivot tables, and through compatible widgets.

As shown in Figure 5, Serafad is also capable of retrieving semantic data from external sources, such as Elasticsearch, Fuseki or DBPedia. Data retrieving is done by an client (e.g. Elasticsearch) located at the dashboard. This client stores data and it is shared with all the widgets within that dashboard.

At the time of writing, this is a categorized list of popular widgets in the toolkit:

- **Data statistics widgets:** These widgets are used to visualize data statistics from an Elasticsearch index at a glance. We include inside this category Google-chart-elasticsearch, number-chart, spider-chart, Liquid-fluid-d3, wordcloud...
- **Sentiment widgets:** These widgets are used to visualize sentiment information. We include inside this category chernoff-faces, field-chart, tweet-chart, wheel-chart, youtube-sentiment...

[6] Sefarad's documentation: http://sefarad.readthedocs.io/
[7] https://github.com/gsi-upm/sefarad

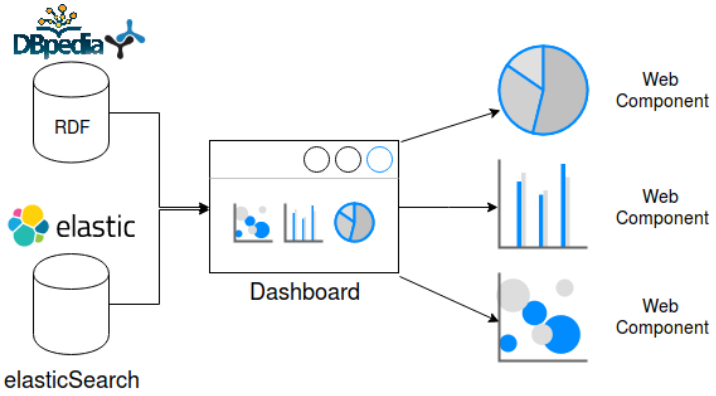


Fig. 5: Architecture of the Visualization module

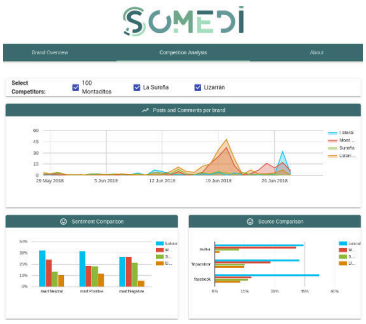
- **NER widgets:** These widgets are used to visualize recognized entities from an Elasicsearch index. We include inside this category entities-chart, people-chart, aspect-chart, wheel-chart...
- **Location widgets:** This group of widgets visualize data geolocated in different maps. Spain-chart, happymap and leaflet-maps are some examples of this kind of widgets.
- **Document widgets:** Inside this group we can find tweet-chart and news-chart. These widgets are used to visualize all documents within an Elasticsearch index.
- **Query widgets:** This widgets add more functionalities to Sefarad framework, they are used to modify or ask queries to different endpoints. We include inside this category material-search, YASGUI-polymer, date-slider...

4 Case Studies

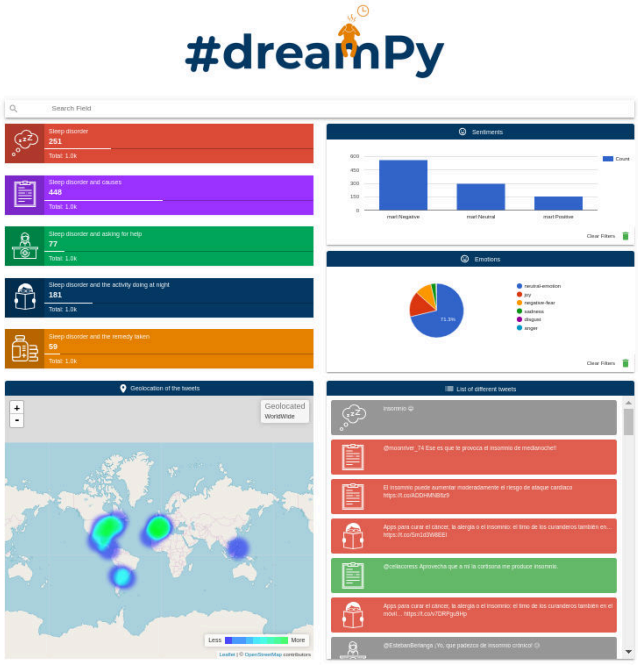
The platform has been used in a number of national and European R&D projects, such as Financial Twitter Tracker [3, 35], FP7 SmartOpenData [36, 7], H2020 Trivalent [2], and ITEA Somedi. In addition, the platform has been applied in several master thesis in sentiment and emotion detection in Twitter, Facebook and web sites in different domains, such as football [28, 25, 27], song lyrics [18], geolocated media [8], political parties [1], financial news [44, 6] and e-commerce [5]. It has also been applied for detection of insomnia [38] and radicalism [16] in Twitter [38]. For illustration purposes, we will describe three cases were the platform has been used.

Figure 6a shows the Trivalent dashboard. In this case, the purpose of the dashboard is to analyze radicalization sources, including Twitter, news papers

(a) Screenshot from the Trivalent Dashboard



(b) Brand monitoring for the SoMeDi project.



(c) Example of analysis of insommia in Twitter.

Fig. 6: Case studies

(CNN, New York Times and Aljazeera) and radicalist magazines (Dabiq and Rumiyah). The crawler collects all the information and processes in different ways. In particular, the current version includes entity recognition and linking, topic identification and sentiment and emotion analysis. Future versions will also include narrative detection.

The second case is brand monitoring and analysis of competitors, within the SoMeDi project. The goal was to compare the social media activity related to a given brand with that of the competition. To this end, the GSICrawler service fetches data about the brand and its competition from social networks (Twitter, Facebook and Tripadvisor). Secondly, this data is enriched using sentiment analysis and named entity recognition. Figure 6b shows a partial view of the dashboard. In this case, it was important to present the results in different points in time, to filter out specific campaigns and to compare the evolution of the activity for each entity.

Another example is shown in Figure 6c. In this case, the system analyzes the timeline of Twitter users for determining if they suffer from insommia and the underlying reasons. In this case, the data ingestion comes from Twitter and a Senpy plugin carries out the classifications. The development of Senpy plugins is straight forward, since Scikit-learn classifiers can be easily exposed as Senpy plugins, by defining the mapping to linked data properties.

5 Conclusions

The motivation of this work was to leverage Linked Data to enable the analysis of social media, and later visualization of the results, with reusable components and configurable workflows. The result is a toolkit that relies on a set of vocabularies and semantic APIs for interoperability. The toolkit's architecture is highly composable, with modularity and loose coupling as driving principles. In particular, the visualization elements are based on web components, which introduce new development paradigms such as the shadow DOM and templates. Adapting to this new paradigm takes some time, but results in highly reusable code. On the processing side, using a semantic approach with a combination of ontologies and the NIF API has made it possible to seamlessly combine different analysis services. The fact that analysis results are semantically annotated has made using components easy.

The toolkit is under an Open Source license, and its modules are publicly available on GitHub[8]. Several demonstrations also showcase the usefulness of the visualization in each use case.

To further expand this toolkit, we are already working on integrating the visualization components with React [9], the JavaScript library by Facebook. Once the integration is complete, the full ecosystem of UI elements in React will be available in widgets and dashboards. On the other hand, the options for data processing are not limited to text. If information such as user relevance or content

[8] Soneti's documentation: https://soneti.readthedocs.io/
[9] https://reactjs.org/

diffusion are important for an application, other techniques like social network analysis are needed. These types of analysis are not covered by any generic specification that we know of. For this reason, we are also working on defining the types of analysis of online social networks, in order to provide a vocabulary and an API just like NIF did for NLP.

Acknowledgements

The authors want to thank Roberto Bermejo, Alejandro Saura, Rubén Díaz and José Carmona for working on previous versions of the toolkit. In addition, we want to thank Marcos Torres, Jorge García-Castaño, Pablo Aramburu, Rodrigo Barbado, Jose Mª Izquierdo, Mario Hernando, Carlos Moreno, Javier Ochoa and Daniel Souto, who have applied the toolkit in different domains. Lastly, we also thank our partners at Taiger and HI-Iberia for using the toolkit and collaborating in the integration of their analysis services with the toolkit as part of project SoMeDi (ITEA3 16011).

This work is supported by ITEA 3 EUREKA Cluster programme together with the National Spanish Funding Agencies CDTI (INNO-20161089) and MINETAD (TSI-102600-2016-1), the Spanish Ministry of Economy and Competitiveness under the R&D project SEMOLA (TEC2015-68284-R) and by the European Union through the project Trivalent (Grant Agreement no: 740934).

References

1. Aramburu García, P.: Design and Development of a Sentiment Analysis System on Facebook from Political Domain. Master's thesis, ETSI Telecomunicación (June 2017)
2. Barbado, R.: Design of a Prototype of a Big Data Analysis System of Online Radicalism based on Semantic and Deep Learning technologies. Tfm, ETSI Telecomunicación (June 2018)
3. Bermejo, R.: Desarrollo de un Framework HTML5 de Visualización y Consulta Semántica de Repositorios RDF. Master's thesis, Universidad Politécnica de Madrid (June 2014)
4. Breslin, J.G., Decker, S., Harth, A., Bojars, U.: Sioc: an approach to connect web-based communities. International Journal of Web Based Communities **2**(2), 133–142 (2006)
5. Carmona, J.E.: Development of a Social Media Crawler for Sentiment Analysis. Master's thesis, ETSI Telecomunicación (feb 2016)
6. Conde-Sánchez, E.: Development of a Social Media Monitoring System based on Elasticsearch and Web Components Technologies. Master's thesis, ETSI Telecomunicación (June 2016)
7. Díaz-Vega, R.: Design and implementation of an HTML5 Framework for biodiversity and environmental information visualization based on Geo Linked Data. Master's thesis, ETSI Telecomunicación (December 2014)
8. García-Castaño, J.: Development of a monitoring dashboard for sentiment and emotion in geolocated social media. Master's thesis, ETSI Telecomunicación (July 2017)

9. Glazkov, D., Weinstein, R., Ross, T.: Html templates w3c working group note 18. Tech. rep., W3C (March 2014)
10. Gormley, C., Tong, Z.: Elasticsearch: The Definitive Guide: A Distributed Real-Time Search and Analytics Engine. " O'Reilly Media, Inc." (2015)
11. Graves, M., Constabaris, A., Brickley, D.: Foaf: Connecting people on the semantic web. Cataloging & classification quarterly **43**(3-4), 191–202 (2007)
12. Guha, R.V., Brickley, D., Macbeth, S.: Schema.org: evolution of structured data on the web. Communications of the ACM **59**(2), 44–51 (2016)
13. Gupta, Y.: Kibana Essentials. Packt Publishing Ltd (2015)
14. Hellmann, S.: Integrating Natural Language Processing (NLP) and Language Resources using Linked Data. Ph.D. thesis, Universität Leipzig (2013)
15. Hellmann, S., Lehmann, J., Auer, S., Brümmer, M.: Integrating nlp using linked data. In: The Semantic Web–ISWC 2013, pp. 98–113. Springer (2013)
16. Hernando, M.: Development of a Classifier of Radical Tweets using Machine Learning Algorithms. Master's thesis, ETSI Telecomunicación (January 2018)
17. Ito, H.: Shadow DOM. Tech. rep., W3C (Mar 2018)
18. Izquierdo-Mora, J.M.: Design and Development of a Lyrics Emotion Analysis System for Creative Industries. Master's thesis, ETSI Telecomunicación (January 2018)
19. Jena, A.: Apache jena fuseki. The Apache Software Foundation (2014)
20. Kotliar, M., Kartashov, A., Barski, A.: CWL-Airflow: a lightweight pipeline manager supporting common workflow language. bioRxiv p. 249243 (2018)
21. Kouzis-Loukas, D.: Learning Scrapy. Packt Publishing Ltd (2016)
22. Krug, M.: Distributed event-based communication for web components. In: Proceedings of Studierendensymposium Informatik 2016 der TU Chemnitz. pp. 133–136 (2016)
23. Lampa, S., Alvarsson, J., Spjuth, O.: Towards agile large-scale predictive modelling in drug discovery with flow-based programming design principles. Journal of cheminformatics **8**(1), 67 (2016)
24. Missier, P., Belhajjame, K., Cheney, J.: The w3c prov family of specifications for modelling provenance metadata. In: Proceedings of the 16th International Conference on Extending Database Technology. pp. 773–776. ACM (2013)
25. Moreno Sánchez, C.: Design and Development of an Affect Analysis System for Football Matches in Twitter Based on a Corpus Annotated with a Crowdsourcing platform. Master's thesis, ETSI Telecomunicación (2018)
26. Morita, H., Glazkov, D.: HTML imports. W3C working draft, W3C (Feb 2016)
27. Ochoa, J.: Design and Implementation of a Scraping system for Sport News. Master's thesis, ETSI Telecomunicación (February 2017)
28. Pascual-Saavedra, A.: Development of a Dashboard for Sentiment Analysis of Football in Twitter based on Web Components and D3.js. Master's thesis, ETSI Telecomunicación (June 2016)
29. Pinterest: Pinball, available at https://github.com/pinterest/pinball
30. Ranic, T., Gusev, M.: Overview of workflow management systems. In: Proceedings of the 14th International Conference for Informatics and Information Technology (CIIT 2017). Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, Macedonia (2017)
31. Sánchez-Rada, J.F., Iglesias, C.A.: Onyx: A linked data approach to emotion representation. Information Processing & Management **52**(1), 99–114 (2016)
32. Sánchez-Rada, J.F., Iglesias, C.A., Corcuera, I., Araque, O.: Senpy: A pragmatic linked sentiment analysis framework. In: Data Science and Advanced Analytics (DSAA), 2016 IEEE International Conference on. pp. 735–742. IEEE (2016)

33. Sánchez-Rada, J.F., Iglesias, C.A., Gil, R.: A Linked Data Model for Multimodal Sentiment and Emotion Analysis. In: Proceedings of the 4th Workshop on Linked Data in Linguistics: Resources and Applications. pp. 11–19. Association for Computational Linguistics, Beijing, China (July 2015)
34. Sánchez-Rada, J.F., Iglesias, C.A., Sagha, H., Schuller, B., Wood, I., Buitelaar, P.: Multimodal multimodel emotion analysis as linked data. In: Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW), 2017 Seventh International Conference on. pp. 111–116. IEEE (2017)
35. Sánchez-Rada, J.F., Torres, M., Iglesias, C.A., Maestre, R., Peinado, R.: A Linked Data Approach to Sentiment and Emotion Analysis of Twitter in the Financial Domain. In: Second International Workshop on Finance and Economics on the Semantic Web (FEOSW 2014). vol. 1240, pp. 51–62 (May 2014), http://ceur-ws.org/Vol-1240/
36. Saura Villanueva, A.: Development of a framework for GeoLinked Data query and visualization based on web components. Pfc, ETSI Telecomunicación (June 2015)
37. Schröder, M., Baggia, P., Burkhardt, F., Pelachaud, C., Peter, C., Zovato, E.: Emotionml – an upcoming standard for representing emotions and related states. In: D'Mello, S., Graesser, A., Schuller, B., Martin, J.C. (eds.) Affective Computing and Intelligent Interaction, Lecture Notes in Computer Science, vol. 6974, pp. 316–325. Springer Berlin Heidelberg (2011)
38. Souto, D.S.: Design and development of a system for sleep disorder characterization using Social Media Mining. Master's thesis, ETSI Telecomunicación, ETSIT, Madrid (June 2018)
39. Sporny, M., Kellogg, G., Lanthaler, M.: Json-ld 1.0 (Jan 2014), http://json-ld.org/spec/latest/json-ld/
40. Spotify: Luigi, available at https://github.com/spotify/luigi
41. Stephen, J.J., Savvides, S., Sundaram, V., Ardekani, M.S., Eugster, P.: STYX: Stream processing with trustworthy cloud-based execution. In: Proceedings of the Seventh ACM Symposium on Cloud Computing. pp. 348–360. ACM (2016)
42. Stokolosa, V.: Communication between components (2018), https://hackernoon.com/communication-between-components-7898467ce15b
43. Thusoo, A., Sarma, J.S., Jain, N., Shao, Z., Chakka, P., Zhang, N., Antony, S., Liu, H., Murthy, R.: Hive-a petabyte scale data warehouse using hadoop. In: Data Engineering (ICDE), 2010 IEEE 26th International Conference on. pp. 996–1005. IEEE (2010)
44. Torres, M.: Prototype of Stock Prediction System based on Twitter Emotion and Sentiment Analysis. Master's thesis, ETSI Telecomunicación (July 2014)
45. Warr, W.A.: Scientific workflow systems: Pipeline pilot and knime. Journal of computer-aided molecular design **26**(7), 801–804 (2012)
46. Westerski, A., Iglesias, C.A., Tapia, F.: Linked opinions: Describing sentiments on the structured web of data. In: Proceedings of the Fourth International Workshop on Social Data on the Web (SDoW2011). pp. 21–32. CEUR (Oct 2011)
47. WHATWG (Apple, Google, Mozilla, Microsoft): Html living standard. Tech. rep., W3C (july 2018)
48. White, T.: Hadoop: The definitive guide. " O'Reilly Media, Inc." (2012)
49. Wilde, E., Duerst, M.: URI Fragment Identifiers for the text/plain Media Type (Apr 2008)
50. Zaharia, M., Xin, R.S., Wendell, P., Das, T., Armbrust, M., Dave, A., Meng, X., Rosen, J., Venkataraman, S., Franklin, M.J., et al.: Apache spark: a unified engine for big data processing. Communications of the ACM **59**(11), 56–65 (2016)

A.2.2 An Emotion Aware Task Automation Architecture Based on Semantic Technologies for Smart Offices

Title	An Emotion Aware Task Automation Architecture Based on Semantic Technologies for Smart Offices
Authors	Muñoz López, Sergio and Araque, Oscar and Sánchez-Rada, J. Fernando and Iglesias, Carlos A.
Journal	Sensors
Impact factor	JCR 2018 Q1 (3.031)
ISSN	1424-8220
Publisher	
Volume	18
Year	2018
Keywords	ambient intelligence, emotion regulation, semantic technologies, smart office, task automation
Pages	1499
Online	http://www.mdpi.com/1424-8220/18/5/1499
Abstract	

sensors

MDPI

Article

An Emotion Aware Task Automation Architecture Based on Semantic Technologies for Smart Offices

Sergio Muñoz *, Oscar Araque, J. Fernando Sánchez-Rada and Carlos A. Iglesias

Intelligent Systems Group, Universidad Politécnica de Madrid, 28040 Madrid, Spain; o.araque@upm.es (O.A.); jf.sanchez@upm.es (J.F.S.-R.); carlosangel.iglesias@upm.es (C.A.I.)

* Correspondence: sergio.munoz@upm.es

Received: 19 March 2018; Accepted: 8 May 2018; Published: 10 May 2018

Abstract: The evolution of the Internet of Things leads to new opportunities for the contemporary notion of smart offices, where employees can benefit from automation to maximize their productivity and performance. However, although extensive research has been dedicated to analyze the impact of workers' emotions on their job performance, there is still a lack of pervasive environments that take into account emotional behaviour. In addition, integrating new components in smart environments is not straightforward. To face these challenges, this article proposes an architecture for emotion aware automation platforms based on semantic event-driven rules to automate the adaptation of the workplace to the employee's needs. The main contributions of this paper are: (i) the design of an emotion aware automation platform architecture for smart offices; (ii) the semantic modelling of the system; and (iii) the implementation and evaluation of the proposed architecture in a real scenario.

Keywords: ambient intelligence; smart office; emotion regulation; task automation; semantic technologies

1. Introduction

The emergence of Internet of Things (IoT) opens endless possibilities for the Information and Communication Technologies (ICT) sector, allowing new services and applications to leverage the interconnection of physical and virtual realms [1]. One of these opportunities is the application of Ambient Intelligence (AmI) principles to the workplace, which results in the notion of smart offices. Smart offices can be defined as "*workplaces that proactively, but sensibly, support people in their daily work*" [2].

A large body of research has been carried out on the impact that emotions have on decision making [3], health [4], emergencies [5] and working life [6]. This states the importance of recognizing and processing the emotions of people in intelligent environments. Particularly in the workplace, emotions play a key role, since the emotional state of workers directly affects other workers [7] and, consequently, company business. The application of emotion aware technologies to IoT environments entails a quantitative improvement in the workers' quality of life, since it allows the environment to be adaptive to these emotions and, therefore, to human needs [8]. In addition, this improvement in worker quality of life directly affects company performance and productivity [9].

Emotion Aware AmI (AmE) extends the notion of intelligent environments to detect, process and adapt intelligent environments to users' emotional state, exploiting theories from psychology and social sciences for the analysis of human emotional context. Considering emotions in the user context can improve customization of services in AmI scenarios and help users to improve their emotional intelligence [10]. However, emotion technologies are rarely addressed within AmI systems and have been frequently ignored [10,11].

A popular approach to interconnect and personalize both IoT and Internet services is the use of Event-Condition-Action (ECA) rules, also known as trigger–action rules [12]. Several now prominent

websites, mobile and desktop applications feature this rule-based task automation model, such as IFTTT (https://ifttt.com/) or Zapier (https://zapier.com/). These systems, so-called Task Automation Services (TASs) [13], are typically web platforms or smartphone applications, which provide an intuitive visual programming environment where inexperienced users seamlessly create and manage their own automations. Although some of these works have been applied to smart environments [14,15], these systems have not been applied yet for regulating users' emotions in emotion aware environments.

This work proposes a solution that consists in an emotion aware automation platform that enables the automated adaption of smart office environments to the employee's needs. This platform allows workers to easily create and configure their own automation rules, resulting in a significant improvement of their productivity and performance. A semantic model for the emotion aware TASs based on the Evented WEb (EWE) [13] ontology is also proposed, which enables data interoperability and automation portability, and facilitates the integration between tools in large environments. Moreover, several sensors and actuators have been integrated in the system as a source of ambient data or as action performers which interact with the environment. In this way, the design of an emotion aware automation platform architecture for smart offices is the main contribution of this paper, as well as the semantic modelling of the system and its implementation and validation in a real scenario.

The rest of this paper is organized as follows. Firstly, an overview about the related work in smart offices, emotion regulation and semantic technologies is given in Section 2. Section 3 presents the semantic modelling of the system, describing different ontologies and vocabularies which have been used and the relationships between them. Then, Section 4 describes the reference architecture of the proposed emotional aware automation platform, describing the main components and modules as well as its implementation. Section 5 describes the evaluation of the system in a real scenario. Finally, the conclusions drawn from this work are described in Section 6.

2. Background

This section describes the background and related work for the architecture proposed in this paper. First, an overview of related work in AmE and specifically in smart offices is given in Sections 2.1 and 2.2, respectively. Then, the main technologies involved in emotion recognition and regulation are described in Sections 2.3 and 2.4. Finally, Section 2.5 gives an overview of the state of art regarding to semantic technologies.

2.1. Emotion Aware AmI (AmE)

The term AmE was coined by Zhou et al. [16]. AmE is "*a kind of AmI environment facilitating human emotion experiences by providing people with proper emotion services instantly*". This notion aims at fostering the development of emotion-aware services in pervasive AmI environments.

AmE are usually structured in three building blocks [10,17]: emotion sensing, emotion analysis and emotion services or applications.

Emotion sensing is the process of gathering affective data using sensors or auto-reporting techniques. There exists many potential sensor sources, including speech, video, mobile data [18], textual and physiological and biological signals. An interesting research for multimodal sensing in real-time is described in [19]. Then, the **Emotion analysis** module applies emotion recognition techniques (Section 2.4) to classify emotions according to emotion models, being the most popular the categorical and dimensional ones and optionally express the result in an emotion expression language (Section 2.5). **Emotion services or applications** exploit the identified emotions in order to improve user's life. The main applications are [17] emotion awareness and sharing to improve health and mental well-being to encourage social change [20], mental health tracking [21], behaviour change support [22], urban affective sensing to understand the affective relationships of people towards specific places [23] and emotion regulation [24] (Section 2.4).

The adaptation of AmI frameworks to AmE presents a number of challenges because of the multimodal nature of potential emotion sensors and the need for reducing ambiguity of

emotion multimodal sources using fusion techniques. In addition, different emotion models are usually used depending on the nature of the emotion sources and the intended application. According to [25], most existing pervasive systems do not consider a multi-modal emotion-aware approach. As previously mentioned, despite the mushrooming of IoT, there are only few experiences in the development of AmE environments that take into account emotional behaviour, and most of them describe prototypes or proofs of concept [10,11,25–29].

From these works, emotion sensing has been addressed using emotion sources such as speech [25,26,29], text [10], video facial and body expression recognition [24] and physiological signals [24]. Few works have addressed the problem of emotion fusion in AmI [24] where a neural multimodal fusion mechanism is proposed. With regard to regulation techniques, fuzzy [24,29] and neurofuzzy controllers [11] have been proposed. Finally, the fields of application have been smart health [24], intelligent classroom [29] and agent-based group decision making [28].

Even though some of the works mention a semantic modelling approach [10], the reviewed approaches propose or use a semantic schema for modelling emotions. Moreover, the lack of semantic modelling of the AmI platform is challenging for integrating new sensors and adapt them to new scenarios. In addition, these works follow a model of full and transparent automation which could leave users feeling out of control [30], without supporting personalization.

2.2. Smart Offices

Although several definitions for smart offices are given in different works [2,31,32], all of them agree in considering a smart office as an environment that supports workers on their daily tasks. These systems use the information collected by different sensors to reason about the environment, and trigger actions which adapt the environment to users' needs by mean of actuators.

Smart offices should be aligned to the business objectives of the enterprise, and should enable a productive environment that maximizes employee satisfaction and company performance. Thus, smart offices should manage efficiently and proactively the IoT infrastructure deployed in the workplace as well as the enterprise systems. Moreover, smart offices should be able to interact with smartphones and help employees to conciliate their personal and professional communications [33].

Focusing on existing solutions whose main goal is the improvement of workers' comfort at the office, Shigeta et al. [34] proposed a smart office system that uses a variety of input devices (such as camera and blood flow sensor) in order to recognize workers' mental and physiological states, and adapts the environment by mean of output devices (such as variable colour light, speaker or aroma generator) for improving workers' comfort. In addition, HealthyOffice [35] deals with a novel mood recognition framework that is able to identify five intensity levels for eight different types of moods, using Silmee TM device to capture physiological and accelerometer data. Li [36] proposed the design of a smart office system that involves the control of heating, illuminating, lighting, ventilating and reconfiguration of the multi-office and the meeting room. With regard to activity recognition, Jalal et al. [37] proposed a depth-based life logging human activity recognition system designed to recognize the daily activities of elderly people, turning these environments into an intelligent space. These works are clear examples of using smart office solutions for improving quality of life, and they propose systems able to perform environment adaption based on users' mental state.

Kumar et al. [38] proposed a semantic policy adaptation technique and its applications in the context of smart building setups. It allows users of an application to share and reuse semantic policies amongst them-selves, based on the concept of context interdependency. Alirezaie et al. [39] presented a framework for smart homes able to perform context activity recognition, and proposed also a semantic model for smart homes. With regard to the use of semantic technologies in the smart office context, Coronato et al. [40] proposed a semantic context service that exploits semantic technologies to support smart offices. This service relies on ontologies and rules to classify several typologies of entities present in a smart office (such as services, devices and users) and to infer higher-level context

information from low-level information coming from positioning systems and sensors in the physical environments (such as lighting and sound level).

One of the first mentions of emotion sensor was in the form of affective wearables, by Picard et al. [41]. As for semantic emotion sensors, there is an initial work proposed by Gyrard et al. [42]. However, to the extent of our knowledge, there is no work in the literature that properly addresses the topics of emotion sensors and semantic modelling in a unified smart automation platform. This paper aims to fill this gap, proposing a semantic automation platform that also takes into account users' emotion.

2.3. Emotion Recognition

Over the last years, emotion detection represents a significant challenge that is gaining the attention of a great number of researchers. The main goal is the use of different inputs for carrying out the detection and identification of the emotional state of a subject. Emotion recognition opens endless possibilities as it has wide applications in several fields such as health, emergencies, working life, or commercial sector. The traditional approach of detecting emotions through questionnaires answered by the participants does not yield very efficient methods. That is the reason for focusing on automatic emotion detection using multimodal approaches (i.e., facial recognition, speech analysis and biometric data), as well as ensemble of different information sources from the same mode [43].

Algorithms to predict emotions based on facial expressions are mature and considered accurate. Currently, there are two main techniques to realize facial expression recognition depending on its way of extracting feature data: appearance-based features, or geometry-based features [44]. Both techniques have in common the extraction of some features from the images which are fed into a classification system, and differ mainly in the features extracted from the video images and the classification algorithm used [45]. Geometric based techniques find specific features such as the corners of the mouth, eyebrows, etc. and extracts emotional data from them. Otherwise, appearance based extraction techniques describe the texture of the face caused by expressions, and extract emotional data from skin changes [46].

Emotion recognition from speech analysis is an area that is gaining momentum in recent years [47]. Speech features are divided into for main categories: continuous features (pitch, energy, and formants), qualitative features (voice quality, harsh, and breathy), spectral features (Linear Predictive Coefficients (LPC) and Mel Frequency Cepstral Coefficients (MFCC)), and Teager energy operator-based features (TEO-FM-Var and TEO-Auto-Env) [48].

Physiological signals are another data source for recognizing people's emotions [49]. The idea of wearables that detect the wearer's affective state dates back to the early days of affective computing [41]. For example, skin conductance changes if the skin is sweaty, which is related to stress situations and other affects. Skin conductance is used as an indicator of arousal, to which it is correlated [50]. A low level of skin conductivity suggests low arousal level. Heart rate is also a physiological signal connected with emotions, as its variability increases with arousal. Generally, heart rate is higher for pleasant and low arousal stimuli compared to unpleasant and high arousal stimuli [50].

2.4. Emotion Regulation

Emotion regulation consists in the modification of processes involved in the generation or manifestation of emotion [51], and results an essential component of psychological well-being and successful social functioning. A popular approach to regulate emotions is the use of colour, music or controlled breathing [52,53].

Xin et al. [54,55] demonstrated that colour characteristics such as chroma, hue or lightness produce an impact on emotions. Based on these studies and on the assumption of the power of colour to change mood, Sokolova et al. [52] proposed the use of colour to regulate affect. Participants of this study indicated that pink, red, orange and yellow maximized their feeling of joy, while sadness correlates with dark brown and gray. Ortiz-García-Cervigón et al. [56] proposed an emotion regulation system

at home, using RGB LED strips that are adjustable in colour and intensity to control the ambience. This study reveals that warm colours are rated as more tensed, hot, and less preferable for lighting, while cold colours are rated as more pleasant.

With regard to music, several studies [57,58] show that listening to music influences mood and arousal. Van der Zwaag [59] found that listening to preferred music significantly improved performance on high cognitive demand tasks, suggesting that music increases efficiency for cognitive tasks. Therefore, it has been demonstrated that listening to music can influence regulation abilities, arousing certain feelings or helping to cope negative emotions [60]. In addition, it has been demonstrated that different types of music may have different demands on attention [61].

The commented studies show that the adaptation of ambient light colour and music are considerable solutions for regulating emotions in a smart office environment, as this adaptation may improve workers' mood and increase their productivity and efficiency.

2.5. Semantic Modelling

Semantic representation considerably improves interoperability and scalability of the system, as it provides a rich machine-readable format that can be understood, reasoned about, and reused.

To exchange information between independent systems, a set of common rules need to be established, such as expected formats, schemas and expected behaviour. These rules usually take the form of an API (application programming interface). In other words, systems need not only to define **what** they are exchanging (concepts and their relationship), but also **how** they represent this information (representation formats and models). Moreover, although these two aspects need to be in synchrony, they are not unambiguously coupled: knowing how data are encoded does not suffice to know what real concepts the refer to, and vice versa.

The semantic approach addresses this issue by replacing application-centric ad-hoc models and representation formats with a formal definition of the concepts and relationships. These definitions are known as ontologies or vocabularies. Each ontology typically represents one domain in detail, and they borrow concepts from one another whenever necessary [62]. Systems then use parts of several ontologies together to represent the whole breadth of their knowledge. Moreover, each concept and instance (entity) is unambiguously identified. Lastly, the protocols, languages, formats and conventions used to model, publish and exchange semantic information are standardized and well known (SPARQL, RDF, JSON-LD, etc.) [63–65].

This work merges two domains: rule-based systems and emotions. We will explore the different options for semantic representation in each domain.

There are plenty of options for modelling and implementing rule-based knowledge, such as RuleML [66], Semantic Web Rule Language (SWRL) [67], Rule Interchange Format (RIF) [68], SPARQL Inferencing Notation (SPIN) [69] and Notation 3 (N3) Logic [70].

EWE [13] is a vocabulary designed to model, in a descriptive approach, the most significant aspects of Task Automation Service (TAS). It has been designed after analyzing some of the most relevant TASs [71] (such as Ifttt, Zapier, Onx, etc.) and provides a common model to define and describe them. Based on a number of identified perspectives (privacy, input/output, configurability, communication, discovery and integration), the main elements of the ontology have been defined, and formalized in an ontology. Moreover, extensive experiments have been developed to transform the automation of these systems into the proposed ontology. Regarding inferences, EWE is based on OWL2 classes and there are implementations of EWE using a SPIN Engine (TopBraid (https://www.w3.org/2001/sw/wiki/TopBraid)) and N3 Logic (EYE (http://eulersharp.sourceforge.net/)).

Four major classes make up the core of EWE: *Channel*, *Event*, *Action* and *Rule*. The class *Channel* defines individuals that either generate *Events*, provide *Actions*, or both. In the smart office context, sensors and actuators such as an emotion detector or a smart light are described as channels, which produce events or provide actions. The class *Event* defines a particular occurrence of a process, and allows users to describe under which conditions should rules be triggered. These conditions

are the configuration parameters, and are modelled as input parameters. Event individuals are generated by a certain channel, and usually provide additional details. These additional details are modelled as output parameters, and can be used within rules to customize actions. The recognition of sadness generated by the emotion detector sensor is an example of entity that belongs to this class. The class *Action* defines an operation provided by a channel that is triggered under some conditions. Actions provides effects whose nature depends on itself, and can be configured to react according to the data collected from an event by means of input parameters. Following the smart office context mentioned above, to change the light colour is an example of action generated by the smart light channel. Finally, the class *Rule* defines an *ECA*, triggered by an event that produces the execution of an action. An example of rule is: "*If sadness is detected, then change the light colour*".

There are also different options for emotion representation. EmotionML [72] is one of the most notable general-purpose emotion annotation and representation languages that offers twelve vocabularies for categories, appraisals, dimensions and action tendencies. However, as shown in previous works [73], the options for semantic representation are limited to a few options, among which we highlight the Human Emotion Ontology (HEO) [74], and Onyx [73], a publicly available ontology for emotion representation. Among these two options, we chose Onyx for several reasons: it is compatible with EmotionML; it tightly integrates with the Provenance Ontology [75], which gives us the ability to reason about the origin of data annotations; and it provides a meta-model for emotions, which enables anyone to publish a new emotion model of their own while remaining semantically valid, thus enabling the separation of representation and psychological models. The latter is of great importance, given the lack of a standard model for emotions. In EmotionML, emotion models are also separated from the language definition. A set of commonly used models is included as part of the vocabularies for EmotionML [76], all of which are included in Onyx.

Moreover, the Onyx model provides a model for emotion conversion, and a set of existing conversions between well known models. Including conversion as part of the model enables the integration of data using different models. Two examples of this would be working with emotion readings from different providers, or fusing information from different modalities (e.g., text and audio), which typically use different models. It also eases a potential migration to a different model in the future.

In addition, Onyx has been extended to cover multimodal annotations [77,78]. Lastly, the Onyx model has been embraced by several projects and promoted by members of the Linked Data Models for Emotion and Sentiment Analysis W3C Community Group [79].

There are three main concepts in the Onyx ontology that are worth explaining, as they are used in the examples in following sections. They are: *Emotion*, *EmotionAnalysis* and *EmotionSet*. They relate to each other in the following way: an EmotionAnalysis process annotates a given entity (e.g., a piece of text or a video segment) with an EmotionSet, and an EmotionSet is in turn comprised of one or more Emotions. Due to the provenance information, it is possible to track the EmotionAnalysis that generated the annotation.

3. Semantic Modelling for the Smart Office Environment

With the purpose of applying a semantic layer to the emotion aware automation system, several vocabularies and relationships between ontologies have been designed. This enables the semantic modelling of all entities in the smart office environment. Figure 1 shows the relationships between the used ontologies described above.

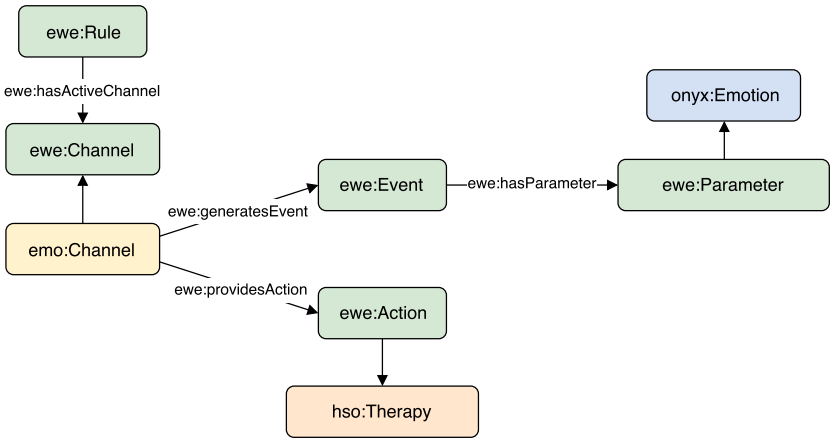


Figure 1. Main classes of the ontologies involved in the semantic modelling.

Automation rules (*ewe:Rule*) are modelled using EWE ontology [13], which presents them in event-condition-action form. Events (*ewe:Event*) and actions (*ewe:Action*) are generated by certain channels. In the proposed architecture, there are different channels that either generate events, provide actions, or both. The class *ewe:Channel* has been subclassed to provide an emotional channel class (*emo:Channel*), which is responsible for generating events and actions related to the emotion recognition and regulation. From this class, the channels *emo:EmotionSensor* and *emo:EmotionRegulator* have been defined. The former is responsible for generating events related to the emotion detection, while the later is responsible for providing certain actions that have the purpose of regulating the emotion. These two classes group all sensors and actuators able to detect or regulate emotions, but should be subclassed by classes representing each device concretely. In addition, events and actions may have parameters. The *emo:EmotionDetected* event has as Parameter the detected emotion. Emotions are modelled using Onyx [73], as described in Section 2.5, so the parameter must subclass *onyx:Emotion*.

The *emo:EmotionRegulator* channel can be subclassed for defining a *SmartSpeaker* or a *SmartLight*, able to provide actions to regulate the emotion such as *emo:PlayRelaxingMusic* or *emo:ChangeAmbientColor*, respectively. The action of playing relaxing music has as parameter (*ewe:Parameter*) the song to be played, while the action of change ambient colour has as parameter the colour to which the light must change. In addition, all these actions are also represented as therapies using Human Stress Ontology (HSO) ontology [80], so *hso:Therapy* has been subclassed. To give a better idea of how specific Channels, Events and Actions have been modelled; Table 1 shows the commented example written in Notation3, describing all its actions with their corresponding parameters.

An example of event and action instances with grounded parameters, which are based on the concepts defined in the listing given above, is presented in Table 2. This table describes the definition of sadness and the actions of playing music and changing ambient colour.

Similarly, automation rules are described using the punning mechanism to attach classes to properties of Rule instances. In the example shown in Table 3, the rule instance describes a rule that is triggered by the event of *sad emotion detection* and produces the action of *changing ambient colour to green* (both defined in Table 2).

Table 1. Semantic representation of Emotion Regulator channel written in Notation3.

```
emo:SmartSpeaker a owl:Class ;
    rdfs:label ''Smart Speaker'' ;
    rdfs:comment ''This channel represents a smart speaker.'' ;
    rdfs:subClassOf emo:EmotionRegulator .

emo:PlayRelaxingMusic a owl:Class ;
    rdfs:label ''Play relaxing music'' ;
    rdfs:comment ''This action will play relaxing music.'' ;
    rdfs:subclassOf ewe:Action ;
    rdfs:subclassOf hso:Therapy ;
    rdfs:domain emo:SmartSpeaker .

emo:SmartLight a owl:Class ;
    rdfs:label ''Smart Light'' ;
    rdfs:comment ''This channel represents a smart light.'' ;
    rdfs:subClassOf emo:EmotionRegulator .

emo:ChangeAmbientColor a owl:Class ;
    rdfs:label ''Change ambient color'' ;
    rdfs:comment ''This action will change ambient color.'' ;
    rdfs:subclassOf ewe:Action ;
    rdfs:subclassOf hso:Therapy ;
    rdfs:domain emo:SmartLight .
```

Table 2. Event and action instances.

```
:sad-emotion-detected a emo:EmotionDetected ;
    ewe:hasEmotion onyx:sadness .

:play-music a emo:PlayRelaxingMusic ;
    ewe:hasSong ''the title of the song to be played'' .

:change-ambient-color-green a emo:ChangeAmbientColor ;
    ewe:hasColor dbpedia:Green .
```

Table 3. Rule instance.

```
:regulate-stress a ewe:Rule ;
    dcterms:title    ''Stress regulation rule''^xsd:string ;
    ewe:triggeredByEvent    :sad-emotion-detected ;
    ewe:firesAction         :change-ambient-color-greenr .
```

4. Emotion Aware Task Automation Platform Architecture

The proposed architecture was designed based on the reference architecture for TAS [81],which was extended to enable emotion awareness. The system is divided into two main blocks: **emotional context recognizer** and **emotion aware task automation server**, as shown in Figure 2. Emotional context recognizer aims to detect and recognize users' emotions and information related to context or Internet services and send them to the automation platform to trigger the corresponding actions.

The automation system that receives these data is a semantic event-driven platform that receives events from several sources and performs the corresponding actions. In addition, it provides several functions for automating tasks by means of semantic rules and integrates different devices and services.

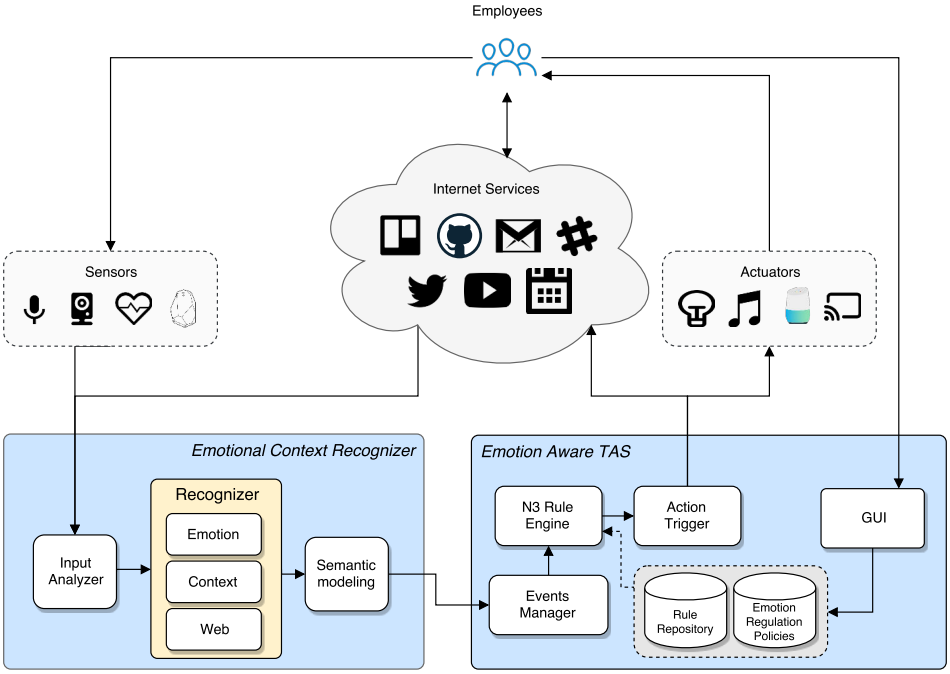


Figure 2. Emotion Aware Automation Platform Architecture.

4.1. Emotional Context Recognizer

The emotional context recognizer block is responsible for detecting users' emotions and contextual events, encoding emotions and events using semantic technologies, and sending these data to the automation platform, where they are evaluated. The block consists of three main modules: input analyzer, recognizer and semantic modelling. In addition, each module is composed of multiple independent and interchangeable sub-modules that provide the required functions, with the purpose of making the system easy to handle.

The input analyzer receives data from sensors involved in emotion and context recognition (such as camera, microphone, wearables or Internet services) and its pre-processing. With this purpose, the input analyzer is connected with the mentioned sensors, and the received data are sent to the recognizer module. The recognizer module receives data captured by the input analyzer. It consists in a pipeline with several submodules that perform different analysis depending on the source of the information. In the proposed architecture, there are three sub-modules: emotion recognizer, context recognizer and web recognizer. The emotion recognizer module provides functions for extracting emotions by means of real time recognition of facial expression, speech and text analysis, and biometric data monitoring; the context recognizer provides functions for extracting context data from sensors (e.g., temperature amd humidity); and the web recognizer provides functions for extracting information from Internet services. Once data have been extracted, they are sent to the semantic modelling module. The main role of semantic modelling is the application of a semantic layer (as described in Section 3), generating the semantic events and sending them to the automation platform.

4.2. Emotion Aware Task Automation Server

The automation block consists in an intelligent automation platform based on semantic ECA rules. The main goal is to enable semantic rule automation in a smart environment, allowing the user to configure custom automation rules or to import rules created by other users in an easy way. In addition, it provides integration with several devices and services such as a smart TV, Twitter, Github, etc., as well as an easy way for carrying out new integrations.

The platform handles events coming from different sources and triggers accordingly the corresponding actions generated by the rule engine. In addition, it includes all the functions for managing automation rules and the repositories where rules are stored, as well as functions for creating and editing channels. With this purpose, the developed platform is able to connect with several channels for receiving events, evaluating them together with stored rules and performing the corresponding actions.

To enable the configuration and management of automation rules, the platform provides a *graphical user interface* (GUI) where users can easily create, remove or edit rules. The GUI connects with the rule administration module, which is responsible for handling the corresponding changes in the repositories. There are two repositories in the platform: *rule repository*, where information about rules and channels is stored; and *emotion regulation policies repository*. The policies are sets of rules which aim to regulate the emotion intensity in different contexts. In the smart office context proposed, they are intended to regulate negative emotions to maximize productivity. The rules may be aimed towards automating aspects such as: ambient conditionsto improve the workers' comfort; work related tasks to improve efficiency; or the rules could adjust work conditions to improve productivity. Some examples of these rules are presented below:

(a) *If stress level of a worker is too high, then reduce his/her task number*. When a very high stress level in a worker has been detected, this rule proposes reducing his/her workload to achieve that his/her stress level falls and his/her productivity rises.
(b) *If temperature rises above 30 °C, then turn on the air conditioning*. To work at high level of temperatures may result in workers' stress, so this rule proposes to automatically control this temperature in order to prevent high levels of stress.
(c) *If average stress level of workers is too high, then play relaxing music*. If most workers have a high stress value, the company productivity will significantly fall. Thus, this rule proposes to play relaxing music in order to reduce the stress level of workers.

In addition, the company human resources department may implement their own emotion regulation policies to adjust the system to their own context. The system adapts rules based on channel description. Rule adaptation is based on identifying if the smart environment includes the channels used by a certain rule. The system detects available channels of the same channel class used by the rule and request confirmation from the user to included the "adapted rule". This enables the adaptation of rules to different channel providers, which can be physical sensors (i.e., different beacons) or internet services (i.e., Gmail and Hotmail). The EWE ontology allows us this adaptation by mean of OWL2 punning mechanism for attaching properties to channels [13].

With regards to event reception, these are captured by the *events manager* module, which sends them to the rule engine to be evaluated along with the stored rules. The rule engine module is a semantic engine reasoner [82] based on an ontology model. It is responsible for the reception of events from the *events manager* and the load of rules that are stored in the repository. When a new event is captured and the available rules are loaded, the reasoner runs the ontology model inferences and the actions based on the incoming events and the automation rules are drawn. These actions are sent to the *action trigger*, which connects to the corresponding channels to perform the actions.

The semantic integration of sensors and services is done based on the notion of adapters [83,84], which interact with both sensors and internet services, providing a semantic output. Adapters, as well as mobile clients, are connected to the rule engine through Crossbar.io (https://crossbar.io/), and IoT

Middleware that provides both REST-through Web Application Messaging Protocol (WAMP)- and Message Queuing Telemetry Transport (MQTT) interfaces.

Finally, the implementation of this architecture, called EWETasker, was made using PHP for the server, HTML/JavaScript for the web client (including the GUI), and Android SDK for a mobile client. The implementation was based on N3 technology and EYE reasoning engine (http://n3.restdesc.org/). Several sensors and services have already been integrated into EWETasker suitable for the smart office use case. In particular, EWETasker supports indoor and temperature sensors (Estimote bluetooth beacons (https://estimote.com)), smart object sensor (Estimote bluetooth stickers), electronic door control based on Arduino, video emotion sensors (based on Emotion Research Lab), social network emotion sensor (Twitter), and mobile-phone sensors (Bluetooth, location, wifi, etc.). With regards to corporate services, several services oriented to software consultancy firms have been integrated for collaboration (Twitter, GMail, Google Calendar, and Telegram) and software development (Restyaboard Scrum board (http://www.restya.com), GitHub (https://github.com) and Slack (https://slack.com)).

5. Experimentation

As already stated, the main experimental contribution of this work was the design and implementation of an emotion aware automation platform for smart offices. In this way, we raised four hypotheses regarding the effectiveness of the proposed system:

- H1: The use of the proposed platform regulates the emotional state of a user that is under stressful conditions.
- H2: The actions taken by the proposed platform do not disturb the workflow of the user.
- H3: The use of the proposed system improves user performance.
- H4: The use of the system increases user satisfaction.

To evaluate the proposed system with respect to these hypotheses, an experiment with real users was performed. For this experiment, a prototype of the proposed system was deployed, which includes the following components. The emotion of the participants was detected from a webcam feed, which feeds a video-based emotion recognizer. As for the semantic layers of the system, the events manager, rule engine and action trigger were fully deployed. Finally, the actuators implemented both hearing and visual signals using a variety of devices. Detailed information on materials is given in Section 5.2. This section covers the design, results and conclusions drawn from the experiment, focusing on its scope.

5.1. Participants

The experiment included 28 participants. Their ages ranged from 18 to 28 years, all of them university students with technical background, of both genders. All of them were unaware of this work, and no information regarding the nature of the experiment was given to the participants beforehand. Since the proposed system is primarily oriented to technical work positions, this selection is oriented to validate the system with participants that are currently working in technical environments, or will in the future.

5.2. Materials

The material used for this experiment is varied, as the proposed automation system needs several devices to properly function. Regarding the deployment of the automation system, the TAS ran in a commodity desktop computer, with sufficient CPU and memory for its execution. The same environment was prepared for the emotion recognizer system. For the sensors and actuators, the following were used:

- Emotion Research software (https://emotionresearchlab.com/). This module provides facial mood detection and emotional metrics that are fed to the automation system. This module is

an implementation that performs emotion classification in two main steps: (i) it makes use of Histogram of Oriented Gradients (HOG) features that are used to train with a SVM classifier in order to localize face position in the image; and (ii) the second step consists in a normalization process of the face image, followed by a Multilayer Perceptron that implements the emotion classification. Emotion Research reports 98% accuracy in emotion recognition tasks.
- A camera (Gucee HD92) feeds the video to the emotion recognizer submodule.
- Room lighting (WS2812B LED strip controlled by WeMos ESP8266 board) is used as an actuator on the light level of the room, with the possibility of using several lighting patterns.
- Google Chromecast [85] transmits content in a local computer network.
- LG TV 49UJ651V is used for displaying images.
- Google Home is used for communicating with the user. In this experiment, the system can formulate recommendations to the user.

Participants accessed the web HTML-based interface using a desktop computer with the Firefox browser (https://www.mozilla.org/en-US/firefox/desktop/).

5.3. Procedure

During the experiment, each participant performed a task intended to keep the participant busy for approximately 10 min. This task consisted in answering a series of basic math related questions that were presented to the participant via a web interface (e.g., "Solve $24 \cdot 60 \cdot 60$"). We used a set of 20 questions of similar difficulty that have been designed so that any participant can answer them within 30 s. The use of a web-based interface allowed us to programmatically perform the session, and to record metrics associated with the experiment.

The workflow of the experiment is as follows. Each participant's session is divided into two parts. In each part of the session half of the task questions are sequentially prompted to the participant by the examiner system. Simultaneously, the automation system is fed with the information provided by the different sensors that are continually monitoring the participant emotional state. The experiment finishes when all the questions have been answered. In addition, a questionnaire is given to the participants just after the sessions concludes. These questions are oriented to offer the participant's view of the system. The raised questions are summarized in Table 4. Questions Q2, Q3, Q4 and Q5 are asked twice, once in regard to the no automation part, and the other time in relation to the part with the automation enabled. Questions Q1 and Q2 are designed so that a check of internal consistency is possible; as, if results from these two questions were to disagree, the experiment would be invalid [86].

Table 4. Questions raised to the participants at the end of the session.

No.	Hypothesis	Question Formulation
Q1	H1, H2	In which section have you been more relaxed?
Q2	H1, H2	What is your comfort level towards the environment?
Q3	H3	Do you think the environment's state has been of help during the completion of the task?
Q4	H4	Would you consider beneficial to work in this environment?
Q5	H4	What is your overall satisfaction with relation to the environment?

The workflow of the system in the context of the experiment is as follows. While the participant is performing the task, the emotion sensor is continuously monitoring the participant's emotional state. The emotion sensor uses the camera as information input, while the Google Home is used when the user communicates with the system. This emotion-aware data are sent to the TAS, which allows the system the have continuous reports. The TAS receives, processes, and forwards these events to the N3 rule engine. Programmed rules are configured to detect changes in the participant emotional state, acting accordingly. As an example, a shift of emotion, such as the change from happy to sad,

is detected by the rule engine which triggers the relaxation actions. If a certain emotion regulation rule is activated, the corresponding action is then triggered through the communication to the action trigger module, which causes the related actuators to start its functioning. The configured actions are aimed at relaxing and regulating the emotion of the participant, so that the performance in the experiment task is improved, as well as the user satisfaction. The actions configured for this experiment are: (i) relaxation recommendations done by the Google Home, such as a recommendation to take a brief walk for two minutes; (ii) lighting patterns using coloured lights that slowly change its intensity and colour; and (iii) relaxing imagery and music that are shown to the user via the TV. A diagram of this deployment is shown in Figure 3.

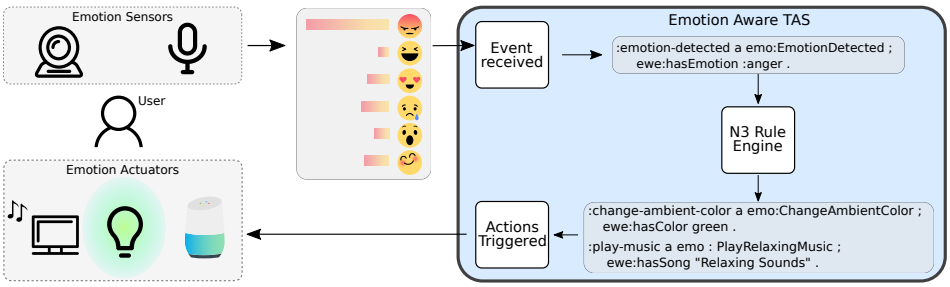


Figure 3. Deployment for the experiment.

While the participants are performing the proposed task, the actions of the automation system are controlled. During half of each session, the automation is deactivated, while, during the other half, the action module is enabled. With this, we can control the environmental changes performed by the automation system, allowing its adaptation at will.

Another interesting aspect that could be included is the integration of learning policies based on employee's emotional state. A related work that models learning policies and their integration with Enterprise Linked Data is detailed in [87].

5.4. Design

The experiment was a within-subject design. As previously stated, the controlled factor is the use of the automation system, which has two levels, activated and not activated. The automation use factor is counterbalanced using a Latin square so that the participants are divided into two groups. One group performs the first half of the session without the automation system, while, for the second half of the session, the system is used. The other group performs the task inversely.

5.5. Results and Discussion

To tackle Hypothesis 1 and Hypothesis 2, Questions 1 and 2 were analyzed. Regarding Question 1, 18 respondents declared that the section with the adaptation system enabled was the most relaxing for them. In contrast, seven users claimed that for them the most relaxing section of the experiment was that without the adaptation system. The results from Question 1 suggest that users prefer to use the adaptation system, although it seems that this is not the case for all the users. Regarding the Question 2, results show that the average in the adaptation part (3.5) is higher than with no adaptation whatsoever (2.5), as shown in Figure 4. An ANOVA analysis shows that this difference is statistically significant ($p = 0.015 < 0.05$). These results support H1 and H2, concluding that users feel more inclined to use the adaptation system rather than performing the task without adaptation.

Following, Question 3 addressed Hypothesis 3. The analysis of the results of this question reveals that users point higher the usefulness of the environment adaptation for the completion of the task, as shown in Figure 4. While the average for the adaptation section is 3.93, it is 2.07 for the no adaptation

part. Through ANOVA, we see that this difference is considerably significant ($p = 2.96 \times 10^{-6} < 0.05$). As expected, Hypothesis 3 receives experimental support, indicating that the use of the automation system can improve the performance of the user in a certain task, as perceived by the users.

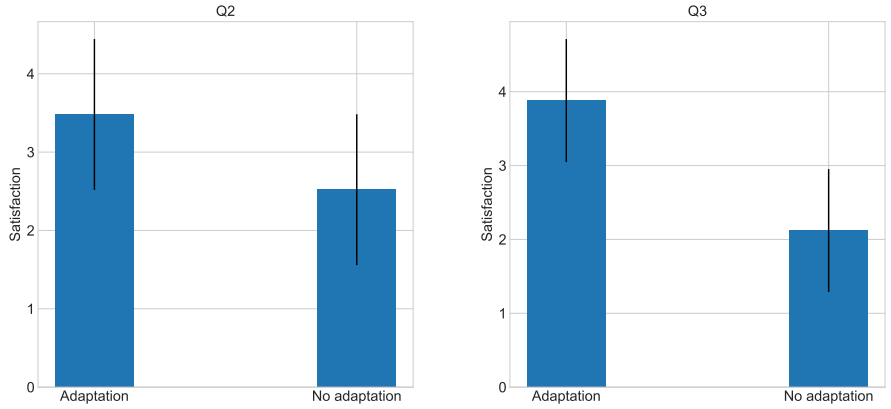


Figure 4. Results for Q2 and Q3.

In relation to Hypothesis 4, both Questions 4 and 5 are aimed to check its validity. As can be seen in Figure 5, users consider more beneficial to work with the adaptation system enabled. The average measure for the adaptation is 3.78, while the no adaptation environment is considered lower on average, with 2.21. Once again, the ANOVA test outputs a significant difference between the two types of environment ($p = 0.0002 < 0.05$). With regard to Question 5, the average for the satisfaction with the adapted environment is 3.83; in contrast, the satisfaction with the no adaptation environment is 2.17, as shown in Figure 5. After performing an ANOVA test, we see that this difference is greatly significant ($p = 1.02 \times 10^{-10} < 0.05$). Attending to this, users seem to consider the adaptation system for their personal workspace, and at the same time, they exhibit a higher satisfaction with an adapted work environment. These data indicate that Hypothesis 4 is true, and that users positively consider the use of the adaptation system.

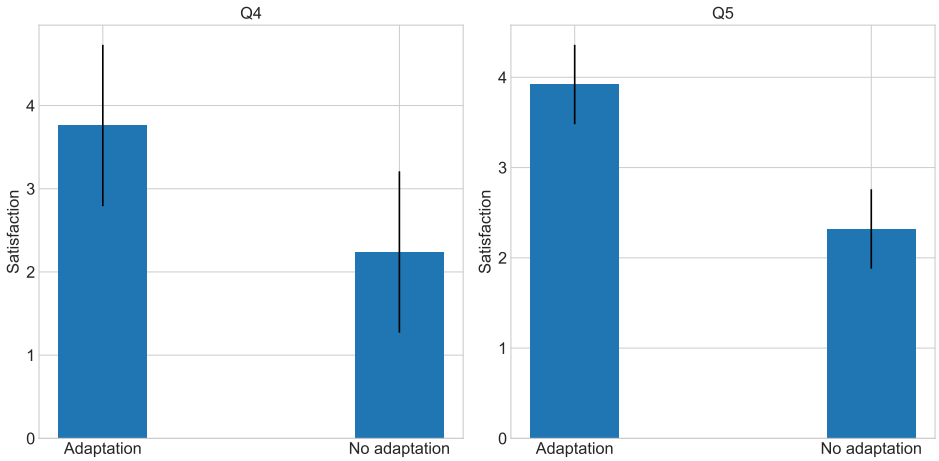


Figure 5. Results for Q4 and Q5.

6. Conclusions and Outlook

This paper presents the architecture of an emotion aware automation platform based on semantic event-driven rules, to enable the automated adaption of the workplaces to the need of the employees. The proposed architecture allows users to configure their own automation rules based on their emotions to regulate these emotions and improve their wellbeing and productivity. In addition, the architecture is based on semantic event-driven rules, so this article also describes the modelling of all components of the system, thus enabling data interoperability and portability of automations. Finally, the system was implemented and evaluated in a real scenario.

Through the experimentation, we verified a set of hypotheses. In summary: (i) using the proposed automation system helps to regulate the emotional state of users; (ii) adaptations of the automation system do not interrupt the workflow of users; (iii) the proposed system improves user performance in a work environment; and, finally, (iv) the system increases user satisfaction. These results encourage the use and improvement of this kind of automation systems, as they seem to provide users with a number of advantages, such as regulation of stress and emotions, and personalized work spaces.

As future work, there are many lines that can be followed. One of these lines is the application of the proposed system to other scenarios different from smart offices. The high scalability offered by the developed system facilitates the extension of both the architecture and the developed tools with the purpose of giving a more solid solution to a wider range of scenarios. Currently, we are working on its application to e-learning and e-commerce scenarios. In addition, another line of future work is the recognition of the activity, as it is useful to know the activity related to the detected emotion of the user.

Furthermore, we also plan to develop a social simulator system based on emotional agents to simplify the test environment. This system will enable testing different configurations and automations of the smart environment before implementing them in a real scenario, resulting in an important reduction of costs and efforts in the implementation.

Author Contributions: S.M. and C.A.I. originally conceived the idea; S.M. designed and developed the system; S.M., O.A. and J.F.S. designed the experiments; S.M. and O.A. performed the experiments; O.A. analyzed the data; O.A. contributed analysis tools; and all authors contributed to the writing of the paper.

Funding: This work was funded by Ministerio de Economía y Competitividad under the R&D projects SEMOLA (TEC2015-68284-R) and EmoSpaces (RTC-2016-5053-7), and by the Regional Government of Madrid through the project MOSI-AGIL-CM (grant P2013/ICE-3019, co-funded by EU Structural Funds FSE and FEDER).

Acknowledgments: The authors express their gratitude to Emotion Research Lab team for sharing their emotion recognition product for this research work. This work is supported by the Spanish Ministry of Economy and Competitiveness under the R&D projects SEMOLA (TEC2015-68284-R) and EmoSpaces (RTC-2016-5053-7), by the Regional Government of Madrid through the project MOSI-AGIL-CM (grant P2013/ICE-3019, co-funded by EU Structural Funds FSE and FEDER).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Miorandi, D.; Sicari, S.; Pellegrini, F.D.; Chlamtac, I. Internet of things: Vision, applications and research challenges. *Ad Hoc Netw.* **2012**, *10*, 1497–1516. [CrossRef]
2. Augusto, J.C. *Ambient Intelligence: The Confluence of Ubiquitous/Pervasive Computing and Artificial Intelligence*; Springer: London, UK; pp. 213–234.
3. Gutnik, L.A.; Hakimzada, A.F.; Yoskowitz, N.A.; Patel, V.L. The role of emotion in decision-making: A cognitive neuroeconomic approach towards understanding sexual risk behavior. *J. Biomed. Inform.* **2006**, *39*, 720–736. [CrossRef] [PubMed]

4. Kok, B.E.; Coffey, K.A.; Cohn, M.A.; Catalino, L.I.; Vacharkulksemsuk, T.; Algoe, S.B.; Brantley, M.; Fredrickson, B.L. How Positive Emotions Build Physical Health : Perceived Positive Social Connections Account for the Upward Spiral Between Positive Emotions and Vagal Tone. *Psychol. Sci.* **2013**, *24*, 1123–1132. [CrossRef] [PubMed]
5. Nguyen, V.T.; Longin, D.; Ho, T.V.; Gaudou, B. Integration of Emotion in Evacuation Simulation. In *Information Systems for Crisis Response and Management in Mediterranean Countries, Proceedings of the First International Conference, ISCRAM-med 2014, Toulouse, France, 15–17 October 2014*; Springer International Publishing: Cham, Switzerland, 2014; pp. 192–205.
6. Pervez, M.A. Impact of emotions on employee's job performance: An evidence from organizations of Pakistan. *OIDA Int. J. Sustain. Dev.* **2010**, *1*, 11–16.
7. Weiss, H.M. Introductory Comments: Antecedents of Emotional Experiences at Work. *Motiv. Emot.* **2002**, *26*, 1–2. [CrossRef]
8. Bhuyar, R.; Ansari, S. Design and Implementation of Smart Office Automation System. *Int. J. Comput. Appl.* **2016**, *151*, 37–42. [CrossRef]
9. Van der Valk, S.; Myers, T.; Atkinson, I.; Mohring, K. Sensor networks in workplaces: Correlating comfort and productivity. In Proceedings of the 2015 IEEE Tenth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), Singapore, 7–9 April 2015; pp. 1–6.
10. Zhou, J.; Yu, C.; Riekki, J.; Kärkkäinen, E. AmE framework: A model for emotion-aware ambient intelligence. In Proceedings of the second international conference on affective computing and intelligent interaction (ACII2007): Doctoral Consortium, Lisbon, Portugal, 12–14 September 2007; p. 45.
11. Acampora, G.; Loia, V.; Vitiello, A. Distributing emotional services in ambient intelligence through cognitive agents. *Serv. Oriented Comput. Appl.* **2011**, *5*, 17–35. [CrossRef]
12. Beer, W.; Christian, V.; Ferscha, A.; Mehrmann, L. Modeling Context-Aware Behavior by Interpreted ECA Rules. In *Euro-Par 2003 Parallel Processing*; Kosch, H., Böszörményi, L., Hellwagner, H., Eds.; Springer: Berlin/Heidelberg, Germany, 2003; pp. 1064–1073.
13. Coronado, M.; Iglesias, C.A.; Serrano, E. Modelling rules for automating the Evented WEb by semantic technologies. *Expert Syst. Appl.* **2015**, *42*, 7979–7990. [CrossRef]
14. Muñoz, S.; Fernández, A.; Coronado, M.; Iglesias, C.A. Smart Office Automation based on Semantic Event-Driven Rules. In Proceedings of the Workshop on Smart Offices and Other Workplaces, Colocated with 12th International Conference on Intelligent Environments (IE'16), London, UK, 14–16 September 2016; Ambient Intelligence and Smart Environments; IOS Press: Clifton, VA, USA, 2016; Volume 21, pp. 33–42.
15. Inada, T.; Igaki, H.; Ikegami, K.; Matsumoto, S.; Nakamura, M.; Kusumoto, S. Detecting Service Chains and Feature Interactions in Sensor-Driven Home Network Services. *Sensors* **2012**, *12*, 8447–8464. [CrossRef] [PubMed]
16. Zhou, J.; Kallio, P. Ambient emotion intelligence: from business awareness to emotion awareness. In Proceedings of the 17th International Conference on Systems Research, Berlin, Germany, 15–17 April 2014; pp. 47–54.
17. Kanjo, E.; Al-Husain, L.; Chamberlain, A. Emotions in context: Examining pervasive affective sensing systems, applications, and analyses. *Pers. Ubiquitous Comput.* **2015**, *19*, 1197–1212. [CrossRef]
18. Kanjo, E.; El Mawass, N.; Craveiro, J. Social, disconnected or in between: Mobile data reveals urban mood. In Proceedings of the 3rd International Conference on the Analysis of Mobile Phone Datasets (NetMob'13), Cambridge, MA, USA, 1–3 May 2013.
19. Wagner, J.; André, E.; Jung, F. Smart sensor integration: A framework for multimodal emotion recognition in real-time. In Proceedings of the 3rd International Conference on Affective Computing and Intelligent Interaction and Workshops, Amsterdam, The Netherlands, 10–12 September 2009; pp. 1–8.
20. Gay, G.; Pollak, J.; Adams, P.; Leonard, J.P. Pilot study of Aurora, a social, mobile-phone-based emotion sharing and recording system. *J. Diabetes Sci. Technol.* **2011**, *5*, 325–332. [CrossRef] [PubMed]
21. Gaggioli, A.; Pioggia, G.; Tartarisco, G.; Baldus, G.; Corda, D.; Cipresso, P.; Riva, G. A mobile data collection platform for mental health research. *Pers. Ubiquitous Comput.* **2013**, *17*, 241–251. [CrossRef]
22. Morris, M.E.; Kathawala, Q.; Leen, T.K.; Gorenstein, E.E.; Guilak, F.; Labhard, M.; Deleeuw, W. Mobile therapy: Case study evaluations of a cell phone application for emotional self-awareness. *J. Med. Internet Res.* **2010**, *12*, e10. [CrossRef] [PubMed]

23. Bergner, B.S.; Exner, J.P.; Zeile, P.; Rumberg, M. Sensing the city—How to identify recreational benefits of urban green areas with the help of sensor technology. In Proceedings of the REAL CORP 2012, Schwechat, Austria, 14–16 May 2012.
24. Fernández-Caballero, A.; Martínez-Rodrigo, A.; Pastor, J.M.; Castillo, J.C.; Lozano-Monasor, E.; López, M.T.; Zangróniz, R.; Latorre, J.M.; Fernández-Sotos, A. Smart environment architecture for emotion detection and regulation. *J. Biomed. Inform.* **2016**, *64*, 55–73. [CrossRef] [PubMed]
25. Jungum, N.V.; Laurent, E. Emotions in pervasive computing environments. *Int. J. Comput. Sci. Issues* **2009**, *6*, 8–22.
26. Bisio, I.; Delfino, A.; Lavagetto, F.; Marchese, M.; Sciarrone, A. Gender-driven emotion recognition through speech signals for ambient intelligence applications. *IEEE Trans. Emerg. Top. Comput.* **2013**, *1*, 244–257. [CrossRef]
27. Acampora, G.; Vitiello, A. Interoperable neuro-fuzzy services for emotion-aware ambient intelligence. *Neurocomputing* **2013**, *122*, 3–12. [CrossRef]
28. Marreiros, G.; Santos, R.; Novais, P.; Machado, J.; Ramos, C.; Neves, J.; Bula-Cruz, J. Argumentation-based decision making in ambient intelligence environments. In Proceedings of the Portuguese Conference on Artificial Intelligence, Guimarães, Portugal, 3–7 December 2007; pp. 309–322.
29. Hagras, H.; Callaghan, V.; Colley, M.; Clarke, G.; Pounds-Cornish, A.; Duman, H. Creating an ambient-intelligence environment using embedded agents. *IEEE Intell. Syst.* **2004**, *19*, 12–20. [CrossRef]
30. Mennicken, S.; Vermeulen, J.; Huang, E.M. From today's augmented houses to tomorrow's smart homes: New directions for home automation research. In Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing, Seattle, WA, USA, 13–17 September 2014; pp. 105–115.
31. Cook, D.; Das, S. *Smart Environments: Technology, Protocols and Applications (Wiley Series on Parallel and Distributed Computing)*; Wiley-Interscience: Seattle, WA, USA, 2004.
32. Marsa-maestre, I.; Lopez-carmona, M.A.; Velasco, J.R.; Navarro, A. Mobile Agents for Service Personalization in Smart Environments. *J. Netw.* **2008**, *3*. [CrossRef]
33. Furdik, K.; Lukac, G.; Sabol, T.; Kostelnik, P. The Network Architecture Designed for an Adaptable IoT-based Smart Office Solution. *Int. J. Comput. Netw. Commun. Secur.* **2013**, *1*, 216–224.
34. Shigeta, H.; Nakase, J.; Tsunematsu, Y.; Kiyokawa, K.; Hatanaka, M.; Hosoda, K.; Okada, M.; Ishihara, Y.; Ooshita, F.; Kakugawa, H.; et al. Implementation of a smart office system in an ambient environment. In Proceedings of the 2012 IEEE Virtual Reality Workshops (VRW), Costa Mesa, CA, USA, 4–8 March 2012; pp. 1–2.
35. Zenonos, A.; Khan, A.; Kalogridis, G.; Vatsikas, S.; Lewis, T.; Sooriyabandara, M. HealthyOffice: Mood recognition at work using smartphones and wearable sensors. In Proceedings of the 2016 IEEE International Conference on Pervasive Computing and Communication Workshops (PerCom Workshops), Sydney, Australia, 14–18 March 2016; pp. 1–6.
36. Li, H. A novel design for a comprehensive smart automation system for the office environment. In Proceedings of the 2014 IEEE Emerging Technology and Factory Automation (ETFA), Barcelona, Spain, 16–19 September 2014; pp. 1–4.
37. Jalal, A.; Kamal, S.; Kim, D. A Depth Video Sensor-Based Life-Logging Human Activity Recognition System for Elderly Care in Smart Indoor Environments. *Sensors* **2014**, *14*, 11735–11759. [CrossRef] [PubMed]
38. Kumar, V.; Fensel, A.; Fröhlich, P. Context Based Adaptation of Semantic Rules in Smart Buildings. In Proceedings of the International Conference on Information Integration and Web-based Applications & Services, Vienna, Austria, 2–4 December 2013; ACM: New York, NY, USA, 2013; pp. 719–728.
39. Alirezaie, M.; Renoux, J.; Köckemann, U.; Kristoffersson, A.; Karlsson, L.; Blomqvist, E.; Tsiftes, N.; Voigt, T.; Loutfi, A. An Ontology-based Context-aware System for Smart Homes: E-care@home. *Sensors* **2017**, *17*, 1586. [CrossRef] [PubMed]
40. Coronato, A.; Pietro, G.D.; Esposito, M. A Semantic Context Service for Smart Offices. In Proceedings of the 2006 International Conference on Hybrid Information Technology, Cheju Island, Korea, 9–11 November 2006; Volume 2, pp. 391–399.
41. Picard, R.W.; Healey, J. Affective wearables. *Pers. Technol.* **1997**, *1*, 231–240. [CrossRef]
42. Gyrard, A. A machine-to-machine architecture to merge semantic sensor measurements. In Proceedings of the 22nd International Conference on World Wide Web, Rio de Janeiro, Brazil, 13–17 May 2013.

43. Araque, O.; Corcuera-Platas, I.; Sánchez-Rada, J.F.; Iglesias, C.A. Enhancing deep learning sentiment analysis with ensemble techniques in social applications. *Expert Syst. Appl.* **2017**, *77*, 236–246. [CrossRef]
44. Pantic, M.; Bartlett, M. Machine Analysis of Facial Expressions. In *Face Recognition*; I-Tech Education and Publishing: Vienna, Austria, 2007; pp. 377–416.
45. Sebe, N.; Cohen, I.; Gevers, T.; Huang, T.S. Multimodal approaches for emotion recognition: A survey. In Proceedings of the Electronic Imaging 2005, San Jose, CA, USA, 17 January 2005; Volume 5670, p. 5670.
46. Mehta, D.; Siddiqui, M.F.H.; Javaid, A.Y. Facial Emotion Recognition: A Survey and Real-World User Experiences in Mixed Reality. *Sensors* **2018**, *18*, 416. [CrossRef] [PubMed]
47. Anagnostopoulos, C.N.; Iliou, T.; Giannoukos, I. Features and Classifiers for Emotion Recognition from Speech: A Survey from 2000 to 2011. *Artif. Intell. Rev.* **2015**, *43*, 155–177. [CrossRef]
48. Ayadi, M.E.; Kamel, M.S.; Karray, F. Survey on speech emotion recognition: Features, classification schemes, and databases. *Pattern Recognit.* **2011**, *44*, 572–587. [CrossRef]
49. Vinola, C.; Vimaladevi, K. A Survey on Human Emotion Recognition Approaches, Databases and Applications. *ELCVIA Electron. Lett. Comput. Vis. Image Anal.* **2015**, *14*, 24–44. [CrossRef]
50. Brouwer, A.M.; van Wouwe, N.; Mühl, C.; van Erp, J.; Toet, A. Perceiving blocks of emotional pictures and sounds: Effects on physiological variables. *Front. Hum. Neurosci.* **2013**, *7*, 1–10. [CrossRef] [PubMed]
51. Campos, J.J.; Frankel, C.B.; Camras, L. On the Nature of Emotion Regulation. *Child Dev.* **2004**, *75*, 377–394. [CrossRef] [PubMed]
52. Sokolova, M.V.; Fernández-Caballero, A.; Ros, L.; Latorre, J.M.; Serrano, J.P. Evaluation of Color Preference for Emotion Regulation. In Proceedings of the Artificial Computation in Biology and Medicine: International Work-Conference on the Interplay Between Natural and Artificial Computation, IWINAC 2015, Elche, Spain, 1–5 June 2015; Springer International Publishing: Cham, Switzerland, 2015; pp. 479–487.
53. Philippot, P.; Chapelle, G.; Blairy, S. Respiratory feedback in the generation of emotion. *Cogn. Emot.* **2002**, *16*, 605–627. [CrossRef]
54. Xin, J.H.; Cheng, K.M.; Taylor, G.; Sato, T.; Hansuebsai, A. Cross-regional comparison of colour emotions Part I: Quantitative analysis. *Color Res. Appl.* **2004**, *29*, 451–457. [CrossRef]
55. Xin, J.H.; Cheng, K.M.; Taylor, G.; Sato, T.; Hansuebsai, A. Cross-regional comparison of colour emotions Part II: Qualitative analysis. *Color Res. Appl.* **2004**, *29*, 458–466. [CrossRef]
56. Ortiz-García-Cervigón, V.; Sokolova, M.V.; García-Muñoz, R.M.; Fernández-Caballero, A. LED Strips for Color- and Illumination-Based Emotion Regulation at Home. In Proceedings of the 7th International Work-Conference, IWAAL 2015, ICT-Based Solutions in Real Life Situations, Puerto Varas, Chile, 1–4 December 2015; pp. 277–287.
57. Lingham, J.; Theorell, T. Self-selected "favourite" stimulative and sedative music listening—How does familiar and preferred music listening affect the body? *Nord. J. Music Ther.* **2009**, *18*, 150–166. [CrossRef]
58. Pannese, A. A gray matter of taste: Sound perception, music cognition, and Baumgarten's aesthetics. *Stud. Hist. Philos. Sci. Part C Stud. Hist. Philos. Biol. Biomed. Sci.* **2012**, *43*, 594–601. [CrossRef] [PubMed]
59. Van der Zwaag, M.D.; Dijksterhuis, C.; de Waard, D.; Mulder, B.L.; Westerink, J.H.; Brookhuis, K.A. The influence of music on mood and performance while driving. *Ergonomics* **2012**, *55*, 12–22. [CrossRef] [PubMed]
60. Uhlig, S.; Jaschke, A.; Scherder, E. Effects of Music on Emotion Regulation: A Systematic Literature Review. In Proceedings of the 3rd International Conference on Music and Emotion (ICME3), Yväskylä, Finland, 11–15 June 2013; pp. 11–15.
61. Freggens, M.J. The Effect of Music Type on Emotion Regulation: An Emotional Stroop Experiment. Ph.D. Thesis, Georgia State University, Atlanta, GA, USA, 2015.
62. Gangemi, A. Ontology design patterns for semantic web content. In Proceedings of the 4th International Semantic Web Conference, Galway, Ireland, 6–10 November 2005; Springer: Berlin/Heidelberg, Germany, 2005; pp. 262–276.
63. Prud, E.; Seaborne, A. *SPARQL Query Language for RDF*; Technical Report; W3C: Cambridge, MA, USA, 2006.
64. Klyne, G.; Carroll, J.J. *Resource Description Framework (RDF): Concepts and Abstract Syntax*; Technical Report; W3C: Cambridge, MA, USA, 2006.
65. Sporny, M.; Longley, D.; Kellogg, G.; Lanthaler, M.; Lindström, N. *JSON-LD 1.0*; Technical Report; W3C: Cambridge, MA, USA, 2014.

66. Boley, H.; Paschke, A.; Shafiq, O. RuleML 1.0: The Overarching Specification of Web Rules. In Proceedings of the Semantic Web Rules: International Symposium, RuleML 2010, Washington, DC, USA, 21–23 October 2010; Springer: Berlin/Heidelberg, Germany, 2010; pp. 162–178.
67. O'Connor, M.; Knublauch, H.; Tu, S.; Grosof, B.; Dean, M.; Grosso, W.; Musen, M. Supporting Rule System Interoperability on the Semantic Web with SWRL. In Proceedings of the Semantic Web–ISWC 2005: 4th International Semantic Web Conference, ISWC 2005, Galway, Ireland, 6–10 November 2005; Springer: Berlin/Heidelberg, Germany, 2005; pp. 974–986.
68. Kifer, M. Rule Interchange Format: The Framework. In Proceedings of the Web Reasoning and Rule Systems: Second International Conference, RR 2008, Karlsruhe, Germany, 31 October–1 November 2008; Springer: Berlin/Heidelberg, Germany, 2008; pp. 1–11.
69. Knublauch, H.; Hendler, J.A.; Idehen, K. *SPIN Overview and Motivation*; Technical Report; W3C: Cambridge, MA, USA, 2011.
70. Berners-Lee, T. *Notation3 Logic*; Technical Report; W3C: Cambridge, MA, USA, 2011.
71. Coronado, M.; Iglesias, C.A.; Serrano, E. Task Automation Services Study. 2015. Available online: http://www.gsi.dit.upm.es/ontologies/ewe/study/full-results.html (accessed on 18 April 2018).
72. Schröder, M.; Baggia, P.; Burkhardt, F.; Pelachaud, C.; Peter, C.; Zovato, E. EmotionML—An upcoming standard for representing emotions and related states. In Proceedings of the International Conference on Affective Computing and Intelligent Interaction, Memphis, TN, USA, 9–12 October 2011; Springer: Berlin/Heidelberg, Germany, 2011; pp. 316–325.
73. Sánchez-Rada, J.F.; Iglesias, C.A. Onyx: A Linked Data Approach to Emotion Representation. *Inf. Process. Manag.* **2016**, *52*, 99–114. [CrossRef]
74. Grassi, M. Developing HEO Human Emotions Ontology. In *Biometric ID Management and Multimodal Communication, Proceedings of the Joint COST 2101 and 2102 International Conference, BioID_MultiComm 2009, Madrid, Spain, 16–18 September 2009*; Springer: Berlin/Heidelberg, Germany, 2009; pp. 244–251.
75. Lebo, T.; Sahoo, S.; McGuinness, D.; Belhajjame, K.; Cheney, J.; Corsar, D.; Garijo, D.; Soiland-Reyes, S.; Zednik, S.; Zhao, J. Prov-o: The prov ontology. In *W3C Recommendation, 30th April*; 00000 bibtex: Lebo2013; W3C: Cambridge, MA, USA, 2013.
76. Schröder, M.; Pelachaud, C.; Ashimura, K.; Baggia, P.; Burkhardt, F.; Oltramari, A.; Peter, C.; Zovato, E. Vocabularies for emotionml. In *W3C Working Group Note, World Wide Web Consortium*; W3C: Cambridge, MA, USA, 2011.
77. Sánchez-Rada, J.F.; Iglesias, C.A.; Gil, R. A linked data model for multimodal sentiment and emotion analysis. In Proceedings of the 4th Workshop on Linked Data in Linguistics: Resources and Applications, Beijing, China, 31 July 2015; pp. 11–19.
78. Sánchez-Rada, J.F.; Iglesias, C.A.; Sagha, H.; Schuller, B.; Wood, I.; Buitelaar, P. Multimodal multimodel emotion analysis as linked data. In Proceedings of the 2017 Seventh International Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW), San Antonio, TX, USA, 23–26 October 2017; pp. 111–116.
79. Sánchez-Rada, J.F.; Schuller, B.; Patti, V.; Buitelaar, P.; Vulcu, G.; Bulkhardt, F.; Clavel, C.; Petychakis, M.; Iglesias, C.A. Towards a Common Linked Data Model for Sentiment and Emotion Analysis. In Proceedings of the LREC 2016 Workshop Emotion and Sentiment Analysis (ESA 2016), Portorož, Slovenia, 23 May 2016; Sánchez-Rada, J.F., Schuller, B., Eds.; 2016; pp. 48–54.
80. Khoozani, E.N.; Hadzic, M. Designing the human stress ontology: A formal framework to capture and represent knowledge about human stress. *Aust. Psychol.* **2010**, *45*, 258–273. [CrossRef]
81. Coronado, M.; Iglesias, C.A. Task Automation Services: Automation for the masses. *IEEE Internet Comput.* **2015**, *20*, 52–58. [CrossRef]
82. Verborgh, R.; Roo, J.D. Drawing Conclusions from Linked Data on the Web: The EYE Reasoner. *IEEE Softw.* **2015**, *32*, 23–27. [CrossRef]
83. Sánchez-Rada, J.F.; Iglesias, C.A.; Coronado, M. A modular architecture for intelligent agents in the evented web. *Web Intell.* **2017**, *15*, 19–33. [CrossRef]
84. Coronado Barrios, M. A Personal Agent Architecture for Task Automation in the Web of Data. Bringing Intelligence to Everyday Tasks. Ph.D. Thesis, Technical University of Madrid, Madrid, Spain, 2016.
85. Williams, K. The Technology Ecosystem: Fueling Google's Chromecast [WIE from Around the World]. *IEEE Women Eng. Mag.* **2014**, *8*, 30–32. [CrossRef]

86. Cortina, J.M. What is coefficient alpha? An examination of theory and applications. *J. Appl. Psychol.* **1993**, *78*, 98. [CrossRef]
87. Gaeta, M.; Loia, V.; Orciuoli, F.; Ritrovato, P. S-WOLF: Semantic workplace learning framework. *IEEE Trans. Syst. Man Cybern. Syst.* **2015**, *45*, 56–72. [CrossRef]

© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).

A.2.3 Enhancing Deep Learning Sentiment Analysis with Ensemble Techniques in Social Applications

Title	Enhancing Deep Learning Sentiment Analysis with Ensemble Techniques in Social Applications
Authors	Araque, Oscar and Corcuera-Platas, Ignacio and Sánchez-Rada, J. Fernando and Iglesias, Carlos A.
Journal	Expert Systems with Applications
Impact factor	JCR Q1 (2.981)
ISSN	0957-4174
Publisher	
Year	2017
Keywords	deep learning, Deep learning, downstream ensemble, Ensemble, machine learning, Machine learning, natural language processing, Natural language processing, sentiment analysis, Sentiment analysis
Pages	
Online	http://www.sciencedirect.com/science/article/pii/S0957417417300751
Abstract	The appearance of new Deep Learning applications for Sentiment Analysis has motivated a lot of researchers, mainly because of their automatic feature extraction and representation capabilities, as well as their better performance compared to the previous feature based techniques. These traditional surface approaches are based on complex manually extracted features, and this extraction process is a fundamental question in feature driven methods. However, these long-established approaches can yield strong baselines on their own, and its predictive capabilities can be used in conjunction with the arising Deep Learning methods. In this paper we seek to improve the performance of these new Deep Learning techniques integrating them with more traditional surface approaches based on manually extracted features. The contributions of this paper are: first, we develop a Deep Learning based Sentiment classifier using the Word2Vec model and a linear machine learning algorithm. This classifier serves us as a baseline with which we can compare subsequent results. Second, we propose two ensemble techniques which aggregate our baseline classifier with other surface classifiers widely used in the field of Sentiment Analysis. Third, we also propose two models for combining deep features with both surface and deep features in order to merge the information from several sources. As fourth contribution, we introduce a taxonomy for classifying the different models we propose, as well as the ones found in the literature. Fifth, we conduct several reproducible experiments with the aim of comparing the performance of these models with the Deep Learning baseline. For this, we employ four public datasets that were extracted from the microblogging domain. Finally, as a result, the experiments confirm that the performance of these proposed models surpasses that of our original baseline using as metric the F1-Score, with improvements ranging from 0.21 to 3.62 %.

Expert Systems With Applications 77 (2017) 236–246

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Enhancing deep learning sentiment analysis with ensemble techniques in social applications

Oscar Araque*, Ignacio Corcuera-Platas, J. Fernando Sánchez-Rada, Carlos A. Iglesias

Universidad Politécnica de Madrid, Escuela Técnica Superior de Ingenieros de Telecomunicación, Departamento de Ingeniería de Sistemas Telemáticos, Avenida Complutense 30, Madrid, Spain

ARTICLE INFO

Article history:
Received 30 June 2016
Revised 31 January 2017
Accepted 1 February 2017
Available online 3 February 2017

Keywords:
Ensemble
Deep learning
Sentiment analysis
Machine learning
Natural language processing

ABSTRACT

Deep learning techniques for Sentiment Analysis have become very popular. They provide automatic feature extraction and both richer representation capabilities and better performance than traditional feature based techniques (i.e., surface methods). Traditional surface approaches are based on complex manually extracted features, and this extraction process is a fundamental question in feature driven methods. These long-established approaches can yield strong baselines, and their predictive capabilities can be used in conjunction with the arising deep learning methods. In this paper we seek to improve the performance of deep learning techniques integrating them with traditional surface approaches based on manually extracted features. The contributions of this paper are sixfold. First, we develop a deep learning based sentiment classifier using a word embeddings model and a linear machine learning algorithm. This classifier serves as a baseline to compare to subsequent results. Second, we propose two ensemble techniques which aggregate our baseline classifier with other surface classifiers widely used in Sentiment Analysis. Third, we also propose two models for combining both surface and deep features to merge information from several sources. Fourth, we introduce a taxonomy for classifying the different models found in the literature, as well as the ones we propose. Fifth, we conduct several experiments to compare the performance of these models with the deep learning baseline. For this, we use seven public datasets that were extracted from the microblogging and movie reviews domain. Finally, as a result, a statistical study confirms that the performance of these proposed models surpasses that of our original baseline on F1-Score.

© 2017 The Authors. Published by Elsevier Ltd.
This is an open access article under the CC BY-NC-ND license.
(http://creativecommons.org/licenses/by-nc-nd/4.0/)

1. Introduction

The growth of user-generated content in web sites and social networks, such as Twitter, Amazon, and Trip Advisor, has led to an increasing power of social networks for expressing opinions about services, products or events, among others. This tendency, combined with the fast spreading nature of content online, has turned online opinions into a very valuable asset. In this context, many Natural Language Processing (NLP) tasks are being used in order to analyze this massive information. In particular, Sentiment Analysis (SA) is an increasingly growing task (Liu, 2015), whose goal is the classification of opinions and sentiments expressed in text, generated by a human party.

The dominant approaches in sentiment analysis are based on *machine learning* techniques (Pang, Lee, & Vaithyanathan, 2002; Read, 2005; Wang & Manning, 2012). Traditional approaches frequently use the Bag Of Words (BOW) model, where a document is mapped to a feature vector, and then classified by machine learning techniques. Although the BOW approach is simple and quite efficient, a great deal of the information from the original natural language is lost (Xia & Zong, 2010), e.g., word order is disrupted and syntactic structures are broken. Therefore, various types of features have been exploited, such as higher order *n*-grams (Pak & Paroubek, 2010). Another kind of feature that can be used is Part Of Speech (POS) tagging, which is commonly used during a syntactic analysis process, as described in Gimpel et al. (2011). Some authors refer to this kind of features as *surface* forms, as they consist in lexical and syntactical information that relies on the pattern of the text, rather than on its semantic aspect.

Some prior information about sentiment can also be used in the analysis. For instance, by adding individual word polarity to the previously described features (Pablos, Cuadros, & Rigau, 2016). This

* Corresponding author.
E-mail addresses: o.araque@upm.es (O. Araque), ignacio.cplatas@alumnos.upm.es (I. Corcuera-Platas), jfernando@dit.upm.es (J.F. Sánchez-Rada), cif@gsi.dit.upm.es (C.A. Iglesias).

http://dx.doi.org/10.1016/j.eswa.2017.02.002
0957-4174/© 2017 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license. (http://creativecommons.org/licenses/by-nc-nd/4.0/)

prior knowledge usually takes the form of *sentiment lexicons*, which have to be gathered. Sentiment lexicons are used as a source of subjective sentiment knowledge, where this knowledge is added to the previously described features (Cambria, 2016; Kiritchenko, Zhu, & Mohammad, 2014; Melville, Gryc, & Lawrence, 2009; Nasukawa & Yi, 2003).

The use of lexicon-based techniques has a number of advantages (Taboada, Brooke, Tofiloski, Voll, & Stede, 2011). First, the linguistic content can be taken into account through mechanisms such as sentiment valence shifting (Polanyi & Zaenen, 2006) considering both intensifiers (e.g. very bad) and negations (e.g. not happy). In addition, sentiment orientation of lexical entities can be differentiated based on their characteristics. Moreover, language-dependent characteristics can be included in these approaches. Nevertheless, lexicon-based approaches have several drawbacks: the need of a lexicon that is consistent and reliable (Taboada et al., 2011), as well as the variability of opinion words across domains (Turney, 2002), contexts (Ding, Liu, & Yu, 2008) and languages (Perez-Rosas, Banea, & Mihalcea, 2012). These dependencies make it hard to maintain domain independent lexicons (Qiu, Liu, Bu, & Chen, 2009).

In general, extracting complex features from text, figuring out which features are relevant, and selecting a classification algorithm are fundamental questions in the machine learning driven methods (Agarwal, Xie, Vovsha, Rambow, & Passonneau, 2011; Sharma & Dey, 2012; Wilson, Wiebe, & Hoffmann, 2009). Traditional approaches rely on manual feature engineering, which is time consuming.

On the other hand, deep learning is a promising alternative to traditional methods. It has shown excellent performance in NLP tasks, including Sentiment Analysis (Collobert et al., 2011). The main idea of deep learning techniques is to learn complex features extracted from data with minimum external contribution (Bengio, 2009) using deep neural networks (Alpaydin, 2014). These algorithms do not need to be passed manually crafted features: they automatically learn new complex features. Nevertheless, a characteristic feature of deep learning approaches is that they need large amounts of data to perform well (Mikolov, Chen, Corrado, & Dean, 2013). Both automatic feature extraction and availability of resources are very important when comparing the traditional machine learning approach and deep learning techniques.

However, it is not clear whether the domain specialization capacity of traditional approaches can be surpassed with the generalization capacity of deep learning based models in all NLP tasks, or if it is possible to successfully combine these two techniques in a wide range of applications.

In this paper, we propose a combination of these two main sentiment analysis approaches through several ensemble models in which the information provided by many kinds of features is aggregated. In particular, this work considers an ensemble of classifiers, where several sentiment classifiers trained with different kinds of features are combined, and an ensemble of features, where the combination is made at the feature level. In order to study the complementarity of the proposed models, we use six public test datasets from two different domains: Twitter and movie reviews. Moreover, we performed a statistical study on the results of these ensemble models in comparison to a deep learning baseline we have also developed. We also present the complexity of the proposed ensemble models. Besides, we present a taxonomy that classifies the models found in the literature and the ones proposed in this work.

With our proposal we seek answers to the following questions, using the empirical results we have obtained as basis:

1. Is there a framework for characterizing existing approaches in relation to the ensemble of deep and traditional techniques in sentiment analysis?
2. Can deep learning approaches benefit from their ensemble with surface approaches?
3. How do different deep and surface ensembles compare in terms of performance?

The rest of the paper is organized as follows. Section 2 shows previous work on both ensemble techniques and deep learning approaches. Section 3 describes the proposed taxonomy for classifying ensemble methods that merge surface and deep features, whereas Section 4 addresses the proposed classifier and ensemble models. In Section 5, we describe the designed experimental setup. Experimental results are presented and analyzed in Section 6. Finally, Section 7 draws conclusions from previous results and outlines the future work.

2. Related work

In this section we offer a brief summary of the previous work in the context of ensemble methods and deep learning algorithms for Sentiment Analysis.

2.1. Ensemble methods for sentiment analysis

In the field of ensemble methods, the main idea is to combine a set of models (base classifiers) in order to obtain a more accurate and reliable model in comparison with what a single model can achieve. The methods used for building upon an ensemble approach are many, and a categorization is presented in Rokach (2005). This classification is based on two main dimensions: how predictions are combined (rule based and meta learning), and how the learning process is done (concurrent and sequential).

Regarding the first dimension, on the one hand, in *rule based* approaches predictions from the base classifiers are treated by a rule, with the aim of averaging their predictive performance. Examples of rule based ensembles are the majority voting, where the output prediction per sample is the most common class; and the weighted combination, which linearly aggregates the base classifiers predictions. On the other hand, *meta learning* techniques use predictions from component classifiers as features for a meta-learning model.

As explained in Xia, Zong, and Li (2011), weighted combinations of feature sets can be quite effective in the task of sentiment classification, since the weights of the ensemble represent the relevance of the different feature sets (e.g. n-grams, POS, etc.) to sentiment classification, instead of assigning relevance to each feature individually. The benefits of rule based ensembles were shown also in Fersini, Messina, and Pozzi (2014), where several variants of voting rules are exhaustively studied in a variety of datasets, with an emphasis on the complexity that results from the use of these approaches. In a different work, Fersini, Messina, and Pozzi (2016) have compared the majority voting rule with other approaches, using three types of subjective signals: adjectives, emoticons, emphatic expressions and expressive elongations. They report that adjectives are more impacting that the other considered signals, and that the average rule is able to ensure better performance than other types of rules. Also, in Xia et al. (2011) a meta-classifier ensemble model is evaluated, obtaining performance improvements as well. An adaptive meta-learning model is described in Aue and Gamon (2005), which offers a relatively low adaptation effort to new domains. Besides, both rule based and meta-learning ensemble models can be enriched with extra knowledge, as illustrated in Xia and Zong (2011). These authors propose the use of a number of rule based ensemble models, namely a sum rule and two weighted combination approaches trained with different loss functions. The base classifiers are trained with n-grams and POS features. These models obtain significant results for cross-domain sentiment classification.

As for the second dimension, *concurrent* models divide the original dataset into several subsets from which multiple classifiers learn in a parallel fashion, creating a classifier composite. The most popular technique that processes the sample concurrently is bagging (Rokach, 2005). Bagging intends to improve the classification by combining the predictions of classifiers built on random subsets of the original data. On the contrary, *sequential* approaches do not divide the dataset but there is an interaction between the learning steps, taking advantage from previous iterations of the learning process to improve the quality of the global classifier. An interesting sequential approach is boosting, which consists in repeatedly training low-performance classifiers on different training data. The classifiers trained in this manner are then combined into a single classifier that can achieve better performance than the component classifiers.

An example of bagging performance in the sentiment analysis task can be found in Sehgal and Song (2007), where bagging and other classification algorithms are used to show that the sentiment evolution and the stock value trend are closely related. Fersini et al. (2014) also show several experimental results in relation to the bagging techniques, attending also to the associated model complexity. Moreover, some authors have shown that bagging techniques are fairly robust to noisy data, while boosting techniques are quite sensitive (Maclin & Opitz, 1997; Melville, Shah, Mihalkova, & Mooney, 2004; Prusa, Khoshgoftaar, & Dittman, 2015). The suitability of bagging and boosting ensembles is also experimentally confirmed by Wang, Sun, Ma, Xu, and Gu (2014). This work also includes the study of a different ensemble technique, random subspace, that consists in modifying the training dataset in the feature space, rather than on the instance space. The authors stand out the better performance of random subspace in comparison with similar approaches, such as bagging and boosting. Another study (Whitehead & Yaeger, 2010) shows a comparison between bagging and boosting on a standard opinion mining task. Besides, Lin, Wang, Li, and Zhou (2015) proposes a three phase framework of multiple classifiers, where an optimal subset of classifiers is automatically chosen and trained. This framework is tested in several real-world datasets for sentiment classification.

Nevertheless, these works also show that ensemble techniques not always improve the performance in the sentiment analysis task, and that there is not a global criteria to select a certain ensemble technique.

2.2. Deep learning approaches

In the realm of Natural Language Processing much of the work in deep learning has been oriented towards methods involving learning word vector representations using neural language models (Kim, 2014). Continuous representations of words as vectors has proven to be an effective technique in many NLP tasks, including sentiment analysis (Tang, Wei, Yang et al., 2014). In this sense, *word2vec* is one of the most popular approaches that allows modeling words as vectors (Mikolov, Chen et al., 2013). Word2vec is based on the Skip-gram and CBOW models to perform the computation of the distributed representations. While CBOW aims to predict a word given its context, Skip-gram predicts the context given a word. Word2vec computes continuous vector representations of words form very large datasets. The computed word vectors retain a huge amount of syntactic and semantic regularities present in the language (Mikolov, Yih, & Zweig, 2013), expressed as relation offsets in the resulting vector space. These word-level embeddings are encoded by column vectors in an embedding matrix $W \in \mathrm{IR}^{d \times |V|}$, where $|V|$ is the size of the vocabulary. Each column $W_i \in \mathrm{IR}^d$ corresponds to the word embeddings vector of the i-th word in the vocabulary. The transformation of a word w into its word embedding vector r_w is made by using the matrix-vector product:

$$r_w = W v_w$$

where v_w is an one-hot vector of size $|V|$ which has value index at w and zero in the rest. The matrix W components are parameters to be learned, and the dimension of the word vectors d is a hyperparameter to be chosen. The vector representations computed by these techniques can result very effective when used with a traditional classifier (e.g. logistic regression) for sentiment classification, as shown by Zhang, Xu, Su, and Xu (2015). An approach based in word2vec is *doc2vec* (Le & Mikolov, 2014), that models entire sentences or documents as vectors. An additional method in representation learning is the auto-encoder, which is a type of artificial neural network applied to unsupervised learning. Auto-encoders have been used for learning new representations on a wide range of machine learning tasks, such as learning representations from distorted data, as illustrated in Chen, Weinberger, Sha, and Bengio (2014).

In deep learning for SA, an interesting approach is to augment the knowledge contained in the embedding vectors with other sources of information. This added information can be sentiment specific word embedding as in Tang, Wei, Yang et al. (2014), or as in a similar work, a concatenation of manually crafted features and these sentiment specific word embeddings (Tang, Wei, Qin, Liu, & Zhou, 2014). In the work presented by Zhang and He (2015) the feature set extracted from word embeddings is enriched with latent topic features, combining them in an ensemble scheme. They also experimentally demonstrate that these enriched representations are effective for improving the performance of polarity classification. Another approach that incorporates new information to the embeddings is described in Su, Xu, Zhang, and Xu (2014), in which deep learning is used to extract sentiment features in conjunction with semantic features. Severyn and Moschitti (2015) describe an approach where distant supervised data is used to refine the parameters of the neural network from the unsupervised neural language model. Also, a collaborative filtering algorithm can be used, as is detailed in Kim et al. (2013), where the authors add sentiment information from a small fraction of the data. In the line of adding sentiment information, in Li et al. (2015) is portrayed how a sentiment Recursive Neural Network (RNN) can be used in parallel to another neural network architecture. In general, there is a growing tendency which tries to incorporate additional information to the word embeddings created by deep learning networks. An interesting work is the one described in Vo and Zhang (2015), where both sentiment-driven and standard embeddings are used in conjunction with a variety of pooling functions, in order to extract the target-oriented sentiment of Twitter comments. Enriching the information contained in word embeddings is not the only trend in deep learning for SA. The study of the compositionality in the sentiment classification task has proven to be relevant, as shown by Socher et al. (2013). This work proposes the Recursive Neural Tensor Network (RNTN) model, and it also illustrates that RNTN outperforms previous models on both binary and fine-grained sentiment analysis. The RNTN model represents a phrase using word vectors and a parse tree, computing vectors for higher nodes in the tree using a tensor-based composition function. In relation to the ensemble schemes showed in Section 2.1, some authors (Mesnil, Mikolov, Ranzato, & Bengio, 2014) have used a geometric mean rule to combine three sentiment models: a language model approach, continuous representations of sentences and a weighted BOW. That ensemble exhibits a high performance on sentiment estimation of movie reviews, and better performance that its component classifiers.

To the best of our knowledge, a hybrid approach in which deep learning algorithms, classic feature engineering and ensemble tech-

Table 1
Proposed taxonomy for ensemble of surface and Deep features. S represents surface features, G and A stand for generic word vectors and affect word vectors, respectively. The combination of the features and/or word vectors is indicated with '+'. We consider the combination *No ensemble*/*S+G+A* not possible since it requires different types of features. Proposed approaches are marked with '*'.

	S	S+G	G	G+A	A	S+A	S+G+A
No ensemble	Pang et al. (2002); Read (2005) Pak and Paroubek (2010); Wang and Manning (2012) Gimpel et al. (2011); Kouloumpis, Wilson, and Moore (2011) Nasukawa and Yi (2003); Taboada et al. (2011) Melville et al. (2009); Qiu et al. (2009) Kiritchenko et al. (2014)	Su et al. (2014), Kim et al. (2013)	**M_G***, Shirani-Mehr (2012), Collobert et al. (2011)	Severyn and Moschitti (2015)	Tang, Wei, Yang et al. (2014), Socher et al. (2013)		–
Classifier ensemble	Xia and Zong (2011); Xia et al. (2011), Aue and Gamon (2005); Fersini et al. (2016), Rokach (2005); Sehgal and Song (2007), Prusa et al. (2015); Whitehead and Yaeger (2010) Fersini et al. (2014); Lin et al. (2015) Wang et al. (2014)	**CEM_{SG}***, Zhang and He (2015) Mesnil et al. (2014)					**CEM_{SGA}***
Feature ensemble	Agarwal et al. (2011); Wilson et al. (2009) Xia and Zong (2010)	**M_{SG}***		**M_{GA}***, Li et al. (2015), Vo and Zhang (2015)		Tang, Wei, Qin et al. (2014)	**M_{SGA}***

niques for sentiment analysis are used has not been thoroughly studied.

3. Ensemble taxonomyy

This section presents the proposed taxonomy for ensemble techniques applied to Sentiment Analysis in both surface and deep domains. This classification intends to summarize the work found in the literature as well as to compare these models with the ones we propose. Also, with this, we address the first question raised in Section 1 regarding how combination techniques can be classified.

The taxonomy can be expressed as combination of two different dimensions. Each dimension represents a characteristic of the studied approaches. On the one hand, one dimension considers which features are used in the model. Those features can be either surface features (which stands for *S*), generic automatic word vectors (*G*), or affect word vectors specifically trained for the sentiment analysis task (*A*). On the other hand, the other dimension attends to how the different model resources are combined. These combinations can be: using no ensemble method at all, through a ensemble of classifiers, or taking advantage of a feature ensemble. Table 1 shows a representation of this taxonomy, where the two dimensions appear as rows for the first dimension, and columns for the second dimension. We have classified all the reviewed work in this paper using the proposed taxonomy, obtaining a visual layout of the techniques that are used in each approach in relation with both ensemble methods and the combination of surface and deep features.

Regarding the dimension that tackles the ensemble techniques, in the *No ensemble* category we find the classifiers that do not make use of an ensemble technique. Under the *Classifier ensemble* category we classify the approaches that are based on ensemble techniques (Section 2.1), such as the voting rule or a meta-learning technique, to name a few. In the same manner, the *Feature ensemble* category contains the approaches that make use of feature combination techniques. The feature ensemble consists in combining different set of features into an unified set that is then fed to a learning algorithm.

As for the dimension that represents which features are used, several possibilities are represented: only surface features, generic or affect words vectors (*S, G* and *A* respectively), where only one type of feature is used. Besides, this dimension also takes into account the combination of different types of features: *S+G* (surface features combined with generic word vectors), *G+A* (generics word vectors with affect embeddings), *S+A* (surface features combined with affect word embeddings), and *S+G+A* (all three types of features combined in the same model).

These two dimensions are combined, creating a grid where the different approaches can be classified. The blank spaces in the taxonomy represent techniques that, to the extent of our knowledge, have not been studied. As such, they represent work that can be addressed in the future.

In conclusion, the introduced taxonomy provides a framework for characterizing and comparing ensemble approaches in sentiment analysis. This framework provides us with the opportunity to characterize and compare existing research works in sentiment analysis using ensemble techniques. Moreover, the framework can help us to provide guidelines to choose the most efficient and appropriate ensemble method for a specific application.

4. Sentiment analysis models

This section presents the sentiment analysis models proposed in our work. These models have been validated in the Twitter and

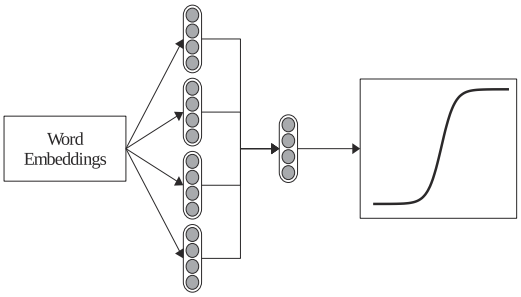


Fig. 1. Schematic representation of the baseline model, M_G. The word vectors are combined into a fixed dimension vector, and then fed to a logistic regressor, which determines the polarity of the document.

movie reviews domains (Section 5). First we describe the developed deep learning based analyzer used as baseline for the rest of the paper, and after this we detail the proposed ensemble models. These models are: ensemble of classifiers and ensemble of features. Regarding the ensemble of classifiers, we tackle two main approaches in further experiments: fixed rule and meta-learning models.

4.1. Deep learning classifier (M_G)

Generic word vectors, also denoted as pre-trained word vectors, can be captured by word embeddings techniques such as *word2vec* (Mikolov, Chen et al., 2013) and *GloVe* (Pennington, Socher, & Manning, 2014). Generic vectors are extracted in an unsupervised manner i.e., they are not trained for a specific task. These word vectors contain semantic and syntactic information, but do not enclose any specific sentiment information. Nevertheless, with the intention of exploiting the information contained in these generic word vectors, we have developed a sentiment analyzer model based on deep word embedding techniques for feature extraction, in order to compare it to other approaches in the task of Sentiment Analysis. The computed word vectors are combined into a unique vector of fixed dimension that represents the whole message. Then, this vector is fed to a logistic regression algorithm. The computation of the combined vector can be made using a set of convolutional functions, or using a embedding that transforms documents into a vector. In this way, the proposed baseline model codification is input dependent, as can be seen in Section 5. In this paper, we propose the use of word2vec for short texts, where the word vectors of the text are combined with convolutional functions; and doc2vec for long ones, representing each document with a vector. The combination of word vectors from short texts are obtained through the *min, average* and *max* convolutional functions. These functions may be combined through the concatenation of its resulting vectors. The combination of n of these functions produces a vector of nd dimensions, where d is the original dimension of the word vectors.

A diagram of this model is shown in Fig. 1. We refer to this model as M_G, with the G standing for generic word vectors.

4.2. Ensemble of classifiers (CEM)

Our objective is to combine the information from surface and deep features. The most straightforward method is to combine them at the classification level. In this way, we propose an ensemble model which combines classifiers trained with deep and surface features. Thus, knowledge from the two sets of features is combined, and this composition has more information than its base components. This model combines several base classifiers which make predictions from the same input data. These predictions can be subsequently used as new data for extracting a single prediction of sentiment polarity. This ensemble model aims to improve the sentiment classification performance that each base classifier can achieve individually, obtaining better performance. There are many possibilities for the combination of the base classifiers predictions that outputs a final sentiment polarity (e.g. a fixed rule or a meta learning technique). Also, any number of base classifiers can be combined into this ensemble model. A schematic diagram of this proposal is illustrated in Fig. 2. We denote this model as *CEM*, which stands for Classifier Ensemble Model. The subscript indicates the types of features its base classifiers have been trained with, like in CEM_{SG}, where the ensemble combines classifiers trained with surface features and generic word vectors.

Next, the two ensemble techniques used in this ensemble model in the experimentation section are further described.

4.2.1. Fixed rule model

This model seeks to combine the predictions from different classifiers using a simple fixed rule. Consequently, this ensemble does not need to learn from examples. The rules used in this approach can be any fixed rule used in ensemble models. In this work the rule used for the ensemble is the voting rule by majority. This rule counts the predictions of component classifiers and assigns the input to the class with most component predictions. In case of a match, a fixed class can be selected as the predicted by the model.

4.2.2. Meta classifier model

In the meta-classifier technique, the outputs of the component classifiers are treated as features for a meta-learning model. One advantage of this approach is that it can learn, i.e. adapt to different situations. As for the selection of the learning algorithm of this approach, there is no indication as of which one should be used. In this work, we select the Random Forest algorithm, as it can achieve high performance metrics in sentiment analysis (da Silva, Hruschka, & Hruschka, 2014; Zhang et al., 2011).

4.3. Ensemble of features (M_{SG} and M_{GA})

This model is proposed with the aim of combining several types of features into a unified feature set and, consequently, combine the information these features give. In this way, a learning model that learns from this unified set could achieve better performance scores that one that learns from a feature subset.

In this sense, we can distinguish two main types of ensembles of features. The first type is the ensemble of features that combines both surface and deep learning features. We address to this first model type as M_{SG}, as it combines surface features and generic word vectors. The second type consists on an ensemble of features that were completely extracted using deep learning techniques. This second type is referred as M_{GA}, combining both generic and affect word vectors. We refer to affect vectors as the result from training a set of pre-trained word vectors for a specific task, which in this case it would be SA.

Additionally, we also propose a third feature ensemble model, where all the three types of features are combined. This model, where surface features, generic word vectors and affect word vectors are combined is denoted by M_{SGA}. A diagram representing two instances of the model is shown in Fig. 2.

5. Experimental study

This section describes the experiments conducted in order to answer the questions formulated in the introduction (Section 1).

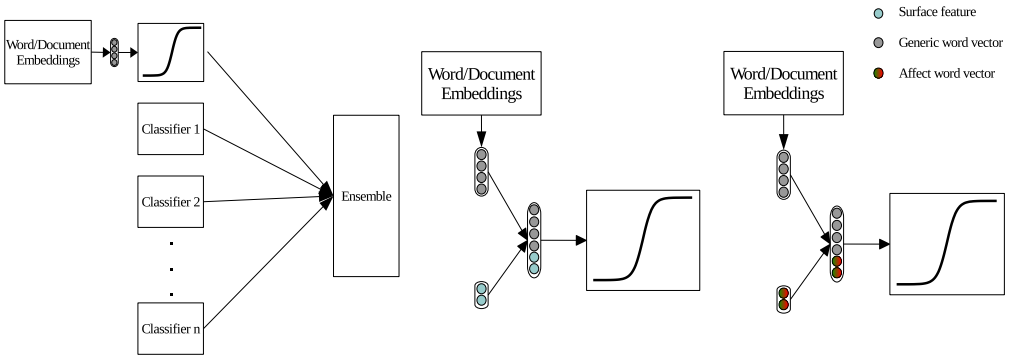


Fig. 2. Diagram of how the different classifiers and features are combined in the CEM_{SG} (Classifier Ensemble Model combining surface features and generic word vectors), M_{SG} and M_{GA} models.

Table 2
Statistics of the SemEval2014/2014, Vader, STS-Gold and Sentiment140 datasets.

Dataset	Positive	Negative	Total	Average #words
SemEval2013	2315	861	3176	23
SemEval2014	2509	932	3441	22
Vader	2901	1299	4200	16
STS-Gold	632	1402	2034	16
Sentiment140	800,000	800,000	1,600,000	15
IMDB	25,000	25,000	50,000	255
PL04	1000	1000	2000	723

Each performance experiment is made with six different datasets, widely used by the community of Sentiment Analysis. The metric used in this work is the macro averaged F1-Score. Accuracy, Precision and Recall are also computed for all the experiments, and the interested reader can find these results in the web[1]. We also publish the computed vectors that have been used in the deep models.

These experiments (Section 6.2) are aimed to compare the performance between the deep learning baseline we have developed (M_G) and the proposed ensemble models. Also, some experiments (Section 6.1) are also aimed to characterize the sentiment analysis performance for each individual classifier of the CEM models. For the last purpose, we have collected several sentiment analyzers for composing a classifier ensemble.

As for the sentiment analysis of natural language, it is conducted at the message level, so it is not necessary to split the input data into sentences. The classifiers label each comment as either positive or negative.

5.1. Datasets

The datasets used for testing are *SemEval 2013, SemEval 2014* (Rosenthal, 2014), *Vader* (Hutto, 2015), *STS-Gold* (Saif, Fernandez, He, & Alani, 2013), *IMDB* (Mass et al., 2011) and *PL04* (Pang et al., 2002). Also, we use the *Sentiment 140* (Go, Bhayani, & Huang, 2009) and IMDB datasets for training and developing our deep learning baseline, M_G. These datasets are described next, and some statistics are summarized in Table 2.

The *SemEval 2013* test corpus is composed of English comments extracted from Twitter on a range of topics: several entities, products and events. Similarly, we have also use the *SemEval 2014* test dataset. In both *SemEval* datasets, the data is not public but must be downloaded from the source first. As some users have already deleted their comments online, we have not been able to recover the original datasets, but subsets of it. Besides, since the development dataset contains only binary targets (positive and negative), we have made an alignment processing of the *SemEval* datasets, filtering other polarity values. The obtained sizes are detailed in Table 2.

The *Vader* dataset contains 4200 tweet-like messages, originally inspired by real Twitter comments. A subset of these messages is specifically designed to test some syntactical and grammatical features that appear in the natural language. The *STS-Gold* dataset for Twitter, which has been collected as a complement for Twitter sentiment analysis evaluations processes (Saif et al., 2013).

As for the training data of our Twitter baseline model, the selected dataset is the *Sentiment 140* dataset, containing 1,600,000 Twitter messages extracted using a distant supervision approach (Go et al., 2009). The abundance of data in this dataset is very beneficial to our deep learning approach, as it requires large quantities of data to extract a fairly good model, as pointed out by Mikolov, Chen et al. (2013).

Regarding the movie reviews domain, *IMDB* contains 50,000 polarized messages, using the score of each review as a guide for the polarity value. Besides, this dataset contains 50,000 unlabeled messages that have been used for training the movie reviews baseline model. We use this dataset for the training of the movie reviews baseline model. The *PL04* dataset is a well-known dataset in this domain. For the results in this dataset, we report the 10-fold cross validation metrics using the authors' public folds, in order to make our results comparable with the ones found in the literature.

5.2. Baseline training

Due to the different characteristics of the two studied domains (Twitter and movie reviews), the vector computing process for the baseline model has been made differently. In the Twitter domain, the word vectors computed by word2vec are combined using convolutional functions. For the movie reviews domain, doc2vec is used for the combination of the word vectors. For the implementation of this model, we use the gensim library (Řehůřek & Sojka, 2010).

We found that the use of the convolutional functions in large text documents does not yield better performance than doc2vec

[1] http://gsi.dit.upm.es/~oaraque/enhancing-dl.

Table 3
Effectiveness of the convolutional functions on the Sentiment140 development dataset.

Convolutional function	F-Score
max	74.82
avg	77.53
min	74.99
max + *avg*	77.63
max + *min*	76.7
avg + *min*	77.70
max + *avg* + *min*	77.73

combinations. Hence, we performed an evaluation of these two approaches on the two development datasets. While the convolutional combinations yield a F1 score of 77.53% in the Sentiment140 dataset, they also achieve 73.66& in the IMDB dataset. When using doc2vec the F1 scores are 75.00% and 89.45% in the Sentiment140 and IMDB datasets respectively. Considering the average number of words presented in Table 2 and the difference on the performance of each approach, we use the convolutional functions for the twitter domain, where short texts are analyzed and the doc2vec technique for the movie reviews domain, which contains large documents.

Regarding the training process for the short text word embeddings, we empirically fixed the dimension of the word vectors generated to 500. We use 1,280,000 tweets randomly selected from the Sentiment 140 dataset. Once this model is extracted, we feed a logistic regression model (implementation from scikit-learn) with the vectors of each tweet and the labels from the original dataset. The movie reviews baseline has been similarly trained, with the 50,000 unsupervised documents of the IMDB dataset, setting the dimension of the document vectors to 100. The same linear model is use for the classification of the document vectors. All the performance metrics have been obtained using K-fold cross validation, with folds of 10.

With respect to the convolutional functions, we have conducted an effectiveness test of the *max, average* and *min* functions on the Sentiment140 development set. The results are shown in Table 3. As can be seen, the *avg* function is very close to the performance of the complete set of functions *max, avg* and *min*. Consequently, we select the *avg* function as the one used for further experiments, as it provides very good results compared to the rest, and it also reduces the computational complexity of the experimentation. No pooling functions are used in the movie reviews, as there is no need to combine different word vectors.

Lastly, the preprocessing of natural language, we tokenized the input data and removed punctuation, excepting the most common (‘.,!?’). We also transformed URLs, numbers and usernames (@username) into especial characters to normalize the data. The preprocessing is applied to all the texts before generating the word vectors.

5.3. Ensemble of classifiers

In order to improve the performance of the deep learning baseline, we have built an ensemble composed of this analyzer and six different sentiment classifiers. Following, a list and a brief description of each of these classifiers is shown:

- sentiment140 (Go et al., 2009). It uses Naive Bayes, Maximum Entropy and Support Vector Machines trained with unigrams, bigrams and POS features.
- Stanford CoreNLP (Manning et al., 2014) is the RNTN approach shown in Section 2.2, proposed by Socher et al. (2013).

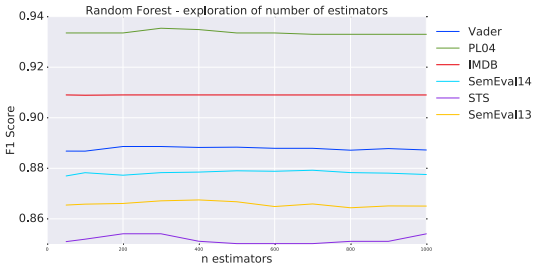


Fig. 3. Cross validation of the number of estimators on the Random Forest algorithm used for the meta-learning ensemble.

- Sentiment WSD (Kathuria, 2015), which uses SentiWordNet (Esuli & Sebastiani, 2006), performing the sentiment estimation based on the polarities of each word.
- Vivekn (Narayanan, Arora, & Bhatia, 2013). It is based in a Naive Bayes classifier trained with word n-grams and using several techniques, such as negation handling, feature selection and laplacian smoothing.
- pattern.en (De Smedt & Daelemans, 2012) uses a Support Vector Machines algorithm fed with polarity and subjectivity values for each word, WordNet vocabulary information and POS annotation.
- TextBlob Sentiment Classifier (Loria, 2016), a modular approach which as default configuration uses a Naive Bayes classifier trained with unigram features.

We have built ensemble classifiers using two combining techniques in the CEM model: a rule based method and a meta learning approach, both using the predictions of the classifiers composing the ensemble as features for the next step. For the meta-learning approach, we use the implementation of scikit-learn of the Random Forest algorithm. For this algorithm we have used 100 as the default number of estimators. As is shown in Fig. 3, the value of this parameter does not affect to the classification performance in the range from 50 to 1000.

Additionally, two versions of the CEM model have been implemented for the experiments. While the CEM_{SG} combines the six aforementioned classifiers and the M_G model; the CEM_{SGA} version combines the base classifiers from CEM_{SG} with the M_{SG}, $M_{SG+bigrams}$ and M_{GA} models.

5.4. Ensemble of features

Based on the work by Mohammad, Kiritchenko, and Zhu (2013), we have selected the following surface features: SentiWordnet (Esuli & Sebastiani, 2006) lexicon values for each word, as well as total number of positive, neutral and negative words extracted with this lexicon; number of exclamation, interrogation and hashtags marks ‘!?#’; number of words that are all in caps and number of words that have been elongated ‘goooooood’. This feature set has been cross validated on the development sets, with the objective of obtaining the smaller surface feature set that yields the best classification performance.

With the aim of complementing the surface features, we have also explored the role of n-grams. More specifically, bigrams are used, as the introduction of unigrams and trigrams did not improve the classification performance. The ensemble of features model that includes generic word vectors, the described surface features and bigrams is represented as $M_{SG+bigrams}$.

As for the M_{GA} model, we use the word vectors obtained by Tang, Wei, Yang et al. (2014). More specifically, we use the vectors extracted using the $SSWE_r$ neural model. These vectors have been

Table 4
Macro averaged F-Score of all the base sentiment classifiers. TB represents the TextBlob classifier.

Dataset	sent140	CoreNLP	WSD	vivekn	pattern	TB
SemEval2013	78.92	46.95	76.18	72.14	82.50	82.51
SemEval2014	60.67	42.95	75.35	59.97	71.86	71.92
Vader	78.76	60.19	77.75	63.54	85.98	85.71
STS-Gold	75.07	59.68	75.35	69.61	82.86	68.27
IMDB	72.91	88.56	87.41	87.45	75.43	75.47
PL04	71.66	86.09	79.03	86.22	70.64	70.82

extracted for learning sentiment-specific word vectors but not general semantic information, so we use them as affect word vectors. Also, for the composition of a tweet vector, we have used the *average* function on the word vectors, as the combination of the other convolutional functions did not improve the results of the model.

6. Results

The conducted experiments show the sentiment classification performance of each base classifier separately (including our deep learning baseline) on each of the six test datasets, as well as the metrics for the ensemble of classifiers and ensembles of features. In this section, the experimental results are shown and discussed. The experimental results for the proposed models are gathered in Table 5.

6.1. Base classifiers performance

As it can be seen in Table 4, the better F-score performance is achieved by TextBlob in SemEval2013, by the WSD classifier with an important difference over TextBlob in SemEval2014; while pattern.en has a slightly better performance than the rest in the Vader dataset and has also the better performance in the STS-Gold dataset by far. The classifier with the lower performance is CoreNLP in all Twitter test sets. In contrast, in the IMDB and PL04 datasets (movie reviews), CoreNLP achieves the better performance in IMDB and the second best in PL04.

As expected, the nature of the domains has a strong impact on the performance of the base classifiers, since they have been trained in a specific domain. Some classifiers (e.g. WSD, pattern and TextBlob) are more suitable to the Twitter domain, while others (e.g. CoreNLP and vivekn) are better adapted to the review domain. In general, none of the base classifiers exhibits a high performance in all the baseline datasets.

Finally, the average F1 score performance for the base classifiers is 73.02, 63.79, 75.32, 71.81, 81.20 and 77.41% in the SemEval2013, SemEval2014, Vader, STS-Gold, IMDB and PL04 datasets respectively.

6.2. Classifiers and features classifiers performance

CEM models gather the predictions from M_G baseline and the other six base classifiers whose classification performance has been analyzed. The voting and meta-learning techniques are used as ensemble techniques. It can be seen in the Table 5 that nearly all the ensemble models surpass the baseline, as well as all the other base classifiers. In fact, the best performance is achieved in 4 out of 6 datasets by these CEM classifiers.

As for the feature ensemble models, they also push the performance further than the baseline. The $M_{SG+bigrams}$ ensemble is very close to the best metrics in almost all the test datasets. Also, it seems that $M_{SGA+bigrams}$ is suffering overfitting, as the combination of all three types of features decreases the performance when comparing to $M_{SG+bigrams}$ model. This could be due to the increase in the number of features used to train the model.

Moreover, as an additional observation, it can be seen that the better improvements are achieved by CEM^{Vo}_{SGA} and CEM^{MeL}_{SG} models, with 3.65 and 2.53% of performance gain in STS-Gold and SemeEval2013 datasets respectively, and by CEM^{MeL}_{SGA} model in the IMDB and PL04 datasets, with improvements of 1.48% and 5.77% respectively.

Although the biggest improvements have been achieved with CEM models, the feature ensemble models also improve the baseline, sometimes by a large margin. Considering this type of models, the better results are achieved by the $M_{SG+bigrams}$ model. This fact could be explained attending to fact that bigrams can successfully capture modified verbs and nouns (Wang & Manning, 2012), such as the negation.

Also, the M_{SG} model results are comparable, generally, to the best performances in the Twitter domain. Nevertheless, this model does not yield such results in the movie reviews domain. This result indicates that combining word vectors through convolutional functions in long texts does not lead to high sentiment classification performances. We can conclude that the transformation of the convolutional functions on the sentiment signals that are contained in the affect word vectors is retained when applied to short texts, but lost in long texts.

Attending to the difference of performance between the $M_{SGA+bigrams}$ and the CEM^{Vo}_{SGA} and CEM^{MeL}_{SGA} models, we see that the same types of features do not yield the same result. We make the assumption that dividing the features into smaller sets, as it is done in the ensemble models, benefits the classification performance. The division is made at the classifier level, since these ensemble models combine the predictions of classifiers trained with features (e.g. surface, generic and affect). Considering that the whole set SGA (including bigrams)is a complex collection of features, and based on the experimental results, the assumption is that this division of features prevents overfitting. These results are in line with those by Alpaydin (2014) and Xia et al. (2011). This shows that when dividing a complex set of features into simpler subsets, an ensemble can yield better performance.

6.3. Statistical analysis

In order to compare the different proposed models in this work, a statistical test has been applied on the experimental results. Concretely, the Friedman test with the corresponding Bonferroni-Dun

Table 5
Macro averaged F-Score of the proposed sentiment classifiers. The last row shows the Friedman rank. In bold the best classifier for each dataset, and in the last row the best classifier attending to the Bonferroni-Dunn test.

Dataset	M_G	CEM^{Vo}_{SG}	CEM^{MeL}_{SG}	M_{SG}	$M_{SG+bigrams}$	M_{GA}	$M_{SGA+bigrams}$	CEM^{Vo}_{SGA}	CEM^{MeL}_{SGA}
SemEval2013	85.34	87.78	**87.87**	86.36	86.53	87.54	86.26	86.26	86.97
SemEval2014	86.16	84.16	87.63	87.03	**88.19**	88.05	86.94	85.90	88.07
Vader	87.71	87.92	88.85	88.07	88.93	88.89	88.89	**89.52**	89.48
STS-Gold	83.43	83.52	84.56	84.73	**89.24**	85.27	85.26	87.08	85.59
IMDB	89.45	84.06	89.68	89.58	90.41	89.50	90.41	89.92	**90.93**
PL04	88.72	86.48	93.65	88.39	94.33	88.67	86.76	93.87	**94.49**
Friedman rank	7.83	7.5	4.5	6.17	**2.67**	4.33	5.58	4.25	**2.17**

post-hoc test, that are described by Demšar (2006). These tests are specially oriented to the comparison of several classifiers on multiple data sets.

Friedman's test is based on the rank of each algorithm in each dataset, where the best performing algorithm gets the rank of 1, the second best gets rank 2, etc. Ties are resolved by averaging their ranks. r_j^i is the rank of the jth of the k algorithms and on the i-th of N datasets. Friedman test uses the comparison of average ranks $R_j = \frac{1}{N}\sum_i r_i^j$. and states that under the null-hypothesis (all the algorithms are equal so their ranks R_j are equal) the Friedman statistic, with $k-1$ degrees of freedom, is:

$$\chi_F^2 = \frac{12N}{k(k+1)}\left(\sum_j R_j^2 - \frac{k(k+1)^2}{4}\right)$$

Nevertheless, Demšar (2006) shows that there is a better statistic that is distributed according to the F-distribution, and has $k-1$ and $(k-1)(N-1)$ degrees of freedom:

$$F_F = \frac{(N-1)\chi_F^2}{N(k-1)-\chi_F^2}$$

If the null-hypothesis of the Friedman test is rejected, post-hoc tests can be conducted. In this work we employ the Bonferroni-Dunn test, as it allows to compare the results of several algorithms to a baseline. In this case, all the proposed models are compared against M_G. This test can be computed through comparing the critical difference (CD) with a series of critical values (q_α), which Demšar (2006) summarizes. The critical difference can be computed as:

$$CD = q_\alpha\sqrt{\frac{k(k+1)}{6N}}$$

For the computation of both tests, the ranks have been obtained. The average ranks (R_j) are showed in Table 5. The α values is set to 0.05 for the following calculations. With those averages, $\chi_F^2 = 24.56$, $F_F = 5.24$, and the critical value $F(k-1,(k-1)(N-1)) = 2.18$. As $F_F > F(8, 40)$, the null-hypothesis is rejected and the post-hoc test can be conducted. According with this, the critical difference is 4.31. Following, the difference between the average ranks of the baseline and the jth model is compared to the CD and, if greater, we can conclude that the jth algorithm performs significantly better than the baseline.

Friedman's test has pointed the CEM_{SGA}^{MeL} and the $M_{SG+bigrams}$ models as the two best classification models, followed by the CEM_{SGA}^{Vo} and M_{GA} models. Besides, the Bonferroni-Dunn test points out that CEM_{SGA}^{MeL} and $M_{SG+bigrams}$ models perform significantly better that the baseline. These results indicate that the hypothesis suggested in question 2 is supported, since the combination of different sources of information improves the performance of sentiment analysis. As for the rest of the models, it is not possible to reach a conclusion with the current data.

On top of this, an interesting result of these experiments is that the performance of the meta learning approach is higher than that of the fixed rule scheme. While the meta learning ensemble with all types of features (SGA + bigrams) is significantly better than the baseline, the voting model is not. This could be caused by the learning capabilities of the meta-classifier technique, feature that the fixed ensemble methods like voting rule do not have.

6.4. Computational complexity

One possible drawback of ensemble approaches is their higher cost in terms of computational resources. With the aim of analyzing the efficiency of the proposed models, the computational cost is presented. The results for this cost analysis are studied at train time, since the costs in the test phase do not result relevant for this analysis. All these measurements were made in a Intel Xeon with 12 cores available and a memory friendly environment.

Table 6
Computational complexity of the word embeddings approaches, in both model training and vector computation.

Word Embeddings approach		Sentiment140	IMDB
word2vec	Train model	109.7 s	96.1 s
	Compute vectors	152.9 s	80.4 s
doc2vec	Train model	7 h	2 h
	Compute vectors	8.5 s	6.4 s
SSWE	Train model	–	25 d
	Compute vectors	87.5 s	179.2 s

Table 7
Computational complexity of the proposed models in training time.

Model	Sentiment140	IMDB
M_G	12.7 s	0.9 s
M_{SG}	13.7 s	0.9 s
$M_{SG+bigrams}$	977.5 s	37.2 s
M_{GA}	12.9 s	1 s
$M_{SGA+bigrams}$	977.8 s	37.4 s

In relation to the training and computation of the word embeddings approaches, Table 6 presents the associated computational cost. It can be seen that word2vec is the lighter model at train time by a large margin. Also, the implementation of SSWE largely increases the computational complexity. We believe that implementing this model for a GPU environment can have a great impact on the time performance of the SSWE training. Please note that the SSWE trained model for Twitter is available for research, and so the training using the Sentiment140 dataset has not been performed. Besides, we can see that computing the pooling functions on the word vectors increases the complexity, as can be seen by comparing with the doc2vec approach.

Combining different sets of features increases the computational complexity, as Table 7 shows. The largest increment can be found in the $M_{SG+bigrams}$ and $M_{SGA+bigrams}$ models, which use bigrams in the learning process. In this way, it can be seen that using feature ensemble with bigrams and other sets of features leads to the addition of complexity to the model. The difference of training times between the Sentiment140 and IMDB datasets is due to their number of instances, being larger in the first.

Finally, the CEM models do not introduce a relevant complexity to the model at training time. The ensemble of classifiers based on the voting scheme do hardly introduce a cost to the computation, as there is no learning process in this case. For the meta-learning scheme, the maximum time taken in the meta learning process is 1.5 s, with no significant difference between the training data.

7. Conclusions and future work

This paper proposes several models where classic hand-crafted features are combined with automatically extracted embedding features, as well as the ensemble of analyzers that learn from these varied features. In order to classify these different approaches, we propose a taxonomy of ensembles of classifiers and features that is based on two dimensions. Furthermore, with the aim of evaluating the proposed models, a deep learning baseline is defined, and the classification performances of several ensembles are compared to the performance of the baseline. With the intention of conducting a comparative experimental study, six public datasets are used for the evaluation of all the models, as well as six public sentiment classifiers. Finally, we conduct an statistical analysis in order to empirically verify that combining information from varied fea-

tures and/or analyzers is an adequate way to surpass the sentiment classification performance.

There were three main research questions that drove this work. The first question was whether there is a framework to characterize existing approaches in relation to the ensemble of traditional and deep techniques in sentiment analysis. To the best of our knowledge, our proposal of a taxonomy and the resulting implementations is the first work to tackle this problem for sentiment analysis.

The second question was whether the sentiment analysis performance of a deep classifier can be leveraged when using the proposed ensemble of classifiers and features models. Observing the scores table and the Friedman ranks (Table 5), we see that the proposed models generally improve the performance of the baseline. This indicates that the combination of information from diverse sources such as surface features, generic and affect word vectors effectively improves the classifier's results in sentiment analysis tasks.

Lastly, we raised the concern of which of the proposed models stand out in the improvement of the deep sentiment analysis performance. In this regard, the statistical results point out the CEM^{MeL}_{SGA} and $M_{SG+bigrams}$ models as the best performing alternatives. As expected, these models effectively combine different sources of sentiment information, resulting in a significant improvement with respect to the baseline. We remark the $M_{SG+bigrams}$ model, as it does not involve an ensemble of many classifiers, but only a classifier that is trained with an ensemble of deep and surface features.

To summarize, this work takes advantage of the ensemble of existing traditional sentiment classifiers, as well as the combination of generic, sentiment-trained word embeddings and manually crafted features. Nevertheless, Considering the results of this work, we believe that a possible line of work would be applying these models to the task of aspect based sentiment analysis, with the hope of improving the classification performance. Furthermore, we intend to extend the domain of the proposed models to other languages and even paradigms, like Emotion analysis.

Acknowledgements

This research work is supported by the EC through the H2020 project MixedEmotions (Grant Agreement no: 141111), the Spanish Ministry of Economy under the R&D project Semola (TEC2015-68284-R) and the project EmoSpaces (RTC-2016-5053-7); by ITEA3 project SOMEDI (15011); and by MOSI-AGIL-CM (grant P2013/ICE-3019, co-funded by EU Structural Funds FSE and FEDER).

References

Agarwal, A., Xie, B., Vovsha, I., Rambow, O., & Passonneau, R. (2011). Sentiment analysis of twitter data. In *Proceedings of the workshop on languages in social media* (pp. 30–38). Association for Computational Linguistics.

Alpaydin, E. (2014). *Introduction to machine learning*. MIT press.

Aue, A., & Gamon, M. (2005). Customizing sentiment classifiers to new domains: A case study. In *Proceedings of recent advances in natural language processing (RANLP): vol. 1*. 2–1

Bengio, Y. (2009). Learning deep architectures for ai. *Foundations and Trends® in Machine Learning, 2*, 1–127.

Cambria, E. (2016). Affective computing and sentiment analysis. *IEEE Intelligent Systems, 31*, 102–107.

Chen, M., Weinberger, K. Q., Sha, F., & Bengio, Y. (2014). Marginalized denoising auto-encoders for nonlinear representations. In *ICML* (pp. 1476–1484).

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., & Kuksa, P. (2011). Natural language processing (almost) from scratch. *The Journal of Machine Learning Research, 12*, 2493–2537.

De Smedt, T., & Daelemans, W. (2012). Pattern for python. *The Journal of Machine Learning Research, 13*, 2063–2067.

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine Learning Research, 7*, 1–30.

Ding, X., Liu, B., & Yu, P. S. (2008). A holistic lexicon-based approach to opinion mining. In *Proceedings of the 2008 international conference on web search and data mining WSDM '08* (pp. 231–240). New York, NY, USA: ACM. doi:10.1145/1341531.1341561.

Esuli, A., & Sebastiani, F. (2006). Sentiwordnet: A publicly available lexical resource for opinion mining. In *Proceedings of LREC: 6* (pp. 417–422). Citeseer.

Fersini, E., Messina, E., & Pozzi, F. (2014). Sentiment analysis: Bayesian ensemble learning. *Decision Support Systems, 68*, 26–38. doi:10.1016/j.dss.2014.10.004. http://www.sciencedirect.com/science/article/pii/S0167923614002498.

Fersini, E., Messina, E., & Pozzi, F. (2016). Expressive signals in social media languages to improve polarity detection. *Information Processing & Management, 52*, 20–35.

Gimpel, K., Schneider, N., O'Connor, B., Das, D., Mills, D., Eisenstein, J., et al. (2011). Part-of-speech tagging for twitter: Annotation, features, and experiments. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies: Short papers - volume 2 HLT '11* (pp. 42–47). Stroudsburg, PA, USA: Association for Computational Linguistics. http://dl.acm.org/citation.cfm?id=2002736.2002747.

Go, A., Bhayani, R., & Huang, L. (2009). Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford, 1*, 12.

Hutto, C. (2015). *Vader sentiment github repository*. https://github.com/cjhutto/vaderSentiment. Accessed on May 30, 2016.

Kathuria, P. (2015). *Sentiment wsd github repository*. https://github.com/kevincobain2000/sentiment_classifier/. Accessed on May 30, 2016.

Kim, J., Yoo, J.-B., Lim, H., Qiu, H., Kozareva, Z., & Galstyan, A. (2013). Sentiment prediction using collaborative filtering.. *ICWSM*.

Kim, Y. (2014). *Convolutional neural networks for sentence classification*. arXiv preprint arXiv:1408.5882.

Kiritchenko, S., Zhu, X., & Mohammad, S. M. (2014). Sentiment analysis of short informal texts. *Journal of Artificial Intelligence Research*, 723–762.

Kouloumpis, E., Wilson, T., & Moore, J. D. (2011). Twitter sentiment analysis: The good the bad and the omg!. *Icwsm, 11*, 538–541.

Le, Q. V., & Mikolov, T. (2014). Distributed representations of sentences and documents.. In *ICML: vol. 14* (pp. 1188–1196).

Li, C., Xu, B., Wu, G., He, S., Tian, G., & Zhou, Y. (2015). Parallel recursive deep model for sentiment analysis. In T. Cao, E.-P. Lim, Z.-H. Zhou, T.-B. Ho, D. Cheung, & H. Motoda (Eds.), *Advances in knowledge discovery and data mining*. In *Lecture Notes in Computer Science: Vol. 9078* (pp. 15–26). Springer International Publishing. doi:10.1007/978-3-319-18032-8_2. http://dx.doi.org/10.1007/978-3-319-18032-8_2.

Lin, Y., Wang, X., Li, Y., & Zhou, A. (2015). Integrating the optimal classifier set for sentiment analysis. *Social Network Analysis and Mining, 5*, 1–13.

Liu, B. (2015). *Sentiment analysis: Mining opinions, sentiments, and emotions*. Cambridge University Press.

Loria, S. (2016). *Textblob documentation page*. https://textblob.readthedocs.org/en/dev/index.html. Accessed on May 30, 2016.

Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., & Potts, C. (2011). Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1* (pp. 142–150). Association for Computational Linguistics.

Maclin, R., & Opitz, D. (1997). An empirical evaluation of bagging and boosting. *AAAI/IAAI, 1997*, 546–551.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., & McClosky, D. (2014). The stanford CoreNLP natural language processing toolkit. In *Association for computational linguistics (ACL) system demonstrations* (pp. 55–60). http://www.aclweb.org/anthology/P/P14/P14-5010.

Melville, P., Gryc, W., & Lawrence, R. D. (2009). Sentiment analysis of blogs by combining lexical knowledge with text classification. In *Proceedings of the 15th ACM SIGKDD international conference on knowledge discovery and data mining KDD '09* (pp. 1275–1284). New York, NY, USA: ACM. doi:10.1145/1557019.1557156. http://doi.acm.org/10.1145/1557019.1557156.

Melville, P., Shah, N., Mihalkova, L., & Mooney, R. J. (2004). Experiments on ensembles with missing and noisy data. In F. Roli, J. Kittler, & T. Windeatt (Eds.), *Multiple classifier systems: 5th international workshop, MCS 2004, Cagliari, Italy, june 9–11, 2004. proceedings* (pp. 293–302). Berlin, Heidelberg: Springer Berlin Heidelberg. doi:10.1007/978-3-540-25966-4_29. http://dx.doi.org/10.1007/978-3-540-25966-4_29.

Mesnil, G., Mikolov, T., Ranzato, M., & Bengio, Y. (2014). Ensemble of generative and discriminative techniques for sentiment analysis of movie reviews. *arXiv preprint arXiv:1412.5335*.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mikolov, T., Yih, W.-t., & Zweig, G. (2013). Linguistic regularities in continuous space word representations.. In *HLT-NAACL* (pp. 746–751).

Mohammad, S. M., Kiritchenko, S., & Zhu, X. (2013). Nrc-canada: Building the state-of-the-art in sentiment analysis of tweets. *CoRR, abs/1308.6242*. http://arxiv.org/abs/1308.6242.

Narayanan, V., Arora, I., & Bhatia, A. (2013). Fast and accurate sentiment classification using an enhanced naive bayes model. *CoRR, abs/1305.6143*. http://arxiv.org/abs/1305.6143.

Nasukawa, T., & Yi, J. (2003). Sentiment analysis: Capturing favorability using natural language processing. In *Proceedings of the 2nd international conference on Knowledge capture* (pp. 70–77). ACM.

Pablos, A. G., Cuadros, M., & Rigau, G. (2016). A comparison of domain-based word polarity estimation using different word embeddings. In N. C. C. Chair), K. Choukri, T. Declerck, S. Goggi, M. Grobelnik, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, & S. Piperidis (Eds.), *Proceedings of the tenth international conference on language resources and evaluation (LREC 2016)*. Paris, France: European Language Resources Association (ELRA).

Pak, A., & Paroubek, P. (2010). Twitter as a corpus for sentiment analysis and opinion mining.. In *LREc: vol. 10* (pp. 1320–1326).

Pang, B., Lee, L., & Vaithyanathan, S. (2002). Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on empirical methods in natural language processing-Volume 10* (pp. 79–86). Association for Computational Linguistics.

Pennington, J., Socher, R., & Manning, C. D. (2014). Glove: Global vectors for word representation.. In *EMNLP: vol. 14* (pp. 1532–1543).

Perez-Rosas, V., Banea, C., & Mihalcea, R. (2012). Learning sentiment lexicons in spanish.. In *LREC: vol. 12* (p. 73).

Polanyi, L., & Zaenen, A. (2006). Computing attitude and affect in text: Theory and applications. In *chapter Contextual Valence Shifters* (pp. 1–10)). Dordrecht: Springer Netherlands. http://dx.doi.org/10.1007/1-4020-4102-0_1.

Prusa, J., Khoshgoftaar, T., & Dittman, D. (2015). Using ensemble learners to improve classifier performance on tweet sentiment data. In *Information reuse and integration (IRI), 2015 IEEE international conference on* (pp. 252–257). doi:10.1109/IRI.2015.49.

Qiu, G., Liu, B., Bu, J., & Chen, C. (2009). Expanding domain sentiment lexicon through double propagation.. In *IJCAI: vol. 9* (pp. 1199–1204).

Read, J. (2005). Using emoticons to reduce dependency in machine learning techniques for sentiment classification. In *Proceedings of the ACL student research workshop* (pp. 43–48). Association for Computational Linguistics.

Řehůřek, R., & Sojka, P. (2010). Software framework for topic modelling with large corpora. In *Proceedings of the LREC 2010 workshop on new challenges for NLP frameworks* (pp. 45–50). Valletta, Malta: ELRA. http://is.muni.cz/publication/884893/en.

Rokach, L. (2005). Ensemble methods for classifiers. In O. Maimon, & L. Rokach (Eds.), *Data mining and knowledge discovery handbook* (pp. 957–980). Springer US. http://dx.doi.org/10.1007/0-387-25465-X_45.

Rosenthal, S. (2014). *Semeval 2014 task 9 description*. http://alt.qcri.org/semeval2014/task9/. Accessed on May 30, 2016.

Saif, H., Fernandez, M., He, Y., & Alani, H. (2013). *Evaluation datasets for twitter sentiment analysis: a survey and a new dataset, the sts-gold*.

Sehgal, V., & Song, C. (2007). Sops: Stock prediction using web sentiment. In *Data mining workshops, 2007. ICDM workshops 2007. seventh IEEE international conference on* (pp. 21–26). doi:10.1109/ICDMW.2007.100.

Severyn, A., & Moschitti, A. (2015). Twitter sentiment analysis with deep convolutional neural networks. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 959–962). ACM.

Sharma, A., & Dey, S. (2012). A comparative study of feature selection and machine learning techniques for sentiment analysis. In *Proceedings of the 2012 ACM Research in Applied Computation Symposium* (pp. 1–7). ACM.

Shirani-Mehr, H. (2012). *Applications of deep learning to sentiment analysis of movie reviews*.

da Silva, N. F., Hruschka, E. R., & Hruschka, E. R. (2014). Tweet sentiment analysis with classifier ensembles. *Decision Support Systems, 66*, 170–179.

Socher, R., Perelygin, A., Wu, J. Y., Chuang, J., Manning, C. D., & Ng, A. Y. (2013). Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the conference on empirical methods in natural language processing (EMNLP): 1631* (p. 1642). Citeseer.

Su, Z., Xu, H., Zhang, D., & Xu, Y. (2014). Chinese sentiment classification using a neural network tool word2vec. In *Multisensor fusion and information integration for intelligent systems (MFI), 2014 international conference on* (pp. 1–6). doi:10.1109/MFI.2014.6997687.

Taboada, M., Brooke, J., Tofiloski, M., Voll, K., & Stede, M. (2011). Lexicon-based methods for sentiment analysis. *Computational linguistics, 37*, 267–307.

Tang, D., Wei, F., Qin, B., Liu, T., & Zhou, M. (2014). Coooolll: A deep learning system for twitter sentiment classification. In *Proceedings of the 8th international workshop on semantic evaluation (semeval 2014)* (pp. 208–212).

Tang, D., Wei, F., Yang, N., Zhou, M., Liu, T., & Qin, B. (2014). Learning sentiment-specific word embedding for twitter sentiment classification.. In *ACL (1)* (pp. 1555–1565).

Turney, P. D. (2002). Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th annual meeting on association for computational linguistics* (pp. 417–424). Association for Computational Linguistics.

Vo, D.-T., & Zhang, Y. (2015). Target-dependent twitter sentiment classification with rich automatic features. In *Proceedings of the twenty-fourth international joint conference on artificial intelligence (IJCAI 2015)* (pp. 1347–1353).

Wang, G., Sun, J., Ma, J., Xu, K., & Gu, J. (2014). Sentiment classification: The contribution of ensemble learning. *Decision support systems, 57*, 77–93.

Wang, S., & Manning, C. D. (2012). Baselines and bigrams: Simple, good sentiment and topic classification. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers-Volume 2* (pp. 90–94). Association for Computational Linguistics.

Whitehead, M., & Yaeger, L. (2010). Sentiment mining using ensemble classification models. In *Innovations and advances in computer sciences and engineering* (pp. 509–514). Springer.

Wilson, T., Wiebe, J., & Hoffmann, P. (2009). Recognizing contextual polarity: An exploration of features for phrase-level sentiment analysis. *Computational linguistics, 35*, 399–433.

Xia, R., & Zong, C. (2010). Exploring the use of word relation features for sentiment classification. In *Proceedings of the 23rd international conference on computational linguistics: Posters COLING '10* (pp. 1336–1344). Stroudsburg, PA, USA: Association for Computational Linguistics. http://dl.acm.org/citation.cfm?id=1944566.1944719.

Xia, R., & Zong, C. (2011). A pos-based ensemble model for cross-domain sentiment classification.. In *IJCNLP* (pp. 614–622). Citeseer.

Xia, R., Zong, C., & Li, S. (2011). Ensemble of feature sets and classification algorithms for sentiment classification. *Information Sciences, 181*, 1138–1152. doi:10.1016/j.ins.2010.11.023. http://www.sciencedirect.com/science/article/pii/S0020025510005682.

Zhang, D., Xu, H., Su, Z., & Xu, Y. (2015). Chinese comments sentiment classification based on word2vec and svm perf. *Expert Systems with Applications, 42*, 1857–1863.

Zhang, K., Cheng, Y., Xie, Y., Honbo, D., Agrawal, A., Palsetia, D., et al. (2011). Ses: Sentiment elicitation system for social media data. In *Data mining workshops (ICDMW), 2011 IEEE 11th international conference on* (pp. 129–136). IEEE.

Zhang, P., & He, Z. (2015). Using data-driven feature enrichment of text representation and ensemble technique for sentence-level polarity classification. *Journal of Information Science, 41*, 531–549.

A.2.4 A modular architecture for intelligent agents in the evented web

Title	A modular architecture for intelligent agents in the evented web
Authors	Sánchez-Rada, J. Fernando and Iglesias, Carlos A. and Coronado, Miguel
Journal	Web Intelligence
Impact factor	SJR 2016 0.181
ISSN	1570-1263
Publisher	
Volume	15
Year	2017
Keywords	agent architecture, evented web, events, jason, web hooks
Pages	19–33
Online	http://content.iospress.com/articles/web-intelligence/web350
Abstract	The growing popularity of public APIs and technologies such as web hooks is changing online services drastically. It is easier now than ever to interconnect services and access them as a third party. The next logical step is to use intelligent agents to provide a better user experience across services, connecting services with smart automatic behaviors or actions. In other words, it is time to start using agents in the so-called Evented Web. For this to happen, agent platforms need to seamlessly integrate external sources such as web services. As a solution, this paper introduces an event-based architecture for agent systems. This architecture has been designed in accordance with the new tendencies in web programming and with a Linked Data approach. The use of Linked Data and a specific vocabulary for events allows a smarter and more complex use of events. Two use cases have been implemented to illustrate the validity and usefulness of the architecture.

A.2.5 Applying Recurrent Neural Networks to Sentiment Analysis of Spanish Tweets

Title	Applying Recurrent Neural Networks to Sentiment Analysis of Spanish Tweets
Authors	Araque, Oscar and Barbado, Rodrigo and Sánchez-Rada, J. Fernando and Iglesias, Carlos A.
Proceedings	TASS 2017: Workshop on Semantic Analysis at SEPLN
ISBN	
Volume	1896
Year	2017
Keywords	deep learning, natural language processing, recurrent neural networks, sentiment analysis
Pages	
Online	http://ceur-ws.org/Vol-1896/p8_gsi_tass2017.pdf
Abstract	This article presents the participation of the Intelligent Systems Group (GSI) at Universidad Politécnica de Madrid (UPM) in the Sentiment Analysis work- shop focused in Spanish tweets, TASS2017. We have worked on Task 1, aiming to classify sentiment polarity of Spanish tweets. For this task we propose a Recurrent Neural Network (RNN) architecture composed of Long Short-Term Memory (LSTM) cells followed by a feedforward network. The architecture makes use of two different types of features: word embeddings and sentiment lexicon values. The recurrent ar- chitecture allows us to process text sequences of different lengths, while the lexicon inserts directly into the system sentiment information. The results indicate that this feature combination leads to enhanced sentiment analysis performances.

TASS 2017: Workshop on Semantic Analysis at SEPLN, septiembre 2017, págs. 71-76

Applying Recurrent Neural Networks to Sentiment Analysis of Spanish Tweets

Aplicación de Redes Neuronales Recurrentes al Análisis de Sentimientos sobre Tweets en Español

Oscar Araque, Rodrigo Barbado, J. Fernando Sánchez-Rada y **Carlos A. Iglesias**
Intelligent Systems Group, Universidad Politécnica de Madrid
Av. Complutense 30, 28040 Madrid
o.araque@upm.es, rodrigo.barbado.esteban@alumnos.upm.es
{jfernando, carlosangel.iglesias}@upm.es

Abstract: This article presents the participation of the Intelligent Systems Group (GSI) at Universidad Politécnica de Madrid (UPM) in the Sentiment Analysis workshop focused in Spanish tweets, TASS2017. We have worked on Task 1, aiming to classify sentiment polarity of Spanish tweets. For this task we propose a Recurrent Neural Network (RNN) architecture composed of Long Short-Term Memory (LSTM) cells followed by a feedforward network. The architecture makes use of two different types of features: word embeddings and sentiment lexicon values. The recurrent architecture allows us to process text sequences of different lengths, while the lexicon inserts directly into the system sentiment information. The results indicate that this feature combination leads to enhanced sentiment analysis performances.
Keywords: Deep Learning, Natural Language Processing, Sentiment Analysis, Recurrent Neural Network, TensorFlow

Resumen: En este artículo se presenta la participación del Grupo de Sistemas Inteligentes (GSI) de la Universidad Politécnica de Madrid (UPM) en el taller de Análisis de Sentimientos centrado en tweets en Español: el TASS2017. Hemos trabajado en la Tarea 1, tratando de predecir correctamente la polaridad del sentimiento de tweets en español. Para esta tarea hemos propuesto una arquitectura consistente en una Red Neuronal Recurrente (RNN) compuesta de celdas *Long Short-Term Memory* (LSTM) seguida por una red neuronal prealimentada. La arquitectura hace uso de dos tipos distintos de características: *word embeddings* y los valores de un diccionario de sentimientos. La recurrencia de la arquitectura permite procesar secuencias de texto de distintas longitudes, mientras que el diccionario inserta información de sentimiento directamente en el sistema. Los resultados obtenidos indican que esta combinación de características lleva a mejorar los resultados en análisis de sentimientos.
Palabras clave: Aprendizaje Profundo, Procesamiento de Lenguaje Natural, Análisis de Sentimientos, Red Neuronal Recurrente, TensorFlow

1 Introduction

Recent developments in the area of deep learning are strongly impacting sentiment analysis techniques. While traditional methods based on feature engineering are still prevalent, new deep learning approaches are succeeding and reduce the need of labeled corpus and feature definition. Moreover, traditional and deep learning approaches can be combined obtaining improved results (Araque et al., 2017).

This paper describes our participation in TASS 2017 (Martínez-Cámara et al., 2017). Taller de Análisis de Sentimientos en la SEPLN (TASS) is a workshop that fosters the research of sentiment analysis in Spanish for short text such as tweets. The first task of this challenge, Task 1, consists in determining the global polarity at a message level. The dataset for the evaluation of this task

ISSN 1613-0073
Copyright © 2017 by the paper's authors. Copying permitted for private and academic purposes.

considers annotated tweets with 4 polarity labels (P, N, NEU, NONE). P stands for positive, while N means negative and NEU is neutral. It is considered that NONE means absence of sentiment polarity. This task provides a corpus, which contains a total of 1514 tweets written in Spanish, describing a diversity of subjects.

We have faced this challenge as an opportunity to evaluate how these techniques could be applied in the TASS domain, and their results compared with the traditional techniques we used in a previous participation in this challenge (Araque et al., 2015).

The reminder of this paper is organized as follows. Sect. 2 introduces related work. Then Sect.3 describes the proposed polarity classification model and its implementation, which is evaluated in Sect. 4. Finally, conclusions are drawn in Sect. 6.

2 Related work

Many works in the last years involve the use of neural architectures to learn text classification problems and, more specifically, to perform Sentiment Analysis. A relevant example of this are Recursive Neural Tensor Networks (Socher et al., 2013). This architecture makes use of the structure of parse trees to effectively capture the negation phenomena and its scope. A similar work (Tai, Socher, and Manning, 2015) introduces the use of LSTM in tree structures, leveraging both the information contained in these trees and the representation capabilities of gated units. Although parse trees can result very useful in sentiment analysis, many works do not make use of them, as they introduce an additional computation overhead. In (Wang et al., 2015) a data-driven approach is described that learns from noisy annotated data also making use of LSTM units and a error signal processing to avoid the problem of vanishing gradient. Another useful technique is *attention* (Bahdanau, Cho, and Bengio, 2014), that enables weighting the importance of the different words in a given piece of text. Attention has been used in Sentiment Analysis successfully in a recurrent architecture, as presented in (Wang et al., 2016).

In the context of the TASS challenge, it has not been the first time that neural architectures have been proposed for solving the different tasks. In (Vilares et al., 2015), the authors propose a LSTM architecture that is compared to linear classifiers. Also, word embeddings have been leveraged in previous versions, as shown in (Martınez-Cámara et al., 2015). Nevertheless, neural networks have not been thoroughly studied in TASS, and many potentially interesting techniques remain unused.

3 Sentiment analysis Task

3.1 Model architecture

The approach followed for the Sentiment Analysis at Tweet level Task consists in a RNN composed of LSTM cells that parse the input into a fixed-size vector representation. The representation of the text is used to perform the sentiment classification. Two variations of this architecture are used: (i) a LSTM that iterates over the input word vectors or (ii) over a combination of the input word vectors and polarity values from a sentiment lexicon.

The general architecture of the model takes as inputs the words vectors and the lexicon values for each word from an input tweet. Then, the inputs are passed through a one-layer LSTM with a tunnable number of hidden units. The generated representation is then used to determine the polarity of the input text using a feedforward layer with softmax activation as output function. The output of this last layer encodes the probability that the input text belongs to each class. Fig. 2 shows this architecture, which is further described as follows:

1. The input vector is the word embedding of each word in a given tweet. It contains word-level information or sentiment word-level information. Each specific case will be described in more detail afterwards.
2. The RNN number of units is chosen during training for optimization purposes. In this work we use a one-layer LSTM to avoid overfitting of the network to the training data.
3. The weight matrix has as input dimension the RNN size, and the number of classes as output dimension. This means that, taking as inputs the last LSTM output, we obtain a vector whose length is the number of classes. This matrix is also optimized during the training process.

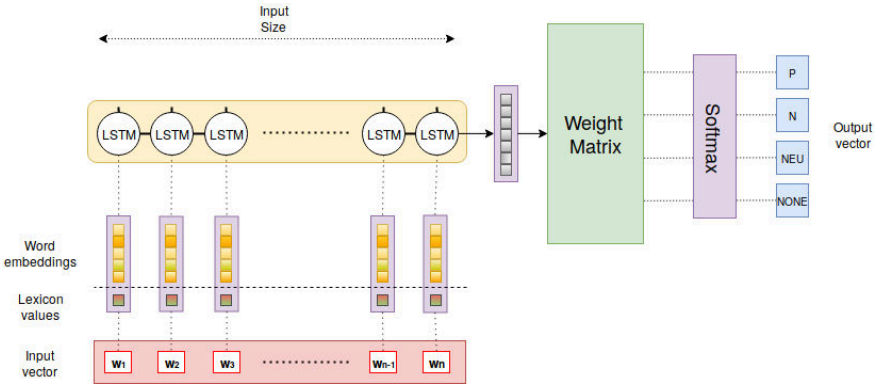


Figure 1: Recurrent Neural Network (RNN) architecture

4. The final probability vector is obtained by passing the result of the previous matrix multiplication through a softmax function, which converts the values of the components of this result vector into probabilities representation. Finally, the predicted label for the tweet is the component of the output vector with the highest probability.

Following, the two types of inputs used are described thoroughly.

3.2 Word-level RNN

For this input, the tweet text is tokenized into word tokens, which are then expressed in a one-hot representation. That is, each token is represented as a $\mathbb{R}^{|V|\times 1}$ vector with all 0s and and one 1 at the index of that token in the sorted token vocabulary. For example, the representations for the tokens *a*, *antes* and *zebra* would appear as:

$$w_{\text{a}} = \left[1\,0\,0\,\cdots\,0\right], w_{\text{antes}} = \left[0\,1\,0\,\cdots\,0\right]$$
$$w_{\text{zebra}} = \left[0\,0\,0\,\cdots\,1\right]$$

We limit the number of words to a certain vocabulary size in order to limit the computational cost of this preprocessing step. Before feeding this data to the network, each tweet is presented by the one-hot representation of all the tokens in the tweet.

3.3 Sentiment word-level RNN

Additionally, we include different sentiment information into the word representations by means of a sentiment lexicon. In this case, a similar approach as the word-level RNN is followed, but instead of using information about the different words contained on each tweet, information about the sentiment of each word is used. In this case, the preprocessing process is modified:

1. First, each tweet is split into tokens.
2. Secondly, a sentiment dictionary is used to map words with sentiment polarity values. In this way, each word is mapped into a positive, neutral or negative value.
3. Finally, the representation of a word consists in its word vector concatenated with its sentiment polarity label.

3.4 Regularization

Given the reduced number of training examples that is available for this task (Sec. 4) a number of regularization techniques has been used in the experiments. Regularization is used in machine learning to control the complexity of a learning model so it does not overfit to the training data and generalized better to the test data.

It is known that Recurrent Neural Network tend to heavily overfit to the training set (Zaremba, Sutskever, and Vinyals, 2014). For this, we employ two regularization techniques to prevent this:

1. L2 regularization (Ng, 2004). This technique is applied on the weights of the feedforward layer of the network. Being W_{MLP} the weights of this layer, this regularization adds to the cost function the following value:

$$\lambda \| W_{MLP}^{T} W_{MLP} \|$$

where λ is a parameter that represents the importance that is assigned to this regularization in the overall cost function.

2. Dropout (Srivastava et al., 2014; Gal and Ghahramani, 2016b). This strategy consists in randomly setting a fraction of units to 0 at each step of the training process to prevent overfitting. During test time, the outputs are averaged by this fraction. Dropout has been recently found to be theoretically similar to applying a bayesian prior to the network weigths (Gal and Ghahramani, 2016a).

3.5 TensorFlow implementation

TensorFlow is an interface for expressing machine learning algorithms, and an implementation for executing such algorithms (Abadi, Agarwal, and et al., 2015). For implementing the model previously described, first we had to define a computation graph composed by the RNN architecture, matrices and operations needed. Once the graph is defined, the training process consists in iteratively adjusting numerical values in order to reach the best results. This task was done following those ideas:

- The values to optimize are the internal parameters and matrices that form the network. Those are: the word embedding representations, the LSTM internal weights and the feedforward weight matrix used as last layer. At the beginning of training, those values are initialized in a random way using a normal distribution $\sim \mathcal{N}(\mu, \sigma)$, with $\mu = 0$, and are considered variables to be optimized at each training step by TensorFlow.
- Having those variables defined, the training process iteratively modifies them in order to reach the better results. In order to obtain a error signal that can be used to modify the learning parameters we use a cost function which has to be minimized. That minimization problem is solved by applying the gradient descent method via backpropagation (LeCun et al., 2012). In this work we employ the Adam algorithm (Kingma and Ba, 2014).
- In each iteration of the training process, which are known as epochs, data from the training set flows through all the computation graph yielding to a prediction result. The cost metric is computed by comparing the obtained result with the true training labels. When the backpropagation is finished, the variable values are updated and the following iteration proceeds.
- In order to enhance performance, we use early stopping on the accuracy on the development set. That is, for each epoch we monitor the performance of the network in the development set. If it has not improved for a number of epochs (in this work, 3 epochs) the training process is stopped and the model weights are freezed.
- The number of iterations can be chosen as well as other parameters such as the RNN size. For testing new examples, we use as input the test data, passing all the tweets through our model having as a result the vector of probabilities of the class each tweet belongs to, choosing the class with a higher probability value for each tweet.

4 Experimental setup

For the development of Task 1 a training and development dataset is made public, containing 1,514 labeled tweets which belong to the InterTASS corpus. Additionally, we use the TASS2015 edition training dataset that was extracted from the general corpus (García Cumbreras et al., 2016). We train the system with the InterTASS and the TASS General Corpus training datasets, and adjust the hyper-parameters with the InterTASS development set. For the lexicon, we used ElhPolar dictionary (Urizar and Roncal, 2013), as it has been previously used in TASS competitions.

There are three test datasets, one belonging to the InterTASS corpus and two belonging to the General Corpus of TASS: the full version, with all the 60,798 tweets; and the 1k version, that contains a subset of 1,000 tweets.

In order to enhance the classification performance several hyper-parameters have been explored, and the values that yield better performance are selected to be used in the testing phase. The vocabulary size is set to 20,000, with a batch size of 256 and the num-

Model	Corpus	Accuracy	Macro-F1
LSTM + MLP	InterTASS	53.70	37.1
LSTM + MLP	TASS (1k)	60.1	45.6
LSTM + MLP	TASS (Full)	63.1	50.9
LSTM + MLP + Lexicon	InterTASS	56.2	38.7
LSTM + MLP + Lexicon	TASS (1k)	63.6	46.8
LSTM + MLP + Lexicon	TASS (Full)	63.1	49.7

Table 1: Results in TASS 2017

ber of epochs being 20. With this value, the early stopping mechanism stopped the training before its completion. Regarding the size of the layers, the number of dimensions of the word embeddings is set to 16, as well as it is done with the number of units in the LSTM layer. The dimensionality of the feedforward layer is given by the output of the LSTM, which is 16, and the number of classes of the classification task (in this case, 4). Note that these values are smaller than in the usual neural architectures in order to further prevent overfitting. Also, we select the λ parameter to 0.05, and the dropout rate to 0.7.

5 Experimental Results

Table 1 shows the results of the two variations of the proposed model: LSTM+MLP stack with or without lexicon values. In light of this results it is possible to affirm that the used architecture shows promising performances in the task of sentiment analysis of tweets. Although, the achieved performances are below the best in this year challenge. This indicates that further work should be done in order to improve the results.

The experimental results confirm the idea that the introduction of a sentiment lexicon into the word presentations results, in general, beneficial for the final performance. We see this improvement in the InterTASS and 1k corpora. Nevertheless, when attending to the Full corpus, a performance decrease in the Marco-F1 is observed.

6 Conclusions and Future Work

In this paper we have described the participation of the GSI in the TASS 2017 challenge. Our proposal relies on a Recurrent Neural Network architecture for Sentiment Analysis with Long Short-Term Memory cells. This network can be fed with both word vectors and sentiment lexicon values. This approach is able to represent a arbitrarily long sequence of text due to the dynamic recurrent structure of the architecture. Also, several techniques have been used for avoiding overfitting. From the experiments, it is seen that adding a sentiment lexicon can enhance the classification performance.

However, the proposed model does not compare with the best results in the TASS competition. This can be due to a number of reasons, but the training process suggests that overfitting is a relevant issue. Although benefit comes from the use of regularization techniques, the network is not able to largely generalize. To address this, we think that future work in this direction should include the expansion of the training set.

Other possible improvement for future work is doing a better preprocessing of input texts at word level. In addition, Convolutional Neural Networks could be used for feature extraction in combination with the Recurrent Neural Network architecture. This could lead to the computation of most complex features, which could also yield better results.

Acknowledgement

The authors gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan X Pascal GPU used in this research. This research work is partially supported through the projects Semola (TEC2015-68284-R), Emo-Spaces (RTC-2016-5053-7), MOSI-AGIL (S2013/ICE-3019), Somedi (ITEA3 15011) and Trivalent (H2020 Action Grant No. 740934, SEC-06-FCT-2016).

Bibliografía

Abadi, M., A. Agarwal, and P. B. et al. 2015. TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.

Araque, O., I. Corcuera, C. Román, C. A.

Iglesias, and J. F. Sánchez-Rada. 2015. Aspect based sentiment analysis of spanish tweets. In *TASS@ SEPLN*, pages 29–34.

Araque, O., I. Corcuera-Platas, J. F. Sánchez-Rada, and C. A. Iglesias. 2017. Enhancing Deep Learning Sentiment Analysis with Ensemble Techniques in Social Applications. *Expert Systems with Applications*, June.

Bahdanau, D., K. Cho, and Y. Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Gal, Y. and Z. Ghahramani. 2016a. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059.

Gal, Y. and Z. Ghahramani. 2016b. A theoretically grounded application of dropout in recurrent neural networks. In *Advances in neural information processing systems*, pages 1019–1027.

García Cumbreras, M. Á., E. Martínez Cámara, J. Villena Román, and J. García Morera. 2016. Tass 2015–the evolution of the spanish opinion mining systems. *Procesamiento del Lenguaje Natural*, 56:33–40.

Kingma, D. and J. Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

LeCun, Y. A., L. Bottou, G. B. Orr, and K.-R. Müller. 2012. Efficient backprop. In *Neural networks: Tricks of the trade*. Springer, pages 9–48.

Martínez-Cámara, E., M. C. Díaz-Galiano, M. A. García-Cumbreras, M. García-Vega, and J. Villena-Román. 2017. Overview of tass 2017. In J. Villena Román, M. A. García Cumbreras, D. G. M. C. Martínez-Cámara, Eugenio, and M. García Vega, editors, *Proceedings of TASS 2017: Workshop on Semantic Analysis at SEPLN (TASS 2017)*, volume 1896 of *CEUR Workshop Proceedings*, Murcia, Spain, September. CEUR-WS.

Martınez-Cámara, E., Y. Gutiérrez-Vázquez, J. Fernández, A. Montejo-Ráez, and R. Munoz-Guillena. 2015. Ensemble classifier for twitter sentiment analysis. In R. Izquierdo, editor, *Proceedings of the Workshop on NLP Applications: completing the puzzle*, number 1386 in CEUR Workshop Proceedings, Aachen.

Ng, A. Y. 2004. Feature selection, l 1 vs. l 2 regularization, and rotational invariance. In *Proceedings of the twenty-first international conference on Machine learning*, page 78. ACM.

Socher, R., A. Perelygin, J. Y. Wu, J. Chuang, C. D. Manning, A. Y. Ng, C. Potts, et al. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the conference on empirical methods in natural language processing (EMNLP)*, volume 1631, page 1642.

Srivastava, N., G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958.

Tai, K. S., R. Socher, and C. D. Manning. 2015. Improved semantic representations from tree-structured long short-term memory networks. *arXiv preprint arXiv:1503.00075*.

Urizar, X. S. and I. S. V. Roncal. 2013. Elhuyar at tass 2013. In *Proceedings of the Workshop on Sentiment Analysis at SEPLN (TASS 2013)*, pages 143–150.

Vilares, D., Y. Doval, M. A. Alonso, and C. Gómez-Rodríguez. 2015. Lys at tass 2015: Deep learning experiments for sentiment analysis on spanish tweets. In *TASS@ SEPLN*, pages 47–52.

Wang, X., Y. Liu, C. Sun, B. Wang, and X. Wang. 2015. Predicting polarities of tweets by composing word embeddings with long short-term memory. In *ACL (1)*, pages 1343–1353.

Wang, Y., M. Huang, X. Zhu, and L. Zhao. 2016. Attention-based lstm for aspect-level sentiment classification. In *EMNLP*, pages 606–615.

Zaremba, W., I. Sutskever, and O. Vinyals. 2014. Recurrent neural network regularization. *arXiv preprint arXiv:1409.2329*.

A.2.6 Aspect based Sentiment Analysis of Spanish Tweets

Title	Aspect based Sentiment Analysis of Spanish Tweets
Authors	Araque, Oscar and Corcuera-Platas, Ignacio and Román-Gómez, Constantino and Iglesias, Carlos A. and Sánchez-Rada, J. Fernando
Proceedings	Proceedings of TASS 2015: Workshop on Sentiment Analysis at SEPLN co-located with 31st SEPLN Conference (SEPLN 2015)
ISBN	
Volume	1397
Year	2015
Keywords	aspect detection, Aspect detection, Machine Learning, natural language processing, Natural Language Processing, Sentiment analysis
Pages	29–34
Online	http://ceur-ws.org/Vol-1397/gsi.pdf
Abstract	This article presents the participation of the Intelligent Systems Group (GSI) at Universidad Politécnica de Madrid (UPM) in the Sentiment Analysis workshop focused in Spanish tweets, TASS2015. This year two challenges have been proposed, which we have addressed with the design and development of a modular system that is adaptable to different contexts. This system employs Natural Language Processing (NLP) and machine-learning technologies, relying also in previously developed technologies in our research group. In particular, we have used a wide number of features and polarity lexicons for sentiment detection. With regards to aspect detection, we have relied on a graph-based algorithm. Once the challenge has come to an end, the experimental results are promising.

TASS 2015, septiembre 2015, pp 29-34 recibido 17-07-15 revisado 24-07-15 aceptado 28-07-15

Aspect based Sentiment Analysis of Spanish Tweets

Análisis de Sentimientos de Tweets en Español basado en Aspectos

Oscar Araque, Ignacio Corcuera, Constantino Román, Carlos A. Iglesias y **J. Fernando Sánchez-Rada**
Grupo de Sistemas Inteligentes, Departamento de Ingeniería de Sistemas Telemáticos,
Universidad Politécnica de Madrid (UPM), España
Avenida Complutense, nº 30, 28040 Madrid, España
{oscar.aiborra, ignacio.cplatas, c.romang}@alumnos.upm.es
{cif, jfernando}@dit.upm.es

Resumen: En este artículo se presenta la participación del Grupo de Sistemas Inteligentes (GSI) de la Universidad Politécnica de Madrid (UPM) en el taller de Análisis de Sentimientos centrado en tweets en Español: el TASS2015. Este año se han propuesto dos tareas que hemos abordado con el diseño y desarrollo de un sistema modular adaptable a distintos contextos. Este sistema emplea tecnologías de Procesado de Lenguaje Natural (NLP) así como de aprendizaje automático, dependiento además de tecnologías desarrolladas previamente en nuestro grupo de investigación. En particular, hemos combinado un amplio número de rasgos y léxicos de polaridad para la detección de sentimento, junto con un algoritmo basado en grafos para la detección de contextos. Los resultados experimentales obtenidos tras la consecución del concurso resultan prometedores.
Palabras clave: Aprendizaje automático, Procesado de lenguaje natural, Análisis de sentimientos, Detección de aspectos

Abstract: This article presents the participation of the Intelligent Systems Group (GSI) at Universidad Politécnica de Madrid (UPM) in the Sentiment Analysis workshop focused in Spanish tweets, TASS2015. This year two challenges have been proposed, which we have addressed with the design and development of a modular system that is adaptable to different contexts. This system employs Natural Language Processing (NLP) and machine-learning technologies, relying also in previously developed technologies in our research group. In particular, we have used a wide number of features and polarity lexicons for sentiment detection. With regards to aspect detection, we have relied on a graph-based algorithm. Once the challenge has come to an end, the experimental results are promising.
Keywords: Machine learning, Natural Language Processing, Sentiment analysis, Aspect detection

1 Introduction

In this article we present our participation for the TASS2015 challenge (Villena-Román et al., 2015a). This work deals with two different tasks, that are described next.

The first task of this challenge, Task 1 (Villena-Román et al., 2015b), consists of determining the global polarity at a message level. Inside this task, there are two evaluations: one in which 6 polarity labels are considered (P+, P, NEU, N, N+, None), and another one with 4 polarity labels considered (P, N, NEU, NONE). P stands for *positive*, while N means *negative* and NEU is *neutral*. The "+" symbol is used for intensification of the polarity. It is considered that NONE means absence of sentiment polarity. This task provides a corpus (Villena-Román et al., 2015b), which contains a total of 68.000 tweets written in Spanish, describing a diversity of subjects.

The second and last task, Task 2 (Villena-Román et al., 2015b), is aimed to detect the sentiment polarity at an aspect level using three labels (P, N and NEU). Within this task, two corpora (Villena-Román et al., 2015b) are provided: SocialTV and STOMPOL corpus. We have restricted ourselves to the SocialTV corpus in this edition. This corpus contains 2.773 tweets captured during the celebration of the *2014 Final of Copa del*

Publicado en http://ceur-ws.org/Vol-1397/. CEUR-WS.org es una publicación en serie con ISSN reconocido ISSN 1613-0073

rey championship[1]. Along with the corpus a set of aspects which appear in the tweets is given. This list is essentially composed by football players, coaches, teams, referees, and other football-related concepts such as crowd, authorities, match and broadcast.

The complexity presented by the challenge has taken us to develop a modular system, in which each component can work separately. We have developed and experimented with each module independently, and later combine them depending on the Task (1 or 2) we want to solve.

The rest of the paper is organized as follows. First, Section 2 is a review of the research involving sentiment analysis in the Twitter domain. After this, Section 3 briefly describes the general architecture of the developed system. Following that, Section 4 describes the module developed in order to confront the Task 1 of this challenge. After this, Section 5 explains the other modules necessaries to address the Task 2. Finally, Section 6 concludes the paper and presents some conclusions regarding our participation in this challenge, as well as future works.

2 Related Work

Centering the attention in the scope of TASS, many researches have experimented, through the TASS corpora, with different approaches to evaluate the performance of these systems. Vilares et al. (2014) present a system relying in machine learning classification for the tasks of sentiment analysis, and a heuristics based approach for aspect-based sentiment analysis. Another example of classification through machine learning is the work of Hurtado and Pla (2014), in which they utilize Support Vector Machine (SVM) with remarkable results. It is common to incorporate linguistic knowledge to this systems, as proposed by Urizar and Roncal (2013), who also employ lexicons in its work. Balahur and Perea-Ortega (2013) deal with this problem using dictionaries and translated data from English to Spanish, as well as machine-learning techniques. An interesting procedure is performed by Vilares, Alonso, and Gómez-Rodríguez (2013): using semantic information added to psychological knowledge extracted from dictionaries, they combine these features to train a machine learning algorithm. Fernández et al. (2013) employ a ranking algorithm using bigrams and added to this a skipgrams scorer, which allow them to create sentiment lexicons that are able to retain the context of the terms. A different approach is by means of the Word2Vec model, used by Montejo-Ráez, García-Cumbreras, and Díaz-Galiano (2104), in which each word is considered in a 200-dimensional space, without using any lexical or syntactical analysis: this allows them to develop a fairly simple system with reasonable results.

3 System architecture

One of ours main goals is to design and develop an adaptable system which can function in a variety of situations. As we have already mentioned, this has taken us to a system composed of several modules that can work separately. Since the challenge proposes two different tasks (Villena-Román et al., 2015b), we will utilize each module when necessary.

Our system is divided into three modules:

- **Named Entity Recognizer (NER)** module. The NER module detects the entities within a text, and classifies them as one of the possibles entities. In the Section 5 a more detailed description of this module and the set of entities given is presented, as it is used in the Task 2.
- **Aspect and Context detection** module. This module is in charge of detecting the remaining aspects -aspects that are not entities and therefore can not be detected as such- and the contexts of all aspects. In the Section 5 this module is described in greater detail since it is only used for tackling the Task 2.
- **Sentiment Analysis** module. As the name suggests, the goal of this module is to classify the given texts using sentiment polarity labels. This module is based on combining NLP and machine learning techniques and is used in both Task 1 and 2. It is explained in more detail next.

3.1 Sentiment Analysis module

The sentiment analysis module relies in a SVM machine-learning model that is trained with data composed of features extracted from the TASS dataset: General corpus for

[1] www.en.wikipedia.org/wiki/2014_Copa_del_Rey_Final

the Task 1 and SocialTV corpus for Task 2 (Villena-Román et al., 2015b).

3.1.1 Feature Extraction

We have used different approaches to design the feature extraction. The reference document taken in the development of the features extraction was made by Mohammad, Kiritchenko, and Zhu (2013). With this in mind, the features extracted from each tweet to form a feature vector are:

- *N-grams*, combination of contiguous sequences of one, two and three tokens consisting on words, lemmas and stem words. As this information can be difficult to handle due to the huge volume of N-grams that can be formed, we set a minimum frequency of three occurrences to consider the N-gram.
- *All-caps*, the number of words with all characters in upper cases that appears in the tweets.
- *POS information*, the frequency of each part-of-speech tag.
- *Hashtags*, the number of hashtags terms.
- *Punctuation marks*, these marks are frequently used to increase the sentiment of a sentence, specially on the Twitter domain. The presence or absence of these marks (?!) are extracted as a new feature, as well as its relative position within the document.
- *Elongated words*, the number of words that has one character repeated more than two times.
- *Emoticons*, the system uses a Emoticons Sentiment Lexicon, which has been developed by Hogenboom et al. (2013).
- *Lexicon Resources*, for each token w, we used the sentiment score $score(w)$ to determine:
  1. Number of words that have a $score(w) \neq 0$.
  2. Polarity of each word that has a $score(w) \neq 0$.
  3. Total score of all the polarities of the words that have a $score(w) \neq 0$.

  The best way to increase the coverage range with respect to the detection of words with polarity is to combine several resources lexicon. The lexicons used are: Elhuyar Polar Lexicon (Urizar and Roncal, 2013), ISOL (Martínez-Cámara et al., 2013), Sentiment Spanish Lexicon (SSL) (Veronica Perez Rosas, 2012), SOCAL (Taboada et al., 2011) and ML-SentiCON (Cruz et al., 2014).
- *Intensifiers*, a intensifier dictionary (Cruz et al., 2014) has been used for calculating the polarity of a word, increasing or decreasing its value.
- *Negation*, explained in 3.1.2.
- *Global Polarity*, this score is the sum of the punctuations from the emoticon analysis and the lexicon resources.

3.1.2 Negation

An important feature that has been used to develop the classifier is the treatment of the negations. This approach takes into account the role of the negation words or phrases, as they can alter the polarity value of the words or phrases they precede.

The polarity of a word changes if it is included in a negated context. For detecting a negated context we have utilized a set of negated words, which has been manually composed by us. Besides, detecting the context requires deciding how many tokens are affected by the negation. For this, we have followed the proposal by Pang, Lee, and Vaithyanathan (2002).

Once the negated context is defined there are two features affected by this: N-grams and lexicon. The negation feature is added to these features, implying that its negated (e.g. positive becomes negative, +1 becomes -1). This approximation is based on the work by Saurí and Pustejovsky (2012).

4 Task 1: Sentiment analysis at global level

4.1 Experiment and results

In this competition it is allowed for submission up to three experiments for each corpus. With this in mind, three experiments have been developed in this task attending to the lexicons that adjust better to the corpus:

- *RUN-1*, there is one lexicon that is adapted well to the corpus, the ElhPolar lexicon. It has been decided to use only this dictionary in the first run.

- *RUN-2*, in this run the two lexicons that have the best results in the experiments have been combined, the ElhPolar and the ISOL.
- *RUN-3*, the last run is a mix of all the lexicon used on the experiments.

Experiment	Accuracy	F1-Score
6labels	61.8	50.0
6labels-1k	48.7	44.6
4labels	69.0	55.0
4labels-1k	65.8	53.1

Table 1: Results of RUN-1 in the Task 1

Experiment	Accuracy	F1-Score
6labels	61.0	49.5
6labels-1k	48.0	44.0
4labels	67.9	54.6
4labels-1k	64.6	53.1

Table 2: Results of RUN-2 in the Task 1

Experiment	Accuracy	F1-Score
6labels	60.8	49.3
6labels-1k	47.9	43.7
4labels	67.8	54.5
4labels-1k	64.6	48.7

Table 3: Results of RUN-3 in the Task 1

5 Task 2: Aspect-based sentiment analysis

This task is an extension of the Task 1 in which sentiment analysis is made at the aspect level. The goal in this task is to detect the different aspects that can be in a tweet and afterwards analyze the sentiment associated with each aspect.

For this, we used a pipeline that takes the provided corpus as input and produces the sentiment annotated corpus as output. This pipeline can be divided into three major modules that work in a sequential manner: first the NER, second the Aspect and Context detection, and third the Sentiment Analysis as described below.

5.1 NER

The goal of this module is to detect the words that represent a certain entity from the set of entities that can be identified as a *person* (players and coaches) or an *organization* (teams).

For this module we used the Stanford CRF NER (Finkel, Grenager, and Manning, 2005). It includes a Spanish model trained on news data. To adapt the model, we trained it instead with the training dataset (Villena-Román et al., 2015b) and a gazette. The model is trained with two labels: *Person* (PER) and *Organization* (ORG). The gazette entries were collected from the training dataset, resulting in a list of all the ways the entities (players, teams or coaches) were named. We verified the performance of the Stanford NER by means of cross-validation on the training data. With this, we obtained an average F1-Score of 91.05%.

As the goal of the NER module is to detect the words that represent a specific entity, we used a list of all the ways these entities were named. In this way, once the Stanford NER detect the general entity our improved NER module search in this list and decides the particular entity by matching the pattern of the entity words.

5.2 Aspect and Context detection

This module aims to detect the aspects that are not entities, and thus have not been detected by the NER module. To achieve this, we have composed a dictionary using the training dataset (Villena-Román et al., 2015b) which contains all the manners that all the aspects -including the entities formerly detected- are named. Using this dictionary, this module can detect words that are related to a specific aspect. Although the NER module already detects entities as players, coaches or teams, this module can detect them too: it treats these detected entities as more relevant than its own recognitions, combining in this way the capacity of aspect/entity detection of the NER module and this module.

As for the context detection, we have implemented a graph based algorithm (Mukherjee and Bhattacharyya, 2012) that allows us to extract sets of words related to an aspect from a sentence, even if this sentence has different aspects and mixed emotions. The context of an aspect is the set of words related

to that aspect. Besides, we have extended this algorithm in such a way that allow us to configure the scope of this context detection.

Combining this two approaches -aspect and context detection- this module is able to detect the word or words which identify an aspect, and extract the context of this aspect. This context allows us to isolate the sentiment meaning of the aspect, fact that will be very interesting for the sentiment analysis at an aspect level.

We have obtained an accuracy of 93.21% in this second step of the pipeline with the training dataset (Villena-Román et al., 2015b). As for the test dataset (Villena-Román et al., 2015b) we obtained an accuracy of 89.27%[2].

5.3 Sentiment analysis

The sentiment analysis module is the end of the processing pipeline. This module is in charge of classifying the detected aspects in polarity values through the contexts of each aspect. We have used the same model used in Task 1 to analyse every detected aspect in Task 2, given that the detected aspect contexts in Task 2 are similar to the texts analysed in Task 1.

Nevertheless, though using the same model, it is needed to train this model with the proper data. For this, we extracted the aspects and contexts from the train dataset, process the corresponding features (explained in Section 3), and then train the model with these. In this way, the trained machine is fed contexts of aspects that will classify in one of the three labels (as mentioned: positive, negative and neutral).

5.4 Results

By means of connecting these three modules together, we obtain a system that is able to recognize entities and aspects, detect the context in which they are enclosed, and classify them at an aspect level. The performance of this system is showed in the Table 4. The different RUNs represent separate adjustments of the same experiment, in which several parameters are controlled in order to obtain the better performance.

As can be seen in Table 4, the global performance obtained is fairly positive, as our system ranked first in F1-Score and second in Accuracy.

Experiment	Accuracy	F1-Score
RUN-1	**63.5**	**60.6**
RUN-2	62.1	58.4
RUN-3	55.7	55.8

Table 4: Results of each run in the Task 2

6 Conclusions and future work

In this paper we have described the participation of the GSI in the TASS 2015 challenge (Villena-Román et al., 2015a). Our proposal relies in both NLP and machine-learning techniques, applying them jointly to obtain a satisfactory result in the rankings of the challenge. We have designed and developed a modular system that relies in previous technologies developed in our group (Sánchez-Rada, Iglesias, and Gil, 2015). These characteristics make this system adaptable to different conditions and contexts, feature that results very useful in this competition given the diversity of tasks (Villena-Román et al., 2015b).

As future work, our aim is to improve aspect detection by including semantic similarity based on the available lexical resources in the Linguistic Linked Open Data Cloud. To this aim, we will integrate also vocabularies such as Marl (Westerski, Iglesias, and Tapia, 2011). In addition, we are working on improving the sentiment detection based on the social context of users within the MixedEmotions project.

Acknowledgement

This research has been partially funded and by the EC through the H2020 project MixedEmotions (Grant Agreement no: 141111) and by the Spanish Ministry of Industry, Tourism and Trade through the project Calista (TEC2012-32457). We would like to thank Maite Taboada as well as the rest of researchers for providing us their valuable lexical resources.

References

Balahur, A. and José M. Perea-Ortega. 2013. Experiments using varying sizes and machine translated data for sentiment analysis in twitter.

[2]We calculated this metric using the output granted by the TASS uploading page www.daedalus.es/TASS2015/private/evaluate.php.

Cruz, Fermín L, José A Troyano, Beatriz Pontes, and F Javier Ortega. 2014. Building layered, multilingual sentiment lexicons at synset and lemma levels. *Expert Systems with Applications*, 41(13):5984–5994.

Fernández, J., Y. Gutiérrez, J. M. Gómez, P. Martínez-Barco, A. Montoyo, and R. Muñoz. 2013. Sentiment analysis of Spanish tweets using a ranking algorithm and skipgrams.

Finkel, Jenny Rose, Trond Grenager, and Christopher Manning. 2005. Incorporating non-local information into information extraction systems by gibbs sampling. pages 363–370.

Hogenboom, A., D. Bal, F. Franciscar, M. Bal, F. De Jong, and U. Kaymak. 2013. Exploiting emoticons in polarity classification of text.

Hurtado, Ll. and F. Pla. 2014. ELiRF-UPV en TASS 2014: Análisis de sentimientos, detección de tópicos y análisis de sentimientos de aspectos en twitter.

Martínez-Cámara, E., M. Martın-Valdivia, MD Molina-González, and L. Ureña López. 2013. Bilingual experiments on an opinion comparable corpus. *WASSA 2013*, 87.

Mohammad, Saif M, Svetlana Kiritchenko, and Xiaodan Zhu. 2013. Nrc-canada: Building the state-of-the-art in sentiment analysis of tweets. In *Second Joint Conference on Lexical and Computational Semantics (* SEM)*, volume 2, pages 321–327.

Montejo-Ráez, A, M.A. García-Cumbreras, and M.C. Díaz-Galiano. 2104. Participación de SINAI Word2Vec en TASS 2014.

Mukherjee, Subhabrata and Pushpak Bhattacharyya. 2012. Feature specific sentiment analysis for product reviews. volume 7181 of *Lecture Notes in Computer Science*, pages 475–487. Springer.

Pang, Bo, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up?: sentiment classification using machine learning techniques. In *Proc. of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pages 79–86. Association for Computational Linguistics.

Sánchez-Rada, J. Fernando, Carlos A. Iglesias, and Ronald Gil. 2015. A Linked Data Model for Multimodal Sentiment and Emotion Analysis. 4th Workshop on Linked Data in Linguistics: Resources and Applications.

Saurí, Roser and James Pustejovsky. 2012. Are you sure that this happened? assessing the factuality degree of events in text. *Computational Linguistics*, 38(2):261–299.

Taboada, Maite, Julian Brooke, Milan Tofiloski, Kimberly Voll, and Manfred Stede. 2011. Lexicon-based methods for sentiment analysis. *Computational linguistics*, 37(2):267–307.

Urizar, Xabier Saralegi and Iñaki San Vicente Roncal. 2013. Elhuyar at TASS 2013.

Veronica Perez Rosas, Carmen Banea, Rada Mihalcea. 2012. Learning sentiment lexicons in spanish. In *Proc. of the international conference on Language Resources and Evaluation (LREC)*, Istanbul, Turkey.

Vilares, D., M. A. Alonso, and C. Gómez-Rodríguez. 2013. LyS at TASS 2013: Analysing Spanish tweets by means of dependency parsing, semantic-oriented lexicons and psychometric word-properties.

Vilares, David, Yerai Doval, Miguel A. Alonso, and Carlos Gómez-Rodríguez. 2014. LyS at TASS 2014: a prototype for extracting and analysing aspects from Spanish tweets.

Villena-Román, Julio, Janine García-Morera, Miguel A. García-Cumbreras, Eugenio Martínez-Cámara, M. Teresa Martín-Valdivia, and L. Alfonso Ureña-López, editors. 2015a. *Proc. of TASS 2015: Workshop on Sentiment Analysis at SEPLN*, number 1397 in CEUR Workshop Proc., Aachen.

Villena-Román, Julio, Janine García-Morera, Miguel A. García-Cumbreras, Eugenio Martínez-Cámara, M. Teresa Martín-Valdivia, and L. Alfonso Ureña-López. 2015b. Overview of TASS 2015.

Westerski, Adam, Carlos A. Iglesias, and Fernando Tapia. 2011. Linked Opinions: Describing Sentiments on the Structured Web of Data. In *Proc. of the 4th International Workshop Social Data on the Web.*

A.2.7 MAIA: An Event-based Modular Architecture for Intelligent Agents

Title	MAIA: An Event-based Modular Architecture for Intelligent Agents
Authors	Sánchez-Rada, J Fernando and Iglesias, Carlos A and Coronado, Miguel
Proceedings	Proceedings of 2014 IEEE/WIC/ACM International Conference on Intelligent Agent Technology
ISBN	
Year	2014
Keywords	agent architecture, event, maia, web hook
Pages	
Abstract	Online services are no longer isolated. The release of public APIs and technologies such as web hooks are allowing users and developers to access their information easily. Intelligent agents could use this information to provide a better user experience across services, connecting services with smart automatic behaviours or actions. However, agent platforms are not prepared to easily add external sources such as web services, which hinders the usage of agents in the so-called Evented or Live Web. As a solution, this paper introduces an event-based architecture for agent systems, in accordance with the new tendencies in web programming. In particular, it is focused on personal agents that interact with several web services. With this architecture, called MAIA, connecting to new web services does not involve any modification in the platform.

2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)

MAIA: An Event-based Modular Architecture for Intelligent Agents

J. Fernando Sánchez-Rada
Carlos A. Iglesias
and Miguel Coronado
Grupo de Sistemas Inteligentes
Universidad Politécnica de Madrid
Email: {jfernando, cif, miguelcb}@dit.upm.es

***Abstract*—Online services are no longer isolated. The release of public APIs and technologies such as web hooks are allowing users and developers to access their information easily. Intelligent agents could use this information to provide a better user experience across services, connecting services with smart automatic behaviours or actions. However, agent platforms are not prepared to easily add external sources such as web services, which hinders the usage of agents in the so-called Evented or Live Web. As a solution, this paper introduces an event-based architecture for agent systems, in accordance with the new tendencies in web programming. In particular, it is focused on personal agents that interact with several web services. With this architecture, called MAIA, connecting to new web services does not involve any modification in the platform.**

***Keywords*—Agent architecture, evented web, events, web hooks, jason**

I. Introduction

Agent architectures provide a valuable general guideline for designing and implementing agent applications [1] and have been a very active research topic in the agent community. In the 1990s, research interest was focused on the investigation of architectural issues raised by three influential threads of agent research (i.e. reactive agents, deliberative agents and interacting agents), as collects the excellent survey by Müller [2].

Software agent platforms are usually specialized in a particular agent architecture.For instance, most platforms for deliberative agents have adopted the Belief-Desire-Intention (BDI) model, as Jadex [3], Jack [4] or Jason [5], while the most popular agent platform for interacting agents, Jade [6], is based on FIPA [7]. Some of these platforms provide facilities to combine reasoning and interacting features, such as Jadex or Jason, which can be integrated with Jade.

The BDI architecture defined by Rao and Georgeff [8] is based on the original model proposed by Bratman for modelling human reasoning [9]. The BDI abstract architecture models human-like reasoning by capturing the mentalistic notions of belief, desire and intention, which are processed according to a generic interpreter. This interpreter assumes that events are atomic and recognized after they have occurred.

Traditionally, both messages and percepts have been managed in the same interpretation cycle, since both are considered forms of external events. As a result, most agent implementations mix reasoning processes with the communication logic and make them hard to reuse, debug and develop. Recently, several works such as ACRE [10] and Alfonso et al. [11] have proposed to delegate conversation management in a specific module external to the agent reasoning process. The interaction between these two modules is done through actions and perceptions. The reasoning module can reason about the outcomes of every conversation through a set of predefined perceptions, and then execute several actions to manage the status of those conversations (e.g. cancelling, forgetting or retrying a conversation).

Furthermore, agent platforms do not provide standardised mechanisms to integrate sensory information. This integration of sensors and actuators typically requires extending the basic agent architecture and a deep understanding of its implementation.

On the other hand, gathering information from external sources is a key aspect of any agent system. Lately, we are relying more and more on web services to store, share and generate new information.

Several works have proposed different mechanisms for integrating agents and web services, as surveyed in [12]. The existing solutions provide mappings between addressing and messaging schemes in web services and agent systems, and are implemented using a gateway that publishes web service descriptions into FIPA's directory facilitator and vice-versa. Nevertheless, there are application domains such as personal agents where the FIPA platform infrastructure is not needed but there is still the need to invoke services as a standard action.

A new trend in web service development is relying on event based interaction to allow services to interact. So much so that it is leading to a new generation of the web, called *real time web* or *evented web* [13]. This new wave of web services is characterised by its capability to process incoming events originated by a wide range of sources, such as social networks, service notifications or sensors.

Our proposal consists in overcoming the typical limitations in agent architectures while keeping them up to the current scenario. We do so by providing an event-based perspective to the internal composition of agent modules. This paper also explains how this architecture, called event-based Modular Architecture for Intelligent Agents (MAIA), can be used in applications that interact with a variable and increasing number of services, as well as its inner workings and implementation

978-1-4799-4143-8/14 $31.00 © 2014 IEEE
DOI 10.1109/WI-IAT.2014.154

IEEE computer society

challenges. To illustrate this, we also present an implementation of a personal cloud agent using MAIA.

This paper is structured as follows: Section III presents an overview of the architecture and describes its components in detail; Section IV covers the format and purpose of events; Section V shows how to use MAIA to build a personal agent; Section VI goes through related work; and in Section VII we present our conclusions and future work.

II. Event-based programming

Event-based programming [14], also called Event-Driven Architecture (EDA) is an architectural style in which one or more components in a software system execute in response to receiving one or more notifications. Event based programming differs from traditional web synchronous request-response interactions, since the main concepts are the events. Then, instead of speaking of clients and services, we refer to event producers and consumers. One of the main advantages of this architecture is that event producers and consumers can be decoupled, which improves its scalability and fault-tolerance capabilities. There are three main interaction styles in event programming [14]:

- *Push event distribution:* event producers emit an event and usually do not expect any specific action by event producers
- *Channel event distributions:* event producers send events to an event channel which acts as a broker, redirecting the event to event consumers subscribed to that particular event. This model is usually implemented using Message-oriented Middleware (MOM).
- *Pull event distribution:* event consumers follow the traditional request-response pattern to request an event from an event producers or from an event channel.

Event-based programming has been traditionally popular for programming user interfaces (e.g., Swing or JavaScript) as well as for integration architectures based on a Enterprise Service Bus. Given the requirements of the Live Web, event-based programming has given a step forward and is one of the cornerstones of highly interactive applications. We review in the following subsections *Node.js*, one of the most popular server-side programming environments, which is an example of the event oriented paradigm. Node.js applications are written in JavaScript and thus rely heavily on events.

III. MAIA architecture

An agent that does not interact with its environment (other software components, sensors, actuators, etc.) is of little practical use. For that reason, it is common practice to modify or extend agent platforms to include external sources. However, as previously explained, agent architectures tend to be monolithic. Connecting to external components is often a tedious and ad-hoc process. Regardless of the specific implementation, the resulting modifications are very heterogeneous and bound to the agent platform they were made for.

In an attempt to adapt generic BDI multi agent systems to seamlessly interact with different sources, we propose a new architecture, called MAIA.

The architecture has been designed to allow easy hot-plugging of new components that expand the capabilities of the system (e.g. new sensors). It consists of independent modules that perform different tasks (e.g. BDI reasoning, User Interface), which are connected using a common interface to a core platform that controls the flow of information between them.

Figure 1 shows an overview of the main modules in the architecture. At its core there is a bus for the modules that are closely related to a typical agent (BDI platform, sensors, actuators, etc.), another bus for the modules that connect to the Evented Web, and a central piece that connects both buses and provides additional services.

This section briefly presents these modules, focusing on the relationship between them. The following sections will describe each module separately in greater detail. The underlying communication mechanism is covered in Section IV.

First of all, the architecture includes a BDI Platform module which encapsulates all BDI functions and logic. This platform can be used to develop and run BDI agents that will communicate with the rest of the modules in the architecture.

An Adapter (labelled BDI Adapter) makes this communication possible by interfacing between the agent platform and the rest of the modules. Part of this adaptation is translating MAIA events to a format the platform understands, and vice versa. It will also make all the high level services from the rest of the modules available to the agents within the platform.

We use Jason as the reference BDI Platform in this paper, but any other platform such as Jade or Jadex would be suitable. The design of the BDI Adapter depends on the platform chosen.

The BDI Adapter is directly connected to the Agent Bus. The role of this bus is to connect the different high level modules of the agent, in contrast with the connectors to web services and other sources, which connect to the Evented Web Bus. This separation serves two main purposes: protecting the agent modules from an overload of events from the web, and providing additional capabilities to the modules connected to the Agent Bus (see Section III-B).

The Event Manager mediates between both buses, providing extra services to the Agent Bus as described in Section III-C. These services will have an important role in the development of BDI agents. Section III-A2 contains several plans and goals in Agent Speak that make use of these services.

A. Adapters

To be able to connect to any of the MAIA buses a module must communicate via events that are MAIA compliant (see Section IV) and use one of the protocols that its bus implements. Unfortunately, not all systems are natively evented. Even when they are, they do not always follow the MAIA events format or use the same protocol as the bus.

An Adapter is a piece of software that mediates between such systems and the rest of the modules. In the best case scenario, which is that of software that is already event oriented, the adaptation process is as simple as translating event formats on the fly and dealing with protocol differences.

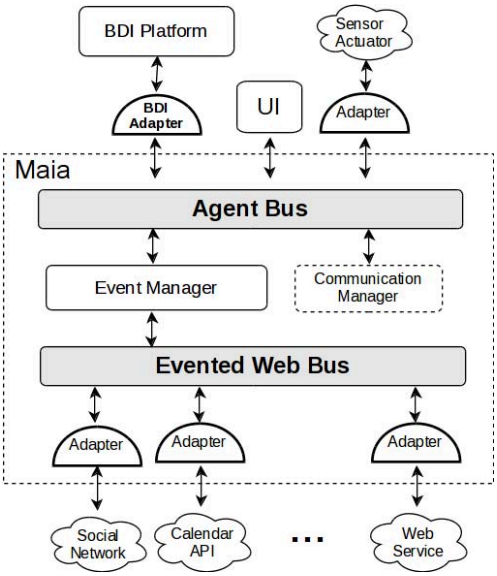


Fig. 1. High level representation of the MAIA architecture.

In the worst case scenario, deeper changes in the software itself might be needed.

We group the adapters in two categories according to the level of integration they provide: basic adapters and Agent Adapters. Basic adapters make the features of an external service or module available to the rest of the modules. Agent Adapters also make the advanced services provided by the Event Manager available to the module in question.

In essence, basic adapters simply add sources of information or interaction with external services, whereas an Agent Adapter connects to a module with more complex logic.

1) Basic Adapters: These adapters take care of: connecting with the Event Manager; translating event formats, back and forth; generating MAIA events and storing events for later consumption. Every adapter that connects to the Evented Web Bus is a basic adapter.

2) Agent Adapter: Agent Adapters are the interface between an agent system, typically an Agent Platform, and the Agent Bus. The role of these agent systems is to implement the logic of the final application, adding intelligence to the system and communicating to the different modules. The Event Manager provides several services to make it easier to perform certain common actions or simply delegate tasks that would otherwise be done by the agent. Thus, an Agent Adapter should integrate these services in the agent platform.

The design and features of the Agent Adapter highly depend on the target Agent Platform, its internals and the programming interface it offers. Hence, we will focus on the development of an adapter for Jason. Nevertheless, most of the

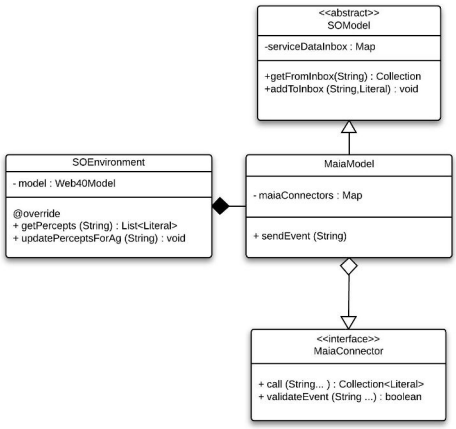


Fig. 2. Adding perceptions to agents in Jason

concepts herein are general and can be used in other Agent Platforms.

We identified three main challenges in the adaptation process. The first one consisted in communicating with the platform itself, and its individual agents. The second one was translating MAIA events to Jason beliefs. Lastly, there needs to be a way to use the extra services provided by the Event Manager from within any Jason agent. This section covers the first two, whereas Section V contains excerpts of Agent Speak code to deal with the most common MAIA services.

Every agent within Jason has its own knowledge database, which is populated by data from the different sources. To be able to actually modify the perceptions of the agents, a custom Jason Environment is needed, along with an ad-hoc model for this scenario. By modifying the basic Jason Environment we are able to control not only the sources through which new information is added, but the life cycle of such information.

More precisely, the custom model follows the data inbox concept, the same as regular mailboxes. All information received by the agent is volatile, and will be discarded after it is fetched. Should the agent find the information interesting or necessary for the future, it will save it as beliefs in its permanent knowledge database.

Using these data boxes it is rather easy to integrate our Java code and our agents in AgentSpeak. A special function allows any Java method to send information to any certain agent, and any Java function can be wrapped and made available to the agents in the platform. Figure 2 shows the custom elements created for the adapter.

Apart from the modifications explained above, events themselves need to be converted to beliefs internally. For this purpose, we created the libraries to translate a a subset of the JSON notation to beliefs and vice versa. Unfortunately,

the limited syntax of beliefs makes it impossible to perform a complete mapping.

Lastly, it is important to note that every agent should subscribe only to those events that are relevant to its functioning, and to avoid permanently storing them. Otherwise, we risk overloading the agents with too many facts, which hinders the reasoning process and might lead to undesired behaviours.

B. The Agent Bus and the Evented Web Bus

The role of the Evented Web Bus is to gather information from different web services and other non-web sensors, and to send information to those services when needed.

In addition to plain message passing, the bus has the following features: event filtering, event subscription and store and forward. Event Filtering provides the ability to select only the relevant events in each situation and for each module. By using Event Subscription modules can indicate their interest in certain kind of event which they wish to receive. Store and Forward means that modules can receive the events they subscribed to and that were sent while they were disconnected. It also means that events will be saved until they can be forwarded to a module. Without it, an overloaded module would not be able to consume all the events sent to it, which might then be discarded.

The Agent Bus connects the different modules that are directly related to the agents. The Agent Platform, the User-Interface and the Communication Manager are the most important examples of such modules.

In essence the Agent Bus works similarly to the Evented Web Bus. However, the modules connected are in charge of some of the highest level functions of the agent architecture. Thence they require some capabilities from the bus that were not necessary for the evented web. These capabilities are exposed to the agents in the form of services that highly ease the development of systems that take advantage of web services. Most of these services are focused on the development of personal agents that interact with social networks.

These services will be transparently provided to the modules in the Agent Bus by the Event Manager, covered in the following section.

It is important to note that the existence of these buses makes it possible to spread the modules that connect to it into several machines. Nonetheless, a simpler local configuration is possible.

C. Event Manager

The Event Manager is the core of the MAIA architecture. It is the bridge between the two buses. One of its roles is to exchange events between them, making Evented Web and sensory information available to agents and forwarding requests from agents to services. However, such information is usually verbose and frequent. Most of the times it is redundant or not critical. In contrast, the communication among agents or between agents and the user interface are usually more critical and sensitive to delays. As a consequence, the exchange between both buses obeys specific rules within the Event Manager. Such rules make the existence of two buses transparent to the clients of both while avoiding unnecessary forwarding between them.

Besides controlling the flow of events between the different modules, it complements the Agent Bus by providing higher level functions that are not present in it. The Event Manager provides several useful services for the development of personal agents.

Namely, these services are: Identity, Event Based Task Automation, Location, Semantic Information, Social Networks, Calendar and Transactions.

The Identity Service allows agents to define virtual identities. These identities can be linked to the rest of the services. For instance, an identity can be linked to several calendars and social networks. These identities are defined via FOAF [15]. Each identity has a unique ID that can be used to subscribe to the events from the sources linked to it. The Event Based Task Automation offers the option for agents to delegate actions to the Event Manager. These actions will be fired by a certain event, and their result will be another event.

The Social Network service homogenises the connection and interaction with different social networks. Social networks are an important part of the average user's everyday activity. By integrating them in a personal agent, we can gather relevant information about the user and improve the user's experience. Each social network profile can be linked to several identities. As we saw before, this means the events from different profiles will share a common namespace, making it easy to subscribe to all of them.

The Location service makes it possible to set locations to each identity. Events are sent every time there is a location change, or when a module queries the location of an identity.

The Calendar Service is a common interface to deal with calendars from different sources within Maia. It is especially meant as an abstraction for online calendar services.

The Information Service offers a simple unified interface for agents to query information from external information sources. As of this writing, the Information service supports SPARQL, being able to send queries to multiple endpoints (DBpedia, data.gov, etc.).

The Transaction service makes it easier for agents to handle operations with online services that follow a known pattern. For instance, the processes between booking a flight and arriving safe to the destination accommodation are quite similar regardless of the flight company, shuttle bus operator, etc. Given that, the Transaction service identifies different events as steps in such processes and acts accordingly to offer extra information to the agents.

IV. MAIA Events

The communication paradigm in MAIA purposely mimics that of the evented web [13]: all modules communicate through atomic messages called events. This paradigm follows the channel event distribution style.

The communication based on events is what confers loose coupling to the architecture. However, it also means that the structure and format of these events must cover a wide range

of scenarios. Furthermore, it is desirable to make events as compatible with the evented web as possible so that the interaction is seamless. This compatibility that must be achieved both in a conceptual level and in the format level.

The conceptual level deals with questions such as: what type of information does an event carry?, how do events relate to each other?, how are modules/services and events related? Most of these questions have already been answered in the previous sections, especially those related to the purpose and usage of events. The Live Web [13] introduces a very generic schema for events. However, a formal definition of the information within events is still missing.

The Evented Web Ontology (EWE) by Coronado et al. [16] formalises the idea of events on the web in the form of an ontology. The ontology itself was created after studying several task automation portals such as IFTTT. These portals either actively access services (web requests) or receive notification from them (web hooks). Either way, any new information from a services is modelled as an event. Users can choose what actions should be triggered when an event is detected (e.g. upload a picture to an image hosting site whenever there is new email with attachments). Interestingly, this scenario can be seen as a particular case of the evented web. The EWE vocabulary allows for such generalisation, which turns it into a consistent semantic model for representation of events. Hence, it provides the formal definition necessary for conceptual compatibility.

Describing EWE in depth is out of the scope of this paper. However, we will describe the concepts that are necessary to understand its use in this work and how it had to be expanded. Among other things, EWE defines Channels, Events and Actions. A Channel is a source of information, such as an e-mail inbox. Channels generate Events whenever there is new information, like whenever there is new mail. Each Channel also has a list of available Actions, like deleting an email.

In MAIA every new module is a Channel. For adapters, this Channel actually represents the source they are adapting. Additionally, an event can be either informative or a request, in the sense that it may inform of an action performed or of an intention to trigger an action in a remote entity. In other words, a module emits an event when there is new information to share, or when it expects another module to perform an action.

On the other hand, there are several possible formats to serialise semantic information. To simplify the task of developing new adapters to the evented web, MAIA events use the JSON-LD [17] format in its compact form. This approach has multiple advantages: it is a lightweight human-readable format; there are libraries to efficiently process JSON in almost every programming language and JSON-LD libraries have been made for most of them; semantic and non-semantic information can coexist in the same JSON object; and plain JSON information from the evented web might be converted to semantic JSON-LD by adding an appropriate context.

In summary, MAIA events are messages in JSON-LD format that are modelled using the EWE ontology. Events have the following fields:

- **id (@id)** Unique identifier of the sent event for the specified entity (source).
- **timestamp (dcterms:created)** Time of the original emission. This makes time reasoning possible and prevents the side effects of asynchronous communications.
- **source (ewe:source)** Unique identifier of the sending entity.
- **name (dcterms:title)** Which describes the event, and is the only required field. Ideally, it will not only consist of a basic string, but of a complete namespace. This allows for a complex processing of the events and an advanced filtering for triggers. We will get into details later in this section.
- **parameters (ewe:hasParameter)** For any kind of non-trivial event, we will need more information about the entities involved in the event, or the parameters if it is a request. This field is a list of ewe:Parameter objects, with description, title and value.
- **expiration** Used to announce other entities that after this time the success or error callbacks will not be called, to prevent them from replying to or acknowledging the event.

```
{
 "@context": {
  "ewe": "http://www.gsi.dit.upm.es/ontologies
      /ewe/ns",
  "dcterms": "http://purl.org/dc/terms",
  "id": "@id",
  "@type": "ewe:Event",
  "source": "ewe:source",
  "timestamp": {
   "@id": "dcterms:created",
  },
  "name": "dcterms:title",
  "parameters": {
    "@id": "ewe:hasParameter",
    "@container": "@list",
    "@type": "ewe:Parameter"
  },
  "description": "dcterms:description",
  "title": "dcterms:title",
  "value": "dcterms:value",
 },
 "id": "http://demos.gsi.dit.upm.es/maia#
     MailChannel_"
 "source": "http://demos.gsi.dit.upm.es/maia#
     MailChannel_ev_1389937684001
 "timestamp": 1389937684,
 "name": "MailChannel::email::new",
 "parameters": [
   {
     "title": "subject",
     "value": "Testing Maia",
     "description": "Subject of the email"
   }
 ],
 "expiration": 1389937694
}
```

Listing 1. Example of an event in MAIA that represents a MailChannel.

In addition to these fields, the complete JSON-LD object also includes a context to provide the semantic metadata of

each field. A complete example of an event can be seen in Listing 1

All events are named following a simple convention, the names are strings separated by double colons, the first string being the name of the module that sent it, for example: *MailChannel::email::new*. Modules use these names to subscribe to events from other sources. For instance, in our previous example a module would need to subscribe to *MailChannel::email::new* to receive the new email events from MailChannel.

What is interesting about MAIA events is that they may contain wildcards $\star$ or double wildcards $\star\star$. Using wildcards, a module can subscribe to a wide range of events. If the name of the event and the name used in the subscription match, the event will be forwarded. A single wildcard replaces/matches any string between double colons (e.g. *a::b::c* and *a::*::c* match). A double wildcard replaces/matches zero or more slots (e.g. *a::b::c* and ***::c* match, and also *a::b::c::***). Wildcards can be used either in the subscription name or in the event name, the comparison is applied symmetrically.

In order to efficiently process these matches and allow a high throughput of events, MAIA buses use an optimised subscription handling algorithm based on subscription trees.

Although one of the aims of the events system is to achieve asynchronous, it is worth noting that namespaces and the expiration information allow some sort of remote method invocation. To reply to an event, another event with the name `<source>::success::<id>` or `<source>::error::<id>` can be sent before *Expiration*, where *¡source¿* is the identifier of the sender and *¡id¿* is the ID of the original event. These events are currently not being forwarded to the rest of the modules.

As a last comment about the format of events, we have developed adapters for SPARQL and Spotlight endpoints. A W3C recommendation [18] can be used to include the results from SPARQL queries in events.

V. Case study: building a personal agent

To clarify some of the concepts explained before and put them in context, we will go through an example implementation of a personal agent in the travelling domain.

The aim of this personal agent is to assist users with their trips. This assistance includes: following the process between booking a ticket and arriving to the destination, alerting of any irregularity such as delays, cancellations or forecast alerts; informing users about flight deals during their free days; checking the activity on social networks about topics related to the trip; and handling emails and social activity on behalf of the users when they are away.

For all this to work, the agent will need to connect to: a flight search service; a forecast service; an email server; and a social network. The interaction between the user and the personal agent will be via text messages. The natural language processing of the messages from the user to an external REST Natural Language Understanding (NLU) Service. Each of the external services has an associated adapter module, as seen in Figure 3.

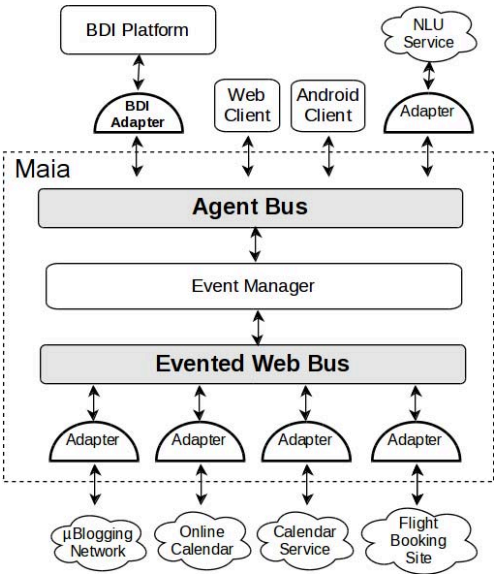


Fig. 3. Architecture of the Prototype.

The logic of the personal agent is provided by a single Jason agent, the travel agent. This section shows excerpts of code and simplified examples that demonstrate how to interact with the Event Manager to make use of its services. More specifically, it contains AgentSpeak plans to: get the semantic information of the country of the flight destination, which can later be used to fetch more information; alert the user via email when the user has confirmed a flight and the forecast information in the city of origin or destination is negative; subscribe to activity in all the subscribed microblogging sites about the country or city of destination two weeks before the flight, and alert the user about suspicious activity.

It is possible to simplify the syntax to emit frequent events, as seen in Listing 2

```
1 email(To,From,Subject,Body) :- parameters((
      name("to"),value(To)),(name("from"),value
      (From)),(name("subject"),value(Subject))
      ,(name("body"),value("body"))).
2 sendEmail(To,From,Subject,Body) :- event(["
      action","email","send"],email(To,From,
      Subject,Body)).
```

Listing 2. Definitions for email handling.

Listing 3 contains a plan to process forecast information during or close to a day of a scheduled flight. To receive such forecast information, the agent must have already subscribed to forecast alerts or any event from the information service.

Listing 4 exemplifies how an agent can query a SPARQL endpoint to get more information. In particular, it fetches the capitals of the capitals in Europe if a new flight is booked but

the country of the destination city is not known. The query is limited to European cities to use a simple query to a public endpoint (DBpedia).

```
+info("forecast",data(Date,City,Temperature,
    Forecast,Chances))
  : flight(Dept,City,From,To)[id(Identity)] |
      flight(City,Arriv,From,To)[id(Identity
    )]  ((Temperature < 20 | Forecast ==
    "rain" )  Chances > 0.3 )
  <-!suggest_deals(Identity,Dept,Arriv,From,
      To);
    sendEmail(email_address(Identity),null,"
       Bad weather for your trip",(Date,
       Temperature,Meteo,Chances)).
```

Listing 3. Process forecast information when a flight has been scheduled.

```
+flight(_,City,_,_)
  : ~country(City,_)
  <-query_sparql("
    SELECT distinct ?country ?capital (SAMPLE
        (?caplat) AS ?caplat) (SAMPLE(?
        caplong) AS ?caplong)
    WHERE {
     ?country rdf:type dbpedia-owl:Country .
     ?country dcterms:subject <http://dbpedia
         .org/resource/Category:
         Countries_in_Europe> .
     ?country  dbpedia-owl:capital ?capital .
     OPTIONAL {
       ?capital geo:lat ?caplat ;
         geo:long ?caplong .
     }
    }
    ORDER BY ?country
    ",country(1,2),location(2,3,4,_)).
```

Listing 4. Demonstrates how to use a SPARQL query to gather new information.

Lastly, Listing 5 presents a simple example which makes use of the social service. More specifically, the agent subscribes to microblogging events up to fifteen days before a flight is scheduled to depart. The social service will then send alerts about activity when there are enough microblogging posts related to the destination city or country. It is easy to imagine that this feature is helpful to detect noteworthy happenings in the destination country (riots, strikes, concerts, etc.)

```
+flight(_,City,Dept(YY,MM,DD,_,_,_),_)[id(
    UserID)]:
  : ((DD > 15 .date(YY,MM,DD-15)) |
      (.date(YY,MM-1,DD+15)))
       country(City,Country)
  <-social(event(["id",UserID,"social","
      ublogging","**","stream","peak"), [
      Country,City], ["alert","activity","
      ublogging","away"]).

+event(["alert", "activity", "ublogging", _],
     data(Volume, Posts))[id(Identity)]
  : Volume > 10
  <-ui_alert(Identity, "Relevant info from
      the social networks about your
      destination:", Posts).
```

Listing 5. Subscribe to notifications about peaks in activity about the destination of a trip and warn the user via the UI upon alert.

The interaction with the user can be done via a Web client (a Google Chrome extension that connects to the Agent Bus), or an Android application. Both clients also send the location of the user, so they are both UIs and sensors.

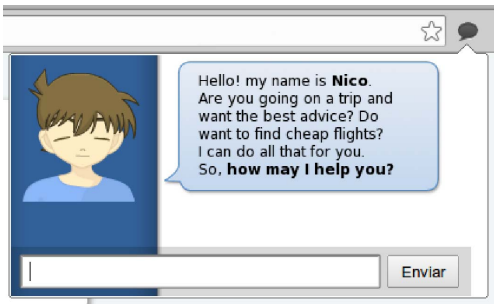


Fig. 4. User Interface as a Chrome Extension

VI. Related work

Several authors have addressed the definition of an event based agent architecture. Munteanu [19] proposes an event-based middleware for Cloud Governance based on multiagent system. Their work is focused on identifying the agent roles for cloud governance and does not deal with engineering an event-based agent system. Thus, our solution can complement their proposal since it provides a suitable architecture for event-based processing.

In the first prototypes of this system, different multi agent system platforms were evaluated. The most promising of them being SPADE (Smart Python multi-Agent Development Environment) [20], as it includes the XMPP protocol in its core and many of its communication features and its advantages: publish-subscribe mechanism to allow push updates, form-data to manage work-flow between user, libraries for many programming languages and platforms, etc..

So far, we referred to communication between modules in the general sense. The elements mentioned make it possible to exchange information between different parties. However, agent communication is a more sophisticated process that has been treated broadly in other texts [10], which describe complex agent communication solutions. Although MAIA focuses on a different problem, it was designed so that these solutions are compatible with and can be implemented on top of it. To make this possible, two possible additions might be needed: one in the agent level, adding the communication logic and protocols; and another one on the platform level, which allows agents to announce or subscribe their services, share protocol definitions or that acts as a mediator in disputes. The first addition would be made on top or within the MAIA adapter, if it is not already contemplated in the agent platform. The second one is labelled as Communication Manager module in the MAIA architecture. This paper will not cover this specific module, but it is important to note that the architecture was created with it in mind.

VII. Conclusions and future work

The architecture presented in this paper proves that it is possible to achieve modern systems that combine the potential of intelligent agent systems and the interconnection and ever-growing applications of the modern web.

The resulting application goes beyond the state of the art, putting together already existing solutions from different fields. It thus shows that we can make good use of the existing technologies to implement innovative ideas.

It is important to note that the most important shift is in the way we understand agents and agent communication. Adapting existing systems and frameworks to MAIA also requires work, especially in the case of Multi Agent Systems. However, such adaptation only needs to be done once, and it allows its connection to a wide range of modules.

There are several aspects in which MAIA can be extended or improved. It also opens the discussion about the integration of the evented programming paradigm and the design of BDI agents.

One of the main aspects to improve from a pragmatic point of view is the security of the information being exchanged and the scope in which it is visible. Currently MAIA allows username/password authentication and mechanisms to control event subscription on a per-module basis.

Another field for future research is to further expand the definition of events to include other concepts such as propagation of events. This might lead to delegation and collective planning, but it also poses challenges related to agent communication.

References

[1] J. P. Müller and M. Pischel, "The agent architecture interrap: Concept and application," German Research Center for Artificial Intelligence (DFKI), Tech. Rep., 1993.

[2] J. P. Müller, "Architectures and applications of intelligent agents: A survey," *The Knowledge Engineering Review*, vol. 13, no. 4, pp. 353–380, 1999.

[3] A. Pokahr and L. Braubach, "From a research to an industry-strength agent platform: Jadex v2," *Business Services: Konzepte, Technologien, Anwendungen. 9. Internationale Tagung Wirtschaftsinformatik*, pp. 769–780, 2009.

[4] P. Wallis, R. Ronnquist, D. Jarvis, and A. Lucas, "The automated wingman - using jack intelligent agents for unmanned autonomous vehicles," in *Aerospace Conference Proceedings, 2002. IEEE*, vol. 5, pp. 5–2615–5–2622 vol.5.

[5] R. H. Bordini and J. F. Hübner, "Bdi agent programming in agentspeak using jason," in *Proceedings of 6th International Workshop on Computational Logic in Multi-Agent Systems. Volume 3900 of lncs*. Springer, 2005, pp. 143–164.

[6] F. L. Bellifemine, G. Caire, and D. Greenwood, *Developing Multi-Agent Systems with JADE (Wiley Series in Agent Technology)*. John Wiley & Sons, 2007.

[7] "Foundations for intelligent physical agents (FIPA)," 2001, available from http://www.fipa.org.

[8] A. S. Rao, M. P. Georgeff *et al.*, "Bdi agents: From theory to practice," in *Proceedings of the first international conference on multi-agent systems (ICMAS-95)*. San Francisco, 1995, pp. 312–319.

[9] I. Bratman, "Plans, and practical reason," *Cambridge, Mass.: Harvard UP*, 1987.

[10] D. Lillis, "Internalising Interaction Protocols as First-Class Programming Elements in Multi Agent Systems," Ph.D. dissertation, University College Dublin, 2012.

[11] B. Alfonso, E. Vivancos, V. Botti, and A. García-Fornes, "Integrating jason in a multi-agent platform with support for interaction protocols." in *Proceedings of the compilation of the co-located workshops on DSM'11, TMC'11, AGERE!'11, AOOPES'11, NEAT'11 and VMIL'11*, ser. SPLASH '11 Workshops. New York, NY, USA: ACM, 2011, pp. 221–226. [Online]. Available: http://doi.acm.org/10.1145/2095050.2095084

[12] D. Greenwood, M. Lyell, A. Mallya, and H. Suguri, "The ieee fipa approach to integrating software agents and web services," in *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems*, ser. AAMAS '07. New York, NY, USA: ACM, 2007, pp. 276:1–276:7. [Online]. Available: http://doi.acm.org/10.1145/1329125.1329458

[13] P. Windley, *The Live Web: Building Event-Based Connections in the Cloud*. Course Technology, 2011. [Online]. Available: http://books.google.es/books?id=_AxfXwAACAAJ

[14] O. Etzion and P. Niblett, *Event Processing in Action*. Manning Publications Co., 2010.

[15] D. Brickley and L. Miller. (2014, Jan.) Foaf vocabulary specification. [Online]. Available: http://xmlns.com/foaf/spec/

[16] M. Coronado and C. A. Iglesias. (2013) Ewe ontology: Modeling rules for automating the evented web. GSI. [Online]. Available: http://www.gsi.dit.upm.es/ontologies/ewe/

[17] M. S. et al. (2014, Jan.) Json-ld 1.0. [Online]. Available: http://json-ld.org/spec/latest/json-ld/

[18] A. Seaborne. (2011, Jan.) Sparql results in json. [Online]. Available: http://www.w3.org/TR/sparql11-results-json/

[19] V. I. Munteanu, T.-F. Fortis, and V. Negru, "An event driven multi-agent architecture for enabling cloud governance," in *Proceedings of the 2012 IEEE/ACM Fifth International Conference on Utility and Cloud Computing*, ser. UCC '12. Washington, DC, USA: IEEE Computer Society, 2012, pp. 309–314. [Online]. Available: http://dx.doi.org/10.1109/UCC.2012.50

[20] M. E. Gregori, J. P. Cámara, and G. A. Bada, "A jabber-based multi-agent system platform," in *Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems*. ACM, 2006, pp. 1282–1284.

A.2.8 EuroLoveMap: Confronting feelings from News

Title	EuroLoveMap: Confronting feelings from News
Authors	Atserias, Jordi and Erp, Marieke van and Maks, Isa and Rigau, Germán and Sánchez-Rada, J. Fernando
Proceedings	Proceedings of Come Hack with OpeNER!" workshop at the 9th Language Resources and Evaluation Conference (LREC'14)
ISBN	978-2-9517408-8-4
Year	2014
Keywords	
Pages	5
Abstract	

EuroLoveMap: Confronting feelings from News

Jordi Atserias[1], Marieke van Erp[2], Isa Maks[2], German Rigau[3], J. Fernando Sánchez-Rada[4]

[1]Yahoo Labs Barcelona, [2]VU University Amsterdam, [3]The University of Basque Country,
[4]Universidad Politécnica de Madrid
jordi@yahoo-inc.com, {marieke.van.erp,e.maks}@vu.nl, german.rigau@ehu.es, jfernando@gsi.dit.upm.es

Abstract

Opinion mining is a natural language analysis task aimed at obtaining the overall sentiment regarding a particular topic. This paper presents a prototype that presents the overall sentiment of a topic based on the geographical distribution of the sources on this topic. The prototype was developed in a single day during the hackathon organised by the OpeNER project in Amsterdam last year. The OpeNER infrastructure was used to process a large set of news articles in four different languages. Using these tools, an overall sentiment analysis was obtained for a set of topics mentioned in the news articles and presented on an interactive worldmap.

Keywords: Opinion Mining, Visualisation, Hackathon

1. Introduction

Different topics are often presented in news from different perspectives. These perspectives may differ between countries and cultures, and are brought to the fore through different communication outlets. We aim to detect these opinions from news articles from different languages to compare the polarity profiles in different countries with respect to a particular topic. Within NLP research, there is a fair body of work on opinion and sentiment analysis (Pang and Lee, 2008; Liu, 2012). Several toolkits have been developed for the detection of polarity in text, but full multilingual opinion detection which includes the holder of the opinion and the target is still lagging. The OpeNER project plans to deliver an opinion detection tool that is trained on an annotated corpus of political news and aims at a sentence-based detection of opinion expressions with their holders and targets. For this demo, however, we use the rule-based opinion tagger that was available in June 2013.

This paper presents a prototype developed in a single day during the June 2013 hackathon organised by the OpeNER project (Agerri et al., 2013)[1] in Amsterdam.[2] OpeNER aims to detect and disambiguate entity mentions and perform sentiment analysis and opinion detection on the texts for six different languages (Maks et al., 2014). Team NAPOLEON used the OpeNER infrastructure[3] and web services[4] to obtain sentiment analyses for news articles in four different languages which were then aggregated into topics per country and presented visually on a map.

In the remainder of this contribution, we detail our system in Section 2., and present some examples in Section 3. We conclude with future work in Section 4.

2. Mining feelings from news using OpeNER

During the hackathon, we processed around 22,000 news articles in four different languages obtained from the RSS service of the European Media Monitor.[5] The content as well as some metadata of the newspaper articles was obtained before the hackathon. For this prototype, we decided to focus on English, Spanish, Italian and Dutch. For instance, the topic *gay marriage* was manually translated to the four languages and news articles relevant to this topic were collected and processed. An overall sentiment score was also obtained per language for each topic. Finally, the aggregated score for every topic-language pair was used for colouring a world map.

During the hackathon, we developed some software modules to process each news article through the OpeNER web services. In the remainder of this section, we detail the different steps in the workflow.

The OpeNER architecture consists of several Natural Language Processing (NLP) components. Each component is configured to take the information it requires to perform a specific analysis. KAF (Bosma et al., 2009) is used as linguistic representation. Each of the NLP processing pipelines is deployed as a Cloud Computing service using Amazon Elastic Computing Cloud[6] (Amazon EC2). Figure 1 presents an overview of the OpeNER components deployed as web services.

At the end of the different natural language processing pipelines, the extracted information is combined to obtain polarity clusters for the different topics selected.

Language Identifier: This component is responsible for detecting the language of an input news article and delivers it to the correct pipeline.

Tokenizer: This component is responsible for tokenising the text on two levels; 1) sentence level and 2) word level. This component is crucial for the rest of NLP components and is the first component in each language processing pipeline.

Part of Speech Tagger: This component is responsible for assigning to each token its morphological label, it also includes the lemmatisation of words. Combining the lemma and morphological label, later modules will consult a sentiment lexicon in order to assign polarity values to the words appearing in the news being processed.

Named Entity Recognition: This module provides Named Entity Recognition (NER) for the six languages covered by

[1] http://www.opener-project.org
[2] http://opener-fp7project.rhcloud.com/2013/07/18/opener-hackathon-in-amsterdam/
[3] http://opener-project.github.io/
[4] http://opener.olery.com/
[5] http://emm.newsbrief.eu/overview.html
[6] http://aws.amazon.com/ec2

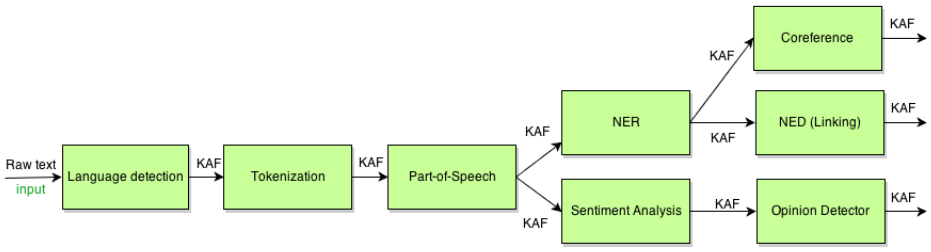


Figure 1: Overview of the components of the OpeNER pipeline

OpeNER and tries to recognize four types of named entities: persons, locations, organisations and names of miscellaneous entities that do not belong to the previous three groups.

Named Entity Linking: Once the named entities are recognised they can be identified or disambiguated with respect to an existing catalogue. This is required because the "surface form" of a Named Entity can actually refer to several different things in the world. Wikipedia has become the de facto standard as named entity catalogue. In OpeNER the NED component is based on the DBpedia Spotlight[7] which uses the DBpedia[8] as the resource for disambiguation entities.

Sentiment Analysis: The Opinion tagger we used is a rule and dictionary based tagger. It detects positive and negative polarity words (such as 'nice' and 'awful'), as well as intensifiers or weakeners (such as 'very' and 'hardly') and polarity shifters (such as 'not'). In addition, the module includes some simple rules that detect the holders and targets of the opinions related to the positive and negative polarity words.

Finally, the processed news in KAF format are stored and indexed using Solr[9] to easily query and retrieve the news articles about a selected topic. A web service was deployed to obtain json results grouping the scores detected by topic and language. The json results are then presented to the user in a world map.

3. Topics on EuroLoveMap

In order to test the prototype we manually selected a small number of topics in English, which were manually translated to Spanish, Italian and Dutch.[10] Table 1 presents the English topics and the corresponding translations in Spanish, Italian and Dutch used in the prototype[11].

Figure 2 presents a screenshot of the EuroLoveMap demo showing the extracted opinions on "gay marriage".

[7] http://github.com/dbpedia-spotlight/dbpedia-spotlight/wiki

[8] http://dbpedia.org

[9] https://lucene.apache.org/solr/

[10] To scope the prototype, we decided to focus only on four out of the six project languages.

[11] The resulting demo can be found at http://eurolovemap.herokuapp.com/.

4. Future Work

As this is only a very first prototype built in a few hours during the previous OpeNER hackathon, there are several different avenues of research as well as engineering issues that spring from it.

To make the prototype more informative and useful for users interested in analysing trending opitions, possible extension to the prototype could be a trend line or the option to look at different snapshots of the EuroLoveMap. This could provide insights into how the opinions on the different topics evolve in different countries.

For selecting the news sources, we currently use language identification, but one preferably uses the publisher information as there may be news sources aimed at expats in languages different from the country's main language. This would not only be more precise, but also give us access to a host of background information about these sources that can be mined in order to obtain more fine-grained information. Different publishers can for example be classified as more left or right leaning. Having this information enables us to present a more fine-grained analysis of the different perspectives within a country. Information about the publisher or authors of the articles could be further mined to create authority and trust profiles using PROV-O(Moreau et al., 2012). Being able to bring up the actual text of the mined articles would make the EuroLoveMap a useful tool to for example communication scientists or anthropologists.

For this prototype, we manually selected the topics and translated them. Ideally, a system picks up on trending topics, for example by plugging into the European Media Monitor or Twitter trends and detecting which topics would be interesting to analyse. To translate these topics automatically one could imagine using DBpedia or a similar resource.

As processing the articles via the NLP pipelines is a time-consuming process, we are currently working with a static dump of processed articles. Research in for example the NewsReader[12] architecture is underway to optimise NLP pipelines further, but until then the most viable option for updating the demo would be with daily batches that are processed overnight.

[12] http://www.newsreader-project.eu

English	Spanish	Italian	Dutch
Berlusconi	Berlusconi	Berlusconi	Berlusconi
Boston	Boston	Boston	Boston
North Korea	Corea del Norte	Corea del Nord	Noord-Korea
Obama	Obama	Obama	Obama
Putin	Putin	Putin	Poetin
CIA	CIA	CIA	CIA
Snowden	Snowden	Snowden	Snowden
Spain	España	Spagna	Spanje
United States, US	Estados Unidos, E.E.U.U.	Stati Uniti	Verenigde Staten van Amerika, VS
Netherlands	Holanda	Olanda	Nederland, Holland
Italy	Italia	Italia	Italië
Germany	Alemania	Germania	Duitsland
Gay marriage, homosexual marriage	matrimonio homosexual, matrimonio gay	matrimonio gay	homohuwelijk

Table 1: Topics and translations

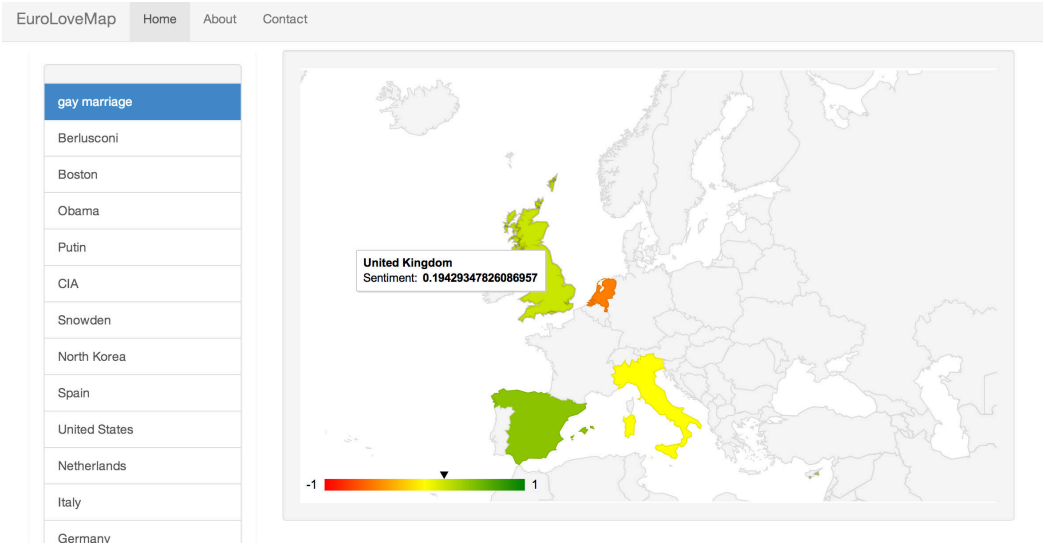


Figure 2: Screenshot of the EuroLoveMap demo showing the extracted opinions on "gay marriage"

Acknowledgements

This research is supported by the European Union's 7th Framework Programme via the OpeNER project (ICT 296541) and the NewsReader Project (ICT-316404).

5. References

Agerri, R., Cuadros, M., Gaines, S., and Rigau, G. (2013). Opener: open polarity enhanced named entity recognition. Sociedad Española para el Procesamiento del Lenguaje Natural (SEPLN'2013).

Bosma, Wauter, Vossen, Piek, Soroa, Aitor, Rigau, German, Tesconi, Maurizio, Marchetti, Andrea, Monachini, Monica, and Aliprandi, Carlo. (2009). Kaf: a generic semantic annotation format. In *Proceedings of the GL2009 Workshop on Semantic Annotation*.

Liu, Bing. (2012). Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies*, 5(1):1–167.

Maks, Isa, Izquierdo, Ruben, Frontini, Francesca, Azpeitia, Andoni, Agerri, Rodrigo, and Vossen, Piek. (2014). Generating polarity lexicons with wordnet propagation in 5 languages. In *In Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC 2014)*, Reykjavik, Iceland, May.

Moreau, Luc, Missier, Paolo, Belhajjame, Khalid, B'Far, Reza, Cheney, James, Coppens, Sam, Cresswell, Stephen, Gil, Yolanda, Groth, Paul, Klyne, Graham, Lebo, Timothy, McCusker, Jim, Miles, Simon, Myers, James, Sahoo, Satya, and Tilmes, Curt. (2012). PROV-DM: The PROV Data Model. Technical report.

Pang, Bo and Lee, Lilian. (2008). Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1-2).