# Neural Domain Adaptation of Sentiment Lexicons

Oscar Araque*, Marco Guerini†, Carlo Strapparava† and Carlos A. Iglesias*
*Grupo de Sistemas Inteligentes, Departamento de Ingeniería Telemática
Universidad Politécnica de Madrid, E.T.S.I. de Telecomunicación, Avda. Complutense, 30, Madrid, Spain
o.araque@upm.es, carlosangel.iglesias@upm.es
†FBK-Irst, Via Sommarive 18, Trento, Italy
guerini@fbk.eu, strappa@fbk.eu

***Abstract*—Sentiment lexicons are widely used in computational linguistics, as they represent a resource that directly contains subjective sentimental knowledge. Usually these sentiment lexica are generic and developed without any specific semantic domain in mind. Nonetheless, the domain context can be highly relevant for sentiment analysis, as it is known that word polarities can be influenced by domain-specific traits. This paper studies the problem of automatically generating domain-adapted sentiment lexicons that can be used in posterior sentiment analysis tasks. We propose a neural network approach that modifies a sentiment lexicon using distantly annotated text of a certain domain. Additionally, we present a completely data-driven domain characterization metric that measures the centrality of a set of documents. Experimental work shows that this metric offers a measure of the generated lexicons' quality. Also, it is shown that the generated lexicons yield higher performance on domain-oriented sentiment analysis than a generic lexicon and other known baselines. Finally, it is also discussed that these extracted lexicons can be used for sentiment analysis even for approaches with no learning capabilities.**

## 1. Introduction

Sentiment Analysis (SA) has become a popular research area in recent years due to its applicability to a wide range of problems, the growth and accessibility of on-line opinions and its commercial interest [1]. For instance, automatic analysis of product reviews are of special interests to a great variety of companies, as it allows them to have a measure on their customers' opinions towards their stocks. Other areas where Sentiment Analysis has been applied are movie reviews [2], congressional floor debates [3], news and blogs [4], among others. This variety of data sources and target domains make it difficult to construct a generic sentiment classifier. In this context, it is specially relevant the role that the domain plays in the Sentiment Analysis process. The sentiment associated with, for example, a product review can be greatly affected by the domain to which it belongs. In this sense, sentiment associated to different words can vary in its intensity or even shift its polarity value altogether (e.g. 'big' when describing a monitor screen usually associates with a positive orientation, while can have a negative sense when referring to clothes).

In relation to this, many sentiment classification tasks make use of opinion lexicons [5]. An opinion lexicon is a collection of words that have associated sentiment polarity values that can be continuous range of numbers or simpler polarity labels. There are many prior polarity lexica (where generic sentiment scores are associated out of context), such as [6], [7] and [8]. However, using a generic lexicon for a domain-specific sentiment task does not yield the best results because, as discussed, words can vary in their sentiment values. On top of this, it is known that words sentiment is domain dependent [9]. For these reasons, the task of automatically generating posterior polarity (domain-adapted) lexicons in a flexible manner is an important research problem.

In this work, we propose a neural network based method for automatically extracting domain-adapted sentiment lexicons. This model makes use of distantly annotated data belonging to a certain domain in order to compile a human-readable sentiment lexicon that can be used for posterior sentiment analysis on that domain. Also, this work introduces a novel metric that characterizes certain features of a set of domain-oriented documents. We show that this metric offers a way of predicting the effectiveness of the proposed model.

The basic idea for the proposed metric is to estimate the centrality of a set of documents, so one can have a measure of the performance of the proposed model before any training. We define the centrality as the measure of how specific is a given set of documents to a variety of topics. The experiments show that the introduced metric gives a clear sense of the quality of the extracted lexicon by means of our model.

As for the proposed lexicon adaptation model, its main characteristic is that it uses the back-propagation algorithm [10] to change the lexicon values. This model makes use of the loss signal of a sentiment classification task to better adjust those values, finally generating the desired domain-adapted sentiment lexicon, which we call SEDLex (SEntiment adapted to the Domain Lexicon).

The rest of the paper is organized as follows. Section 2 shows previous work on lexicon domain adaptation. In Sec-

978-1-5386-0680-3/17/$31.00 

tion 3, the proposed domain characterization metric is described, as well its relation with the proposed model, which is explained in Section 4. After this, the paper continues with an overview on the experimental setup in Section 5. Section 6 follows, illustrating the results obtained in the experiments, as well as the validation of the proposed techniques. Finally, Section 7 draws the conclusions obtained, and outlines the future work.

## 2. Related Work

Many works have been oriented to domain adaptation of sentiment lexicons. As depicted in [5], three main techniques have been explored for generating opinion lexicons: (i) manual approach, (ii) dictionary-based approach and (iii) corpus-based approach. The manual approach is very time-consuming, and thus it is normally combined with the other two, that rely on automatic methods. Many dictionary-based methods consist in bootstrapping from a small set of opinionated words, and using them as seed for the search of similar terms in a known dictionary such as WordNet [11] or SentiWordNet [7]. In order to improve the generated lexicons researchers have used machine learning techniques [12], as well as some external sources of knowledge, such as the definition of WordNet words (glosses) [13]. Nevertheless, as explained in [5], the dictionary-based approaches are not able to obtain domain specific orientations. In this regard, corpus-based techniques can alleviate the problem. This last type of lexicon generation technique relies on co-occurrence patterns detected on a large corpus, as well as a seed set of opinion words to locate opinionated words in the corpus [5].

There is a great variety of statistical information extraction methods for generation of domain-oriented lexicons. Nevertheless, some works share some common methodology, as is the case of *optimization-based* approaches. For instance, the work shown in [14] proposes an information bottleneck problem that blends cross and within-domain knowledge. This problem is then solved with an iterative optimization method that, upon convergence, obtains the desired lexicon. A different approach is explored in [15], where a linear programming technique is aimed to directly reflect the characteristics of a certain domain. This method exploits the relation among words and opinion expressions to compute the most likely polarity value in that domain. Another explored technique is presented in [16], in which several opinion signals are extracted from an unlabeled opinionated text collection. These signals are combined into an optimization problem, and then transformed to a linear programming formulation by means of a mathematical transformation. In this way, the method can learn the sentiment of words and aspects.

Another common technique used when generating domain-specific lexicons is *label propagation*. In this line of work, the technique explained in [17] introduces a method called double propagation. This method consists in several extraction rules that are designed based on dependency trees, and exploit the relations between modifier sentiment words and their associated topics, as well as the sentiment and topics words. A similar research [18] proposes an automatic construction strategy based on constrained label propagation. The strategy firstly extracts candidate sentiment terms, then it uses domain and morphological constraints to spread to the entire set of candidates, improving the lexicon. Label propagation can be also applied onto different structures, such as a lexical graph. This approach is taken in [19], where a set of high-quality word embeddings are transformed into a lexical graph by connecting each word with its nearest neighbors in the semantic space. Then, the sentiment labels are propagated from a set of seed words through the graph using a random walk algorithm.

Lastly, there are other methods that directly extract statistical information from corpora, and transform this knowledge to a domain-adapted sentiment lexicon. The method described in [20] uses frequency information to assign different weights to positive and negative words in the domain of movie reviews. Another example of such approaches is [21], which uses the *TF-IDF* score for each sentiment polarity to adapt an existing general lexicon to a specific domain. Also, the works described in [22] and [23] describe the *Compositional Semantics* methods that, parting from a frequency measure of the words in the corpus and a set of emotion labels (this could be used in sentiment as well), builds a lexicon from a word-emotion matrix.

## 3. Domain characterization

Understanding what a domain is in the context of sentiment analysis can be beneficial for domain adaptation applications. In the context of this work, properly defining a domain can substantiate the lexicon adaptation process. Considering an approach that tries to adapt a certain lexicon to a specific domain, we would like to know some characteristics of the data before any training, so we can have a measure of how well the adaptation is going to perform.

In this work, we tackle this problem by proposing a data-driven metric that aims to characterize a set of documents $D$. The proposed metric aims to estimate the semantic centrality of $D$. That is, how specific to a certain set of topics $T$ the dataset is. As an example, we can consider a dataset that describes some products of the electronics domain. If the majority of the text in that dataset refers to certain recurrent topics (phones, batteries, screens, computers, ...), then it is possibly a centered domain. On the contrary, a dataset that includes a large set of topics has a lower centrality.

Following this, we propose a metric $m$ that measures the centrality of $D$ using a semantic structure. In this work, a word embeddings model is used. It is known that word embeddings retain a semantic structure that can be exploited in a variety of applications [24]. The proposed method aims to leverage the information embedded in this representation for domain characterization. The idea is to measure the distances between a set of selected words $W$ from $D$. These measures retain information of how disperse $W$ is on the vector space, and thus the extent of centrality of $D$. Figure 1

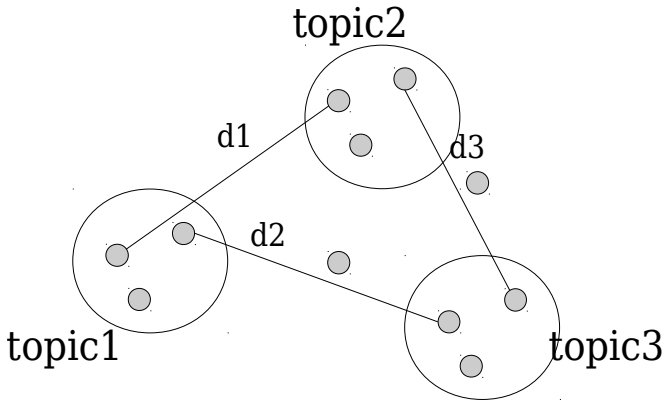


Figure 1. Distances measures between words from the set $W$.

shows a graphical representation of the measures in relation to a set of topics. We define the metric as:

$$m = \frac{1}{|W|} \sum_{i,j}^{|W|} d(v_i, v_j)$$

being $|W|$ the number of elements in $W$. The distance between word vectors $v_i$ and $v_j$ can be expressed as follows, $||\cdot||_2$ being the euclidean norm.

$$d(v_i, v_j) = \frac{v_i v_j}{\|v_i\|_2 \|v_j\|_2}$$

For extracting the set $W$ from the dataset, a number of most common words is first selected, and filtered with the stopwords set from [25]. Then, the embedding vectors for those words are extracted, forming the $V$ set. The metric is computed for all the possible pairs from $V$. If $|V|$ is a moderate amount (e.g. 300-600), the cost of computing all the possible combinations of two word vectors is not too high[1].

## 4. Lexicon Adaptation Model

### 4.1. Lexicon Generation

Automatically generating a domain-centered sentiment lexicon is an open research problem. This work considers the situation where we have access to distantly labeled textual data. We refer to distantly labeled data as the one that is not manually nor automatically annotated, but has some meta-data associated that can be used as sentiment label. In this work, we consider product reviews that have a meta-data value belonging to a 5-score system. From this score value, we extract a sentiment signal with which we train the proposed model. Section 5 further explains this process.

The proposed approach consists in a feedforward neural network (also called multilayer perceptron). The network is trained from representations of a set of documents that have associated distant labels. These inputs are the lexicon values of the words contained in the document representation. That is, if we consider a set of $k$ words, the values from a certain sentiment lexicon are extracted and used as representation for the network. Regarding the initialization of the sentiment lexicon values, the network uses an existing generic lexicon from which starts modifying.

Using the distant labels, the network is trained with a cross-entropy loss function on the task of classifying the sentiment of a certain document using the lexicon values. The key idea is that, at training time, the back-propagation algorithm can be used to let the sentiment error signal flow to the network weights, as well as to the lexicon values. In this way, by training the network to differentiate the sentiment of the documents, it is also modifying the lexicon values, which are a continue range of numeric values. Considering that the training documents are domain-oriented we can expect that, after training, the modified lexicon contains useful sentiment information with respect to the domain.

### 4.2. Lexicon Transformation

When training is finished, the modified values are extracted, forming a new domain-adapted sentiment lexicon. Nevertheless, the resulting lexicon does not normally have the expected distribution of a human-readable lexicon. That is, we would expect a sentiment lexicon to be bounded in a range such as [-1, 1]. Also, positive orientations should be represented by positive numbers, and the contrary for negative sentiment orientations. The modification of the lexicon values relies on the back-propagation algorithm, and this method has no inherent constraints on the distribution of a certain set of parameters. In order to transform the modified lexicon to one that exhibits a "human-readable" distribution, we define a transformation operation. More formally, let $\mathbf{a}$ be the vector that contains the lexicon values so that the value $\mathbf{a}_i$ represents the lexicon value for the word $i$ of the vocabulary. We define the transformation operation as $g((\mathbf{w} * \mathbf{a}) + b)$, $*$ being the element-wise product and $g$ the element-wise function $g(x) = (x - a)\frac{d-c}{b-a} + c$ that transforms the values from one range $[a, b]$ to another $[c, d]$. Figure 2 illustrates how the transformation process operates.

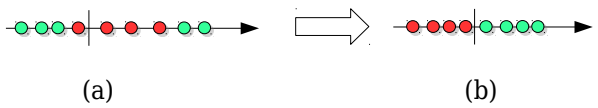


Figure 2. Graphical representation of polarity distribution of the modified lexicon (a) before the transformation process, and (b) after the transformation.

In order to obtain the parameters $\mathbf{w}$ and $b$ we train a logistic regressor on a subset of the training data. The parameters of the defined operation and those of the input to the sigmoid function of the logistic regression are equal [26], so by training the learner we directly obtain the desired parameters. The extracted lexicon, that is, the one

1. Note that the number of combinations is given by the expression $\binom{|V|}{2} = \frac{|V|!}{2!(|V|-2)!}$

TABLE 1. DATASETS STATISTICS

| Name | #Reviews | #Windows |
|---|---|---|
| Toys and Games | 2,252,771 | 8,060,106 |
| Health and Personal Care | 2,982,326 | 11,718,861 |
| Movies and TV | 4,607,047 | 27,042,309 |
| Electronics | 7,824,482 | 38,032,004 |
| Books | 22,507,155 | 138,422,556 |

resulting from the training of the neural network and the transformation process, can be directly used for sentiment analysis on the trained domain.

# 5. Experimental Setup

In order to evaluate the proposed approach, several experiments have been designed. We evaluate the effectiveness of the model through the sentiment analysis performance of the generated lexicons for each domain. For this reason, various known domain-oriented datasets have been used for training and evaluating the model.

## 5.1. Datasets

The datasets used for training are the Amazon Product Data [27], [28], which contain millions of product reviews extracted form Amazon. From these datasets, we have used the reviews text and their corresponding ratings, which are ranging from 1 to 5. For extracting the polarity labels the ratings are mapped to positive and negative values, being 1 and 2 a negative value; while 4 and 5 are transformed as a positive polarity. Documents with ratings of 3 are discarded. The selected domains and their basic statistics are shown in Table 1. The test dataset is the Multi Domain Sentiment Dataset v2 [29], which contains a set of product reviews already annotated. For the evaluation, we selected the same domain for the training and test sets, for each domain, as the test set is also divided domain-wise.

## 5.2. Training Instances Generation

The training examples that are fed to the proposed model are not directly text documents, but rather fixed windows of tokens. For obtaining these windows, each document is processed independently. Firstly, the set of words that constitute each document are intersected with the defined vocabulary (e.g. the vocabulary of the original lexicon). In this way, the words that contain sentiment information are selected. This intersection of words is then downsampled by randomly selecting a fraction of the length of the intersection set. Then, for each target word $w_t$ in the downsampled set we extract a symmetric fixed window containing the words around $w_t$. If the window boundaries lie outside of the document, zero-padding is applied. As for the sentiment label, the window is assigned a label that is equal to that of the original document. Finally, the set of windows obtained are the training instances that correspond to the given document. In this way, we aggregate all the training instances from all the documents, obtaining more instances than reviews in the original dataset. The number of reviews and training instances are described in Table 1.

## 5.3. Evaluation Workflow

For the model evaluation, we use both the training and test sets. First, the proposed model is trained with the training windows extracted from the training data for a specific domain. The training dataset is divided into batches, and the network is allowed to update its weights when each batch has been propagated forward (Sec. 4.1). Following, the lexicon is transformed as explained in Sec. 4.2.

Periodically, the model's modified lexicon is extracted and its individual performance is evaluated on the test set of that same domain via 3-fold cross-validation. To this end, a logistic regression model is trained on sparse representations of the documents. These features are created by generating high-dimensionality sparse vectors, where each index element corresponds to a word in the vocabulary. The value of a element is 0 if the word does not appear in the document, while the value is equal to the sentiment lexicon value for that word if it does appear. When the training is finished, the extracted lexicon is the one that has achieved better performance. In this way, the whole model is not evaluated, only the sentiment lexicon that has been generated by the model.

Apart from the accuracy, we also use the correlation between the values of the lexicon and the labels of the test set. This metric is used in order to have a fair comparison of the generated lexicons performance (i.e. without any syntactic or compositional reasoning that can boost the performance), as done in [8]. That is, for the words in a document of the test set, the lexicon values are extracted for those words, and the average is applied. These values are then compared to the labels of each document through the Pearson correlation.

## 5.4. Baselines

We use several baselines that are evaluated in the same way that the lexicons generated by our proposed model for comparison purposes. All methods have the same vocabulary for each domain. The most simple is the *random* baseline, in which the lexicon word values are randomly generated number from a normal distribution. The random baseline does not yield an accuracy of 50% due to that vocabulary information is introduced in the document representations. That is, if a word appears in a document the baseline has a random value in the corresponding index of the representation vector. If a word does not appear, a zero is used. In this way, the representation vectors are sparse and not completely random, which yields slightly better results than a full random approach.

Another baseline used in this work is the compositional semantics (CS) method [22], [23]. We evaluate three variants of this method: raw frequency counts (CS raw), normalized counts (CS norm), and TF-IDF frequency values (CS tfidf). Finally, the last baseline is the sentiment lexicon Sentiwords,

TABLE 2. ACCURACY ON THE TEST SET FOR BASELINES AND PROPOSED MODEL.

| | Domain | | | | |
|---|---|---|---|---|---|
| **Model** | Movies | Electronics | Health | Books | Games |
| Random | 65.20 | 66.40 | 66.95 | 61.60 | 67.91 |
| CS raw | 79.75 | 80.20 | 81.45 | 79.35 | 80.65 |
| CS norm | 80.15 | 80.50 | 81.45 | 79.70 | 81.55 |
| CS tfidf | 70.75 | 80.80 | 81.85 | **80.75** | 80.90 |
| Sentiwords | 80.05 | 78.75 | 78.95 | 78.10 | 80.30 |
| SEDLex-mod | **81.60** | 80.65 | 81.71 | 79.40 | 84.05 |
| SEDLex-trans | **81.60** | **83.09** | **83.55** | 80.20 | **84.15** |

proposed in [8]. This represents a generic sentiment lexicon in our experiments, as it is not adapted to any domain.

## 6. Results

For the experiments, several hyper-parameters have been tested, and we report the parameters that yield the best results on a randomly selected dev set, drawn from the train set. For the neural model, a two-layer feedforward network is used with 52 and 25 units in each layer. Good results have been achieved with learning rates ranging from 0.0005 to 0.0001 with the Adam algorithm [30] as optimizer. During training the window is set to be symmetric with size 5, and the downsampling ratio used is 0.1. Interestingly, it has been observed that the transformation operation (Sec. 4.2) can be simplified by setting $b = 0$. This simplification further improves the performance results.

The first metric on which we evaluate the proposed model is the Accuracy on the test set. Table 2 shows the performance of the baselines and the proposed model. Our model is evaluated separately for the lexicon without transformation (SEDLex-mod, Sec. 4.1) and with transformation (SEDLex-trans, Sec. 4.2).

It can be seen that the best performing lexicon in almost all domains is the one generated by our full model. This result confirms the idea that the training process of a neural architecture on domain data can generate domain adapted lexicons. Also, as it is shown in the last two rows of Table 2, performing the proposed linear transformation further improves the effectiveness of the generated lexicon. We have confirmed that, after the transformation, positive word polarities are represented by positive values, and vice versa.

For the second evaluation metric, the correlation, Table 3 presents the associated results. It is interesting to see that after the transformation process the correlation is highly improved in comparison to that of the generated lexicon before the transformation. This is an indicative that the formulated transformation can correctly distribute the generated lexicon values as desired, positioning the positive words and the negative words in their corresponding numeric polarity.

Following, we evaluate how well the proposed metric $m$ indicates the centrality of the datasets. For this end, we compute the Pearson correlation between the difference of accuracy score between the Sentiwords baseline and the transformed lexicon for each domain, as Sentiwords is used for initializing the lexicon at training time. In this sense, the measure is understood as the gain in information for sentiment analysis. Table 4 shows the metric values computed from the domain datasets compared to that difference in accuracy. The correlation between these two columns is of **-0.8207**. This result indicates that there is a relation between the value of the proposed metric and the performance improvement of the generated sentiment lexicons.

TABLE 3. CORRELATION BETWEEN SUM OF LEXICON VALUES AND TEST LABELS.

| | Domain | | | | |
|---|---|---|---|---|---|
| **Model** | Movies | Electronics | Health | Books | Games |
| Random | 0.067 | -0.060 | -0.046 | -0.016 | 0.031 |
| CS raw | 0.062 | 0.106 | 0.055 | 0.048 | 0.154 |
| CS norm | 0.069 | 0.127 | 0.074 | 0.051 | 0.172 |
| CS tfidf | 0.076 | 0.168 | 0.132 | 0.041 | 0.287 |
| Sentiwords | 0.153 | 0.147 | 0.199 | 0.083 | 0.152 |
| SEDLex-mod | -0.085 | 0.167 | 0.119 | 0.029 | 0.207 |
| SEDLex-trans | **0.688** | **0.707** | **0.707** | **0.671** | **0.733** |

TABLE 4. DIFFERENCE IN ACCURACY METRIC BETWEEN PROPOSED MODEL AND BASELINE COMPARED TO $m$ VALUES,

| **Domain** | $\mathbf{Acc_{SEDLex\text{-}trans}}$ **-** $\mathbf{Acc_{baseline}}$ | $m$ **value** |
|---|---|---|
| Movies | 1.55 | 0.1380 |
| Electronics | 4.34 | 0.1186 |
| Health | 4.60 | 0.1251 |
| Books | 2.10 | 0.1302 |
| Games | 3.85 | 0.1303 |

## 7. Conclusions and Future Work

This paper proposes a sentiment lexicon domain adaptation framework based on a neural network architecture that, through the process of training for a straightforward sentiment classification task on domain data, modifies a lexicon in order to better contain domain information. The generation process is enhanced by a posterior transformation that provides the generated lexicon with a human-understandable distribution and range, following the majority of known sentiment lexicons. Besides, a domain characterization metric is presented that measures the centrality of a set of documents that relate to a given domain. It is shown that this metric strongly correlates with the improvement of the lexicons generated by the proposed model on a benchmark dataset in comparison to a known generic sentiment lexicon. It is argued that this metric can be used for measuring the effectiveness of the framework and the lexicons it generates provided a set of domain documents with a distant sentiment annotation.

Indeed, the experiments have shown that this model effectively adapts a sentiment lexicon to a certain domain, outperforming the performance of a linear classifier trained with only the lexicon as features with respect to a generic sentiment lexicon. Furthermore, the generated lexicons show

a strong correlation between their values and the sentiment labels of the test set. This allows the use of the generated lexicons for simple sentiment classification, without the need for any learning process.

As future work, we believe that extending this work to the emotion paradigm can be effectively done. Also, adding new sources of information to the model could further improve the lexicons quality. Finally, the proposed metric could be improved to a greater extend by adding new semantic structures to the measurement process.

## Acknowledgments

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), EmoSpaces (RTC-2016-5053-7), MOSI-AGIL (S2013/ICE-3019), Somedi (ITEA3 15011). Researcher Oscar Araque has been financed partially by the project Trivalent (H2020 Action Grant No. 740934, SEC-06-FCT-2016).

## References

[1] B. Liu, *Sentiment analysis: Mining opinions, sentiments, and emotions*. Cambridge University Press, 2015.

[2] L. Zhuang, F. Jing, and X.-Y. Zhu, "Movie review mining and summarization," in *Proceedings of the 15th ACM international conference on Information and knowledge management*. ACM, 2006, pp. 43–50.

[3] M. Thomas, B. Pang, and L. Lee, "Get out the vote: Determining support or opposition from congressional floor-debate transcripts," in *Proceedings of the 2006 conference on empirical methods in natural language processing*. Association for Computational Linguistics, 2006, pp. 327–335.

[4] N. Godbole, M. Srinivasaiah, and S. Skiena, "Large-scale sentiment analysis for news and blogs." *ICWSM*, vol. 7, no. 21, pp. 219–222, 2007.

[5] B. Liu and L. Zhang, "A survey of opinion mining and sentiment analysis," in *Mining text data*. Springer, 2012, pp. 415–463.

[6] 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.

[7] A. Esuli and F. Sebastiani, "Sentiwordnet: A high-coverage lexical resource for opinion mining," *Evaluation*, pp. 1–26, 2007.

[8] L. Gatti, M. Guerini, and M. Turchi, "Sentiwords: Deriving a high precision and high coverage lexicon for sentiment analysis," *IEEE Transactions on Affective Computing*, vol. 7, no. 4, pp. 409–421, 2016.

[9] P. D. Turney, "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*. Association for Computational Linguistics, 2002, pp. 417–424.

[10] Y. A. LeCun, L. Bottou, G. B. Orr, and K.-R. Müller, "Efficient backprop," in *Neural networks: Tricks of the trade*. Springer, 2012, pp. 9–48.

[11] G. A. Miller, "Wordnet: a lexical database for english," *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995.

[12] A. Esuli and F. Sebastiani, "Determining the semantic orientation of terms through gloss classification," in *Proceedings of the 14th ACM international conference on Information and knowledge management*. ACM, 2005, pp. 617–624.

[13] A. Andreevskaia and S. Bergler, "Mining wordnet for a fuzzy sentiment: Sentiment tag extraction from wordnet glosses." in *EACL*, vol. 6, 2006, pp. 209–216.

[14] W. Du, S. Tan, X. Cheng, and X. Yun, "Adapting information bottleneck method for automatic construction of domain-oriented sentiment lexicon," in *Proceedings of the third ACM international conference on Web search and data mining*. ACM, 2010, pp. 111–120.

[15] Y. Choi and C. Cardie, "Adapting a polarity lexicon using integer linear programming for domain-specific sentiment classification," in *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2-Volume 2*. Association for Computational Linguistics, 2009, pp. 590–598.

[16] Y. Lu, M. Castellanos, U. Dayal, and C. Zhai, "Automatic construction of a context-aware sentiment lexicon: an optimization approach," in *Proceedings of the 20th international conference on World wide web*. ACM, 2011, pp. 347–356.

[17] G. Qiu, B. Liu, J. Bu, and C. Chen, "Expanding domain sentiment lexicon through double propagation." in *IJCAI*, vol. 9, 2009, pp. 1199–1204.

[18] S. Huang, Z. Niu, and C. Shi, "Automatic construction of domain-specific sentiment lexicon based on constrained label propagation," *Knowledge-Based Systems*, vol. 56, pp. 191–200, 2014.

[19] W. L. Hamilton, K. Clark, J. Leskovec, and D. Jurafsky, "Inducing domain-specific sentiment lexicons from unlabeled corpora," *arXiv preprint arXiv:1606.02820*, 2016.

[20] S. Almatarneh and P. Gamallo, "Automatic construction of domain-specific sentiment lexicons for polarity classification," in *15th International Conference on Practical Applications of Agents and Multi-Agent Systems*, Springer, Ed. Fernando De la Prieta, Zita Vale, Luis Antunes, Tiago Pinto, Andrew Campbell, Vicente Julián, Antonio J.R. Neves, María N. Moreno, 2017.

[21] G. Demiroz, B. Yanikoglu, D. Tapucu, and Y. Saygin, "Learning domain-specific polarity lexicons," in *Data mining workshops (ICDMW), 2012 IEEE 12th international conference on*. IEEE, 2012, pp. 674–679.

[22] J. Staiano and M. Guerini, "Depechemood: A lexicon for emotion analysis from crowd-annotated news," *arXiv preprint arXiv:1405.1605*, 2014.

[23] K. Song, W. Gao, L. Chen, S. Feng, D. Wang, and C. Zhang, "Build emotion lexicon from the mood of crowd via topic-assisted joint non-negative matrix factorization," in *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, pp. 773–776.

[24] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[25] M. F. Porter, "An algorithm for suffix stripping," *Program*, vol. 14, no. 3, pp. 130–137, 1980.

[26] H.-F. Yu, F.-L. Huang, and C.-J. Lin, "Dual coordinate descent methods for logistic regression and maximum entropy models," *Machine Learning*, vol. 85, no. 1, pp. 41–75, 2011.

[27] J. McAuley, C. Targett, Q. Shi, and A. Van Den Hengel, "Image-based recommendations on styles and substitutes," in *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 2015, pp. 43–52.

[28] R. He and J. McAuley, "Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering," in *Proceedings of the 25th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2016, pp. 507–517.

[29] J. Blitzer, M. Dredze, F. Pereira *et al.*, "Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification," in *ACL*, vol. 7, 2007, pp. 440–447.

[30] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.