applied sciences

MDPI

Article

A Toxic Style Transfer Method Based on the Delete–Retrieve–Generate Framework Exploiting Toxic Lexicon Semantic Similarity

Martín Iglesias [1], Oscar Araque [2] and Carlos Á. Iglesias [2,*]

[1] KTH Royal Institute of Technology, 10044 Stockholm, Sweden; martinig@kth.se
[2] Intelligent Systems Group, ETSI Telecomunicación, Universidad Politécnica de Madrid, Avda. Complutense 30, 28040 Madrid, Spain; o.araque@upm.es
* Correspondence: carlosangel.iglesias@upm.es; Tel.: +34-910671900

Abstract: Whether consciously or inadvertently, our messages can include toxic language which contributes to the polarization of social networks. Intelligent techniques can help us detect these expressions and even change them into kinder expressions by applying style transfer techniques. This work aims to advance detoxification style transfer techniques using deep learning and semantic similarity technologies. The article explores the advantages of a toxicity-deletion method that uses linguistic resources in a detoxification system. For this purpose, we propose a method that removes toxic words from the source sentence using a similarity function with a toxic vocabulary. We present two models that leverage it, namely, LexiconGST and MultiLexiconGST, which are based on the Delete –Retrieve–Generate framework. Experimental results show that our models perform well in the detoxification task compared to other state-of-the-art methods. Finally, this research confirms that linguistic resources can guide deep learning techniques and improve their performance.

Keywords: detoxifcation; text style transfer; deep learning; transformers; linguistics; NLP

Citation: Iglesias, M.; Araque, O.; Iglesias, C.Á. A Toxic Style Transfer Method Based on the Delete–Retrieve–Generate Framework Exploiting Toxic Lexicon Semantic Similarity. *Appl. Sci.* **2023**, *13*, 8590. https://doi.org/10.3390/app13158590

Academic Editor: Gianluca Lax

Received: 19 June 2023
Revised: 17 July 2023
Accepted: 23 July 2023
Published: 26 July 2023

1. Introduction

Digital social communications have become mainstream, enabling citizens to freely express their opinions and providing visibility to minority groups. However, some phenomena, such as toxic language, can become severe barriers that limit fair and nondiscriminatory participation in public forums. Toxic language can be defined as "rude, disrespectful, or unreasonable language that is likely to make someone leave a discussion." [1]. Still, it includes, but is not limited to, other language classification terms, such as hate speech, vulgarity, sexism, racism, or bullying, to cite a few. Its devastating effects and importance have pushed governments and international institutions combat its effects using legislative measures, mainly focused on its detection, to limit its spread. The research community has made considerable effort to contribute to this effort with two tasks: automatic toxic language detection [2–4] and detoxifying language models [5,6]. On the contrary, the task of rephrasing offensive content automatically requires further research. This task has great potential for practical applications, such as promoting a healthier online environment. By suggesting alternative, less emotionally charged language, an automatic rephrasing system could encourage users to reconsider their wording and express their thoughts neutrally. The main approach followed for text detoxification [7–9] is based on text style transfer.

Text style transfer is "a significant Natural Language Generation (NLG) task whose objective is to rephrase the text and maintain its content while altering its style" [10]. Applications range from modifying the conversational style of dialogue agents [11] to masking personal attributes such as gender to protect privacy [12]. Additionally, the task can adjust the formality of texts [13] or even generate poetry, as shown by Yang et al. [14]. The main limitation to developing effective style transfer systems is the lack of parallel

corpora containing sentences in one style and their corresponding counterparts in another, with the original meaning preserved. Consequently, researchers have developed techniques that bypass the requirement for parallel corpora altogether.

Previous studies in the nonparallel data setting try to adversarially train encoder–decoder networks that learn disentangled representations of style and content from text [15–17] with a technique known as "implicit style–content disentanglement" (Section 2). Unfortunately, these models suffer from (a) the sparsity of their latent representations, which makes them take longer to converge and harder to train, (b) a lack of precise control over the style generated, and (c) the need to train from scratch to swap the trade-off between style transfer and content preservation. Other works approach the task with "explicit style–content disentanglement" (Section 2), such as Delete–Retrieve–Generate (DRG) [18], to build on the inductive bias that style attributes are frequently related to a specific set of tokens in text. Although Delete–Retrieve–Generate (DRG) is a step up in quality over previous research, its Delete–Generate approaches are prone to (a) disturb crucial context with the accidental replacement of core context words and (b) miss the replacement of source style words with target style words, and (c) the encoder–decoder generates nonfluent sentences and struggles with longer input text given its Long Short-Term Memory (LSTM) nature.

Sudhakar et al. [19] introduced a novel approach that overcomes some of the drawbacks mentioned above. Their system, called BlindedGST, leverages the attention weights of an unsupervised pre-trained large language model to identify style attributes and removes the need for an encoder–decoder LSTM with the use of a style conditional generative Transformer [20] that can handle long text input and generate fluent text from a sentence whose style words have been removed. The system has been evaluated for sentiment-, political-, formality-, and gender-detection tasks, leaving out the detoxification task.

In this work, we investigate the following research questions (RQs):

RQ1: Can a current nontoxic specific text style transfer method be applied to the detoxification task?. As described above, many current methods build on the DRG framework to achieve consistent results among different styles, and we want to validate their performance on the detoxification task. More specifically, we are interested in the methods that operate on a nonparallel data setting using only the Delete and Generate steps, i.e., not the Retrieve step. The two best models in this category (that we are aware of) are Mask and Infill [21] and BlindedGST [19]. We choose BlindedGST as the baseline, since it has been evaluated on multiple downstream tasks as explained above. In contrast, Mask and Infill has only been applied to sentiment transfer, and, therefore, we understand that BlindedGST is a better starting point since we expect it to generalize better to the detoxification task. Thus, we evaluated BlindedGST on several toxicity datasets.

RQ2: Can we improve this attention-based method by using linguistic resources?. As explained above, BlindedGST operates in a two-step process in which it first removes the style from the sentence and then uses a generator to transfer it to the target style. Therefore, we want to explore the advantages of a style deletion method that uses linguistic resources such as a lexicon instead of an attention-based system. For this purpose, we propose a method that removes toxic words from the source sentence using a similarity function with a toxic lexicon. Finally, our main contributions are LexiconGST and MultiLexiconGST, which are evaluated with different lexicons and similarity functions that have led, in most cases, to better performance than the BlindedGST baseline.

The remainder of this article is organized as follows. Section 2 introduces the related work. Then, Section 3 describes the proposed model. Next, Sections 4 and 5 describe the experiments and evaluation of the model, respectively. Lastly, Section 6 discusses the conclusions of this work.

2. Related Work

A common approach to tackle the style transfer task involves using a supervised encoder–decoder model to "translate" the source sentence into the desired target style [13]. When the source and target belong to the same language, one can leverage pre-trained Language Models (LMs) such as Generative Pre-trained Transformer (GPT) [22] to accomplish this task. This technique can be enhanced by fine-tuning the LMs on relatively small parallel corpora, leading to favourable outputs [23]. Nevertheless, the application of this method is somewhat limited due to the scarcity of sufficiently large parallel data. In contrast, most previous research on text style transfer has been carried out using unsupervised algorithms.

Four approaches are considered, which are described below: explicit style–content disentanglement, implicit style–content disentanglement, without style–content disentanglement, and toxic style transfer.

The first approach is known as "explicit style–content disentanglement" [18]. This style transfer method is relatively straightforward, but highly effective, and involves leaving the sentence unchanged and modifying only the individual words linked to the desired style. The first attempt to achieve such a transfer was introduced by the Delete–Retrieve–Generate [18] framework, where the authors suggest DRG-TemplateBased. First, it removes the style-related words from the input sentence, for example "nasty", and retains only the content-specific information such as "food". Next, it replaces source-style words with target-style words with the help of a retrieval mechanism based on sentences similar in content and a Seq2Seq model to alleviate fluency. Ultimately, the performance depends on the robustness of the replacement mechanism. Li et al. [18] also propose two models which are focused on two phases of DRG, known as DeleteOnly and DeleteAndRetrieve (D&R). Typically, words associated with a particular style are identified based on their frequency in the corpus. Conversely, Sudhakar et al. [19] propose improving the DRG model by using a Delete Transformer (DT) and a Generative Style Transformer (GST). They propose two variants that train the Generative Style Transfomer (GST) in two different ways: BlindedGST (B-GST) and GuidedGST (G-GST). The first variant, B-GST, consists of training the model so that it generates an output sentence given only the content and target style, being blind to specific desired attributes. It is relevant that based on Li's work [18], this variant uses a weak retrieval component, so it can be considered a D&G model. In the second variant, G-GST, the model is guided towards generating a target sentence with target attributes. In our work, we extend the DRG framework following the previously presented extensions [18,19] so that the retrieval component is not used, and we focus our research on improving two components: Delete and Generate.

The second approach that can be followed is "implicit style–content disentanglement". Rather than performing a straightforward replacement of specific parts in the text, some style transfer models attempt to learn the latent representations of content and style from the input corpus [17]. The model then combines the source text content's representation with the target style's representation to generate the desired text. Fu et al. [17] presented a representative study in the field in which they used adversarial learning to train the encoder of an encoder–decoder network where the input text is encoded without style information ($\mathbf{z}$). The decoder generates texts in different target styles when fed with $\mathbf{z}$. A major concern is whether the reconstruction loss is enough to keep the semantics of the input. To improve this approach, other works have proposed a new loss of cycle consistency [24] that also alleviates the problem of model collapse. Yin et al. [25] focused on complementary methods to improve this adversarial learning technique, such as comparators to check for content preservation and style change.

In contrast to previous models, the third approach called "without style–content disentanglement" is not based on style–content disentanglement. DualRL [26] employs a distinct methodology in which style transfer occurs directly from the source to the target. The model is paired with a dual task involving back transfer to the source style to facilitate training. By incorporating this technique, the model can be trained without

needing parallel data. He et al. [27] developed the Deep Latent Sequence Model (DLSM), which performs joint training of models in both primal and dual tasks. This is achieved by implementing amortized variational inference, a powerful computational technique that efficiently approximates complex probability distributions. Building on this end-to-end approach, Lee et al. introduced Stable Style Transformer (SST) [28], which involves training two sequence-to-sequence transformers for primal and dual tasks. To enhance the discriminative capabilities of the model, the method incorporates the cross-entropy of a pre-trained style classifier as an additional loss function. Krishna et al. [29] proposed the Style Transfer as Paraphrase (STRAP) method, in which a style transfer model is considered to be a paraphraser that introduces stylistic attributes into a given text. The authors used a pre-trained general-purpose paraphraser to generate the necessary training data to transform styled texts into neutral texts, thereby generating pseudoparallel data. The resulting neutral-to-styled parallel datasets were then used to train seq2seq models.

The fourth approach is known as "toxic style transfer". Text detoxification is a rather new downstream task from the text style transfer (TST) task [7–9,30]. The pioneering work in this field by dos Santos et al. [7] used nonparallel data to train an end-to-end seq2seq model, which learned to preserve content and transfer style through the combination of autoencoder loss, style classification loss, and cycle consistency loss. Tran et al. [8] proposed a pipeline approach to text detoxification. First, a search engine is used to identify nontoxic sentences similar in content to the input toxic sentence. Next, a masked language model (MLM) is used to fill in any gaps or missing words that do not match the identified sentences. Finally, a seq2seq model improves the fluency of the generated text. Laugier et al. [9] remove the toxic style from the text by fine-tuning T5 as a denoising autoencoder that learns on nonparallel data by reconstructing original noised sentences. Dale et al. [30] present CondBERT, a fine-tuned BERT on toxic data that allows the control of the style of replaced words and better content preservation, as it can perform multi-word replacements. They also introduce ParaGeDi, which extends the GeDi model [31], a discriminative generator, by plugin a paraphraser that shares vocabulary with the GeDi, allowing the model to combine the language distribution of the paraphraser with a style-conditional language distribution that results in different paraphrases of the input text depending on the target style. Lastly, Logacheva et al. [32] collected ParaDetox, a large toxic-to-neutral parallel corpus curated from ParaNMT [33], and fine-tuned BART [34] in this dataset to obtain state-of-the-art (SOTA) results in the detoxification task.

To conclude, a comparison of our methods within the literature scope is presented. Specifically, the works most similar to our work are Mask and Infill [21], BlindedGST [19], and SST [28], which also perform style transfer with a two-step process: Delete and Generate. For the Generate step, we simply modify BlindedGST's method to work with toxic style, but in the Delete step, we present major differences to these methods. First, we use two different style deletion algorithms simultaneously during training, one for the neutral style and one for the toxic style, unlike the literature methods, which use the same algorithm for all styles. However, this neutral style deletion algorithm used in training is simply the attention-based method presented by Sudhakar et al. [19]. Second, the toxic style deletion algorithm we present is similar to the method proposed by Mask and Infill, but instead of using a combination of a frequency-based vocabulary and attention scores, we use a predefined or regression-based toxic vocabulary and a style similarity function. Third, the other two works, BlindedGST and SST, are very different in this matter since they use pre-trained classifier attention scores and an importance score through pre-trained classifier predictions, respectively. Finally, our methods build on BlindedGST because of the similarity in the neutral style deletion algorithm and the generation step, while the novelty lies in the toxic style deletion algorithm.

3. Proposed Approach

This section offers a formalization of the problem to solve and describes how the proposed models approach it.

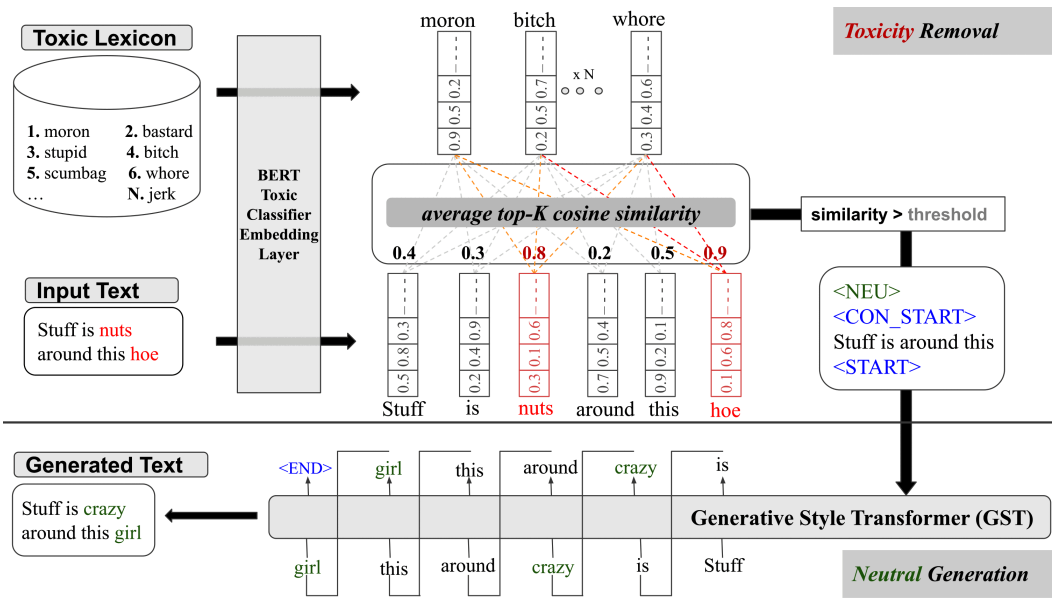
Firstly, the formalization of the problem considers a dataset **D** formed by sentences in natural language of different styles. Each example in the dataset is $(\mathbf{x_i}, \mathbf{s_i})$, in which $\mathbf{x_i}$ is a sentence and $\mathbf{s_i} \in \mathbf{S}$ is the style of the sentence. We refer to the text in the same style as the text that shares specific attributes, that is, $\mathbf{S} = \{positive, formal, toxic, neutral, \dots\}$. We denote **c** as the set of words that represent the content or information that the sentence contains, such that $Style(\mathbf{c}) \notin \mathbf{S}$, and **a** as the collection of words with stylistic information. Therefore, the goal of text style transfer is to, given a source sentence **x** with source style $\mathbf{s^{src}}$ and a desired target style $\mathbf{s^{tgt}}$, output a new sentence **y** that maintains as much content **c** as possible from **x** but whose style is $\mathbf{s^{tgt}}$. In particular, in the detoxification task, we consider $\mathbf{S} = \{toxic, neutral\}$ where $\mathbf{s^{src}} = toxic$ and $\mathbf{s^{tgt}} = neutral$.

In order to generate this transferred style sentence **y** where $Style(\mathbf{y}) = \mathbf{s^{tgt}}$ given an input sentence **x** with style $\mathbf{s^{src}}$, our LexiconGST model tries to learn $P(\mathbf{y}|\mathbf{x}, \mathbf{s^{tgt}})$. We base our work on the BlindedGST [19] method; hence, the LexiconGST model is a two-step process: style deletion and style-conditioned generation.

First, we describe the style deletion component, which aims to separate style and content from the input sentence by modeling the distribution $P(\mathbf{c}, \mathbf{a}|\mathbf{x})$. The original sentence **x** can be fully reconstructed from **c** and **a**. For example, given "your stupid contributions are shit", then the toxic-related words, "stupid" and "shit", should be removed. This approach is founded on the "input reduction" method proposed in the work of Feng et al. [35], which suggests that the style of a sentence is heavily affected by the contribution of specific words or sub-phrases associated with such style. Consequently, a style classifier would be confused about the style of a sentence **x** whose style is **s** if the style-related attributes **a** are removed from the sentence. To detect these style attributes, we use a weighting algorithm to measure the contribution to style that each token in **x** makes.

In the case of neutral style, we measure the contribution of a token using the method of Sudhakar et al. [19]. A BERT classifier is fine-tuned for toxic sentence classification, and then it is used to filter out neutral tokens based on the attention weights of a selected attention head. However, when detecting and removing toxic tokens, we propose an algorithm (see Algorithm 1) that exploits token embedding similarities with a toxic lexicon. First, we use the embedding layer from a RoBERTa [36] toxicity classifier because these embeddings encode some style information. Since we are interested in the style similarity of tokens rather than semantic similarity, we follow the intuition that this embedding representation is better for our task than other semantic embeddings. Then, we employ a similarity function between the sentence's token embeddings and the lexicon word embeddings and remove tokens whose similarity is above a predefined threshold.

Unfortunately, due to how word embeddings are generated, some words like "betrayed" (word inside one of the toxic lexicons used) and "portrayed" have high embedding similarity, which will result in the deletion of "portrayed"" from the source sentence as if it were a toxic style attribute, but in reality it should remain as part of the source content. Therefore, instead of using the maximum similarity of a token with the words in a lexicon, we introduce a parameter K that alleviates this issue by averaging the top-K similarities of a word with a lexicon. This technique follows the intuition that for a given lexicon, in the case of "ass" (clearly a word to remove), the top similarities are "ass": 1.0, "dumbass": 0.81, and "asshole": 0.80, ... so the average top-K similarity is still high for "*ass*" and the model will correctly delete it from the source text. Still, in the case of "*portrayed*", the top similarities are *"betrayed": 0.75, "stoned": 0.42, and "demented": 0.41*, so the top-K average is reduced compared to the maximum and therefore "*betrayed*" is not removed from the source sentence as desired. Furthermore, we found that again due to the nature of word embeddings, very common words like "*had*" or "*they*" tend to have a high similarity with a lot of words and are, in turn, deleted as if they were toxic style attribute words. To alleviate this phenomenon, we have defined the MultiLexiconGST model. This model extends LexiconGST and utilizes an additional lexicon of common words and stop words to ensure that these types of words are not deleted from the source sentence.


Toxic Lexicon
1. moron
2. bastard
3. stupid
4. bitch
5. scumbag
6. whore
…
N. jerk
BERT Toxic Classifier Embedding Layer
Input Text
Stuff is nuts around this hoe
moron
bitch
whore
x N
average top-K cosine similarity
0.4
0.3
0.8
0.2
0.5
0.9
Stuff
is
nuts
around
this
hoe
similarity > threshold
<NEU>
<CON_START>
Stuff is around this
<START>
Toxicity Removal
Generated Text
Stuff is crazy around this girl
<END>
girl
this
around
crazy
is
Generative Style Transformer (GST)
girl
this
around
crazy
is
Stuff
Neutral Generation

removed from the sentence if this similarity exceeds a predefined threshold. Then, a neutral sentence is generated from the toxicity-stripped sentence *"Stuff is around this"*, **c**, by using a style-conditional transformer conditioned on neutral style, $\mathbf{s}^{\mathbf{tgt}}$. As explained previously, MultiLexiconGST extends the LexiconGST's *style deletion component* by using an additional lexicon to preserve common or stop words to alleviate the issues presented below.

4. Experimental Design

In this section, we describe the datasets and lexicons used to evaluate our methods (Section 4.1), as well as define the metrics we use to evaluate them (Section 4.2). In addition, we specify the learning and inference approaches. Finally, we describe the details of the implementation of the model in Section 4.3.

4.1. Resources

To train and evaluate the performance of our style transfer model, we used two datasets that allowed us to compare our results to other works and make our results significant: (i) the English dataset from the previous Jigsaw [37] competition since it is widely used among the most recent detoxification literature [9,30]; and the (ii) ParaDetox dataset [32], which was recently released and designed specifically for the task of detoxification. Since the Jigsaw dataset has no parallel (source-to-target) sentences, we generated a neutral and toxic corpus separately. To construct the toxic corpus, the comments labeled as toxic were first split into individual sentences as the original comments may be too lengthy. Subsequently, each sentence was classified using the RoBERTa-based toxicity classifier from Dale et al. [30], which was trained on English data from the three Jigsaw datasets [37–39]. The toxic half of the dataset consists of sentences classified as toxic by the toxicity classifier, resulting in around 140,000 sentences (see Table 1). An equal number of nontoxic sentences were randomly selected from the sentence-separated Jigsaw data to construct the neutral half of the dataset. The test set was constructed using a similar approach to the Jigsaw competition, where 10,000 sentences with the highest toxicity scores were selected based on the toxicity classifier. On the other hand, ParaDetox is a parallel detoxification dataset collected from the large ParaNMT [33] dataset with the use of a crowd-sourced quality control pipeline. It consists of over 12,000 toxic sentences, each one having 1 to 3 neutral paraphrases. To build the toxic corpus, we just used the toxic sentences, and for the neutral corpus, we merged all available neutral sentences, which resulted in around 20,000 neutral sentences.

Table 1. Number of instances per style and split of used datasets.

Dataset	Style	Train	Dev	Test	Total
ParaDetox [32]	Toxic	11,351	596	671	12,618
	Neutral	19,274	491	671	20,436
Jigsaw [37]	Toxic	135,390	6648	10,000	152,038
	Neutral	135,390	6648	10,000	152,038

We used the lexicons shown in Table 2 to experiment with our models, where five different lexicons were for toxic style deletion, and one lexicon was for the preservation of common and/or stopwords in MultiLexiconGST. The Abusive, Hurtlex, and Orthrus lexicons were taken from other works whose whole purpose was to craft such a lexicon. The other two lexicons, ParaDetox-Lex and Jigsaw-Lex, are a contribution of this work. We refer to them as dataset-specific lexicons since they were crafted specifically for each dataset, Jigsaw or Paradetox. To generate these lexicons, we first trained, for each dataset, a logistic regression toxicity classifier that approximates the weights **W** and biases **B** for the model $Y = \mathbf{WX} + \mathbf{B}$, where Y is either toxic or nontoxic and **X** is a vector representing the words in a sentence. Then, we interpreted the weights **W** given by the logistic regressor to

each word as a toxicity score. Finally, we used a threshold **t** for this toxicity score w_{word} to select which words are included in the lexicon if $w_{word} > t$.

Table 2. Lexicons used in the experiments and their main characteristics.

Lexicon	Size	Domains	Method
Abusive [40]	2961	Abuse, Hate	Linguistic features
Hurtlex [41]	1160	Xenophobia, Immigrant, Misogyny, Insults	WordNet, Manual
Orthrus [42]	1929	Toxicity	Toxic span
Paradetox-lex	503	Toxicity	Toxic Classifier
Jigsaw-Lex	2371	Toxicity	Toxic Classifier
Stopwords [43]	1298	Common words, Stopwords	Crowdsource

4.2. Metrics

Since we approach the problem in a nonparallel fashion, metrics such as BLEU [44], METEOR [45], or ROUGE [46] do not have references for their evaluation. Style transfer models are expected to change the style of a sentence while preserving its content and fluently producing text. These three factors are often in conflict with each other, so a composite metric is needed to strike a balance between them. Our evaluation methodology is based on Krishna et al. [29]. We use the J metric, which is calculated by multiplying three individual metrics at the sentence level: style accuracy, content preservation, and fluency. System-level (J) is then obtained by taking the average of sentence-level J scores.

$$J = ACC \cdot SIM \cdot FL.$$

Sentence-level style accuracy (ACC) is evaluated using the pre-trained toxicity classifier from Dale et al. [30]. *Content preservation (SIM)* is evaluated by comparing the sentence-level embeddings of the original and transformed texts, calculated using the model proposed by Wieting et al. [47]. *Fluency (FL)* is measured using a classifier trained on the CoLA dataset to determine linguistic acceptability [48].

4.3. Implementation Details

For training, since we used a nonparallel data setting, we used the teacher-forcing decoding strategy [19] over the generated tokens of GST to minimize the reconstruction loss. Formally, GST learns to maximize the following function:

$$L(\mathbf{\Theta}) = \sum_{\mathbf{x},\mathbf{s^{src}} \in \mathbf{D}} log[P(\mathbf{x}|\mathbf{c}, \mathbf{s^{src}}; \mathbf{\Theta})]$$

where for an input **x**, GST learns to reconstruct this input by generating $\mathbf{y} = \mathbf{x}$ given **c** and the source style $\mathbf{s^{src}}$. In contrast, at test time, we generated the output sentence with the use of beam search with a beam width of 5 and a look-left window of 1.

Specifically, we used the PyTorch implementation of the GPT model [22] offered by the HuggingFace Transformers library (https://huggingface.co/openai-gpt, accessed on 22 July 2023). The model is pre-trained on 7000 books from the BookCorpus dataset (https://huggingface.co/datasets/bookcorpus, accessed on 22 July 2023). Its architecture accepts sequences of up to 512 tokens, and has 12 layers and 12 attention heads per block. The dimension for all internal states is 768, which includes keys, queries, values, and embeddings. Byte Pair Encoding (BPE) [49] tokenizes input text. More details can be found in the source code (https://github.com/martinigoyanes/LexiconGST, accessed on 22 July 2023).

5. Results

To analyze the performance of the models, we focused attention on the aggregated metric J. More specifically, we wanted to evaluate if the general style transfer method proposed by Sudhakar et al. [19] is valid for the detoxification task and if its model can be improved by changing its style deletion system from attention-based to lexicon-based. Additionally, the effects of different similarity functions and lexicons were analyzed. In this sense, it is important to consider that the evaluation is limited by the accuracy of the pre-trained toxic and language acceptability classifiers, as well as the performance of the model that compares sentence embeddings. Lastly, we selected the different thresholds used in the experiments. To remove toxic words based on their lexicon similarity, we picked $similarity\_thres = 0.7$ after tuning the parameters. On the contrary, for the threshold that determines whether a word is included in our dataset-specific lexicon, we chose a value that produces a lexicon size similar to the lexicons we were working with, i.e., for Jigsaw $toxicity\_thres = 0.8$ and for ParaDetox $toxicity\_thres = 0.7$.

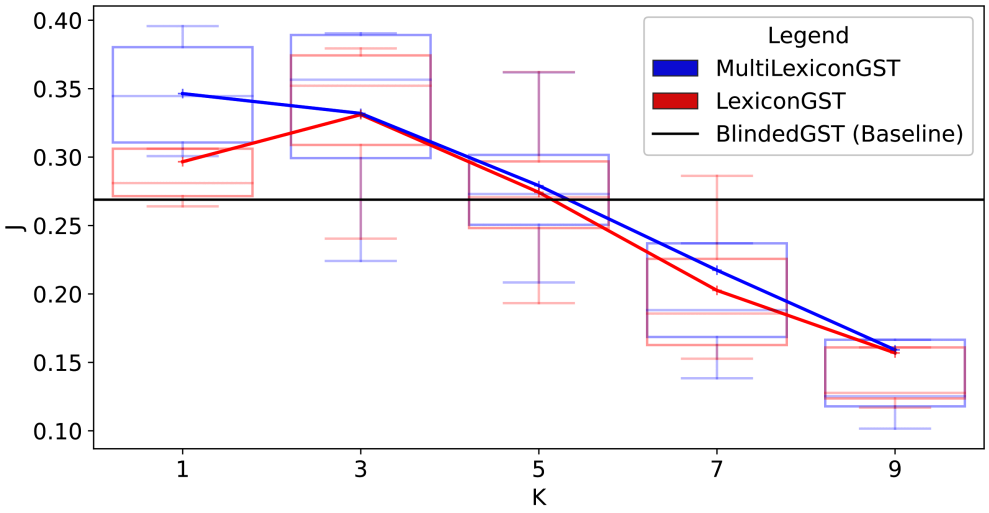
Firstly, we evaluated LexiconGST and MultiLexiconGST on multiple lexicons and used different average top-K similarity functions to find that performance depends a lot on the lexicon and the value of K. However, we discovered that, regardless of the lexicon, the best K are $\{1, 3, 5\}$ by performing a grid search on the validation split. Specifically, Figure 2 shows how similarity functions that consider $K > 5$ words for average similarity always lead to a significant decrease in performance in contrast to when $K < 5$ words are considered. We also observe from this wide evaluation how the fluency (FL) and source similarity tend to increase as the style transfer strength (ACC) decreases; our intuition is that the smaller the change in style, the fewer the words that are changed from the input, which leads to a more fluent and similar sentence to the input. Secondly, in Table 3, we present our best-performing models for each lexicon and show that both MultiLexiconGST and LexiconGST outperform BlindedGST [19]. The proposed dataset-specific lexicons almost always perform best, while the Hurtlex lexicon performs the worst. More specifically, when MultiLexiconGST makes use of the dataset-specific lexicons, it yields the best J score since it is capable of decently changing the style (ACC) while maintaining high fluency (FL) thanks to the leverage of the extra lexicon Stopwords-en, which we believe allows it to be more accurate when deleting toxic words. Concretely, the biggest increase in quality ($\uparrow 0.18$) over BlindedGST is in style transfer strength (ACC) for the Jigsaw dataset. In contrast, an improvement ($\uparrow 0.19$) in fluency (FL) makes the difference for the ParaDetox dataset.

In order to further study the impact of the proposed methods and their performance, we conducted the Friedman statistical test [50]. This test computes a sorted ranking of approaches, offering an aggregated view of their performance across all datasets. In this way, a lower ranking indicates a better result for a certain method in comparison to the rest. Ties are resolved by averaging the ranks obtained. We performed the Friedman test with $\alpha = 0.05$, rejecting the null hypothesis.

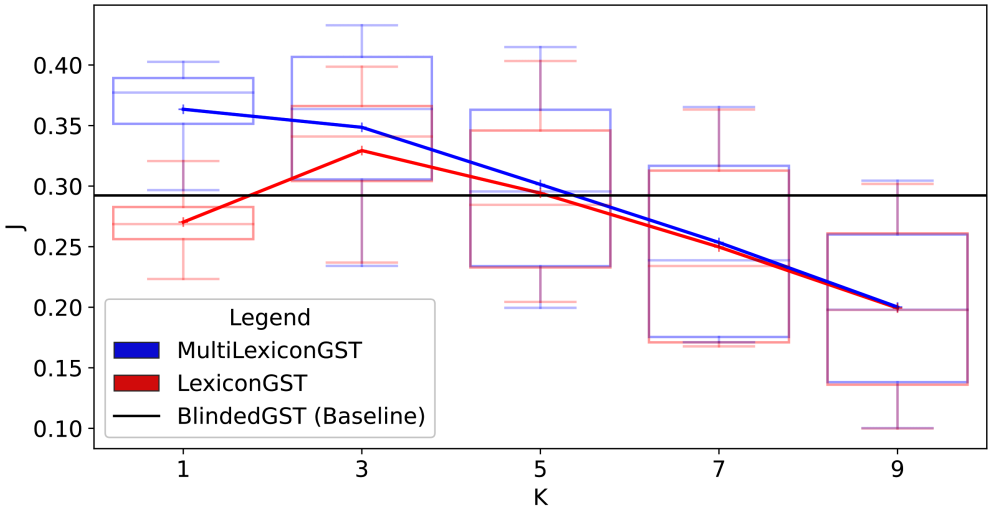
In Table 4, it can be seen that the best model, as indicated by the Friedman test, is the MultiLexiconGST using a dataset-specific lexicon. This reinforces the observation that using domain-oriented resources leads to a performance improvement, and is consistent with our initial hypotheses. In comparison, BlindedGST is placed at the bottom of the ranking, which again indicates that introducing relevant linguistic resources effectively improves the style transfer process.

Finally, we compared our methods with the state-of-the-art (SOTA) models shown in Table 5. To obtain a fair comparison of the methods, we left out all methods with a nontrivial "Retrieval" component (Delete–Retrieve–Generate). For example, we did not consider CondBERT [30] as the authors mention "Reranks" replacement words suggested by BERT based on the similarity of suggestions with the word to be replaced, and we consider this a retrieval method. We also left out ParaGeDi [30] since it makes use of a pre-trained paraphraser model, which we consider to be a large generalized retrieval model. Both ParaGeDi and CondBERT outperform our system, given that they use an additional component that enhances their performance. STRAP [29] uses a paraphrase as well, so it

was not considered. Finally, DLSM [27] uses a very different approach (probabilistic-based rather than framework-based) from all other methods. However, both methods perform much worse than MultiLexiconGST: DLSM generates very nonfluent text and STRAP has very weak style transfer. This collection of State of the Art (SOTA) methods is shown in Table 5, where we show how our best model, MultiLexiconGST with a dataset-specific lexicon and an average top-three similarity function, not only outperforms by a wide margin the baseline BlindedGST but also has the best performance among all known competitor models. In addition, MultiLexiconGST also generates the most fluent (FL) and similar sentences to the input (SIM). Lastly, although MultiLexiconGST is behind Mask and Infill [21] or SST [28] in style transfer strength (ACC), these models perform worse in terms of fluency (FL). We attribute this to the abrupt addition of tokens associated with the target style without maintaining fluency.



(**a**)



(**b**)

Figure 2. Performance as K increases on the validation set of both Jigsaw and ParaDetox: (**a**) ParaDetox. (**b**) Jigsaw.

Table 3. Results of the best-performing models for each lexicon on the test splits of the Jigsaw and Paradetox datasets. Best J results are indicated by bold numbers.

Dataset	Model	Lexicon	K	ACC	SIM	FL	J	BLEU
Paradetox	Lexicon GST	Abusive	5	0.70	0.89	0.75	0.47	0.77
		Hurtlex	3	0.48	0.92	0.79	0.35	0.82
		Orthrus	1	0.78	0.85	0.69	0.47	0.73
		Dataset-specific	1	0.81	0.83	0.60	0.42	0.70
	MultiLexicon GST	Abusive	1	0.80	0.83	0.71	0.48	0.72
		Hurtlex	1	0.65	0.89	0.77	0.45	0.78
		Orthrus	5	0.63	0.89	0.83	0.47	0.77
		Dataset-specific	3	0.71	0.89	0.75	**0.49**	0.77
Jigsaw	Lexicon GST	Abusive	3	0.52	0.85	0.66	0.30	0.75
		Hurtlex	1	0.45	0.86	0.61	0.24	0.76
		Orthrus	3	0.75	0.80	0.55	0.32	0.69
		Dataset-specific	3	0.78	0.77	0.58	0.35	0.69
	MultiLexicon GST	Abusive	1	0.64	0.80	0.65	0.34	0.72
		Hurtlex	1	0.43	0.87	0.70	0.27	0.78
		Orthrus	3	0.66	0.82	0.67	0.37	0.74
		Dataset-specific	3	0.73	0.79	0.64	**0.38**	0.71

Table 4. Friedman rank of the best 9 methods.

Model	Lexicon	Rank
MultiLexiconGST	Dataset-specific	1.0
MultiLexiconGST	Abusive	3.0
MultiLexiconGST	Orthrus	3.0
LexiconGST	Orthrus	4.5
LexiconGST	Abusive	5.0
LexiconGST	Dataset-specific	5.0
MultiLexiconGST	Hurtlex	6.75
BlindedGST	-	8.25
LexiconGST	Hurtlex	8.5

Table 5. SOTA performances of methods that use a Delete and Generate framework. Best J results are indicated by bold numbers.

Dataset	Jigsaw				Paradetox			
Model	**ACC**	**SIM**	**FL**	**J**	**ACC**	**SIM**	**FL**	**J**
MultiLexiconGST w/K = 3 and w/Dataset-specific Lexicon	0.73	0.79	0.64	**0.38**	0.71	0.89	0.75	**0.49**
Mask and Infill [21]	0.78	0.80	0.49	0.31	0.91	0.82	0.63	0.48
BlindedGST [19]	0.55	0.79	0.61	0.27	0.72	0.89	0.56	0.32
SST [28]	0.80	0.55	0.12	0.05	0.86	0.57	0.19	0.10

6. Conclusions and Future Work

In this paper, we introduce a novel algorithm for toxic style removal and present two models that leverage it, LexiconGST and MultiLexiconGST, which are based on the DRG framework. The experimental results show that our models have competitive performance in the detoxification task among the state-of-the-art methods. Specifically, our proposed methods outperform all SOTA models, which are based on the same framework. Furthermore, this research also confirms that linguistic resources can guide deep learning and improve its performance. We have also shown that the hyperparameter *K* helps the similarity function to adapt to the characteristics of the different lexicons.

The proposed models focus on the Delete and Generate phases, which constitutes a limitation of our models. Indeed, including a mechanism that implements a nontrivial

Retrieve phase can enhance the overall performance of the system. As future work, this can be addressed by evaluating whether the proposed models can outperform SOTA methods that use a more complex Retrieve step like CondBERT and ParaGeDi [30].

Furthermore, the proposed methods use lexicons as a source of domain-centered knowledge sources. Nevertheless, deriving these lexicons and evaluating their quality are open challenges, which is a limitation of this work. This opens new avenues of research, oriented to generating and exploiting lexicons to improve the results of the task.

Finally, another line of research that emerges from our work is whether our lexicon-based approach not only outperforms the attention-based approach of Sudhakar et al. [19] on the detoxification task, but that our method can be generalized to other downstream style transfer tasks by simply selecting the appropriate style lexicon.

Author Contributions: Conceptualization: M.I., O.A. and C.Á.I.; methodology: M.I., O.A. and C.Á.I.; validation: M.I.; draft preparation: M.I., O.A. and C.Á.I.; writing—review and editing: M.I., O.A. and C.Á.I. All authors have read and agreed to the published version of the manuscript.

Funding: This research This work was supported by the European Union's Horizon 2020 Research and Innovation Programme under project Participation (grant agreement no. 962547) and by the Spanish Ministry of Science and Innovation through the COGNOS project (PID2019-105484RB-I00).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Code and data are available at https://github.com/martinigoyanes/LexiconGST, accessed on 22 July 2023.

Conflicts of Interest: The funders had no role in the design, writing, or decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

ACC	Sentence-level style accuracy
BLEU	Bilingual Evaluation Understudy
DLSM	Deep Latent Sequence Model
DRG	Delete-Retrieve-Generate
GPT	Generative Pre-trained Transformer
GST	Generative Style Transformer
LM	Language Model
LSTM	Long Short-Term Memory
MLM	Mask Language Modeling
NLG	Natural Language Generation
SOTA	State-of-the-Art
SST	Stable Style Transformer
SOTA	State of the Art

References

1. Jigsaw. The Toxicity Issue. Can Technology Help Improve Conversations Online? Available online: https://jigsaw.google.com/the-current/toxicity/ (accessed on 8 May 2023).
2. Zampieri, M.; Nakov, P.; Rosenthal, S.; Atanasova, P.; Karadzhov, G.; Mubarak, H.; Derczynski, L.; Pitenis, Z.; Çöltekin, Ç. SemEval-2020 Task 12: Multilingual Offensive Language Identification in Social Media (OffensEval 2020). In Proceedings of the Fourteenth Workshop on Semantic Evaluation, Online, 12–13 December 2020; pp. 1425–1447. [CrossRef]
3. D'Sa, A.G.; Illina, I.; Fohr, D. Towards Non-Toxic Landscapes: Automatic Toxic Comment Detection Using DNN. In Proceedings of the 2nd Workshop on Trolling, Aggression and Cyberbullying, Marseille, France, 16 May 2020; pp. 21–25.
4. Han, X.; Tsvetkov, Y. Fortifying Toxic Speech Detectors Against Veiled Toxicity. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), Online, 12 November 2020; pp. 7732–7739. [CrossRef]
5. Welbl, J.; Glaese, A.; Uesato, J.; Dathathri, S.; Mellor, J.; Hendricks, L.A.; Anderson, K.; Kohli, P.; Coppin, B.; Huang, P.S. Challenges in Detoxifying Language Models. In Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2021, Punta Cana, Dominican Republic, 16–20 November 2021; pp. 2447–2469. [CrossRef]

6. Xu, C.; He, Z.; He, Z.; McAuley, J. Leashing the Inner Demons: Self-Detoxification for Language Models. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtual, 22 February–1 March 2022; Volume 36; pp. 11530–11537.
7. Nogueira dos Santos, C.; Melnyk, I.; Padhi, I. Fighting Offensive Language on Social Media with Unsupervised Text Style Transfer. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Melbourne, Australia, 15–20 July 2018; pp. 189–194. [CrossRef]
8. Tran, M.; Zhang, Y.; Soleymani, M. Towards A Friendly Online Community: An Unsupervised Style Transfer Framework for Profanity Redaction. In Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Online, 8–13 December 2020; Scott, D., Bel, N., Zong, C., Eds.; International Committee on Computational Linguistics: New York, NY, USA, 2020; pp. 2107–2114. [CrossRef]
9. Laugier, L.; Pavlopoulos, J.; Sorensen, J.; Dixon, L. Civil Rephrases of Toxic Texts With Self-Supervised Transformers. *arXiv* **2021**, arXiv:2102.05456.
10. Jin, D.; Jin, Z.; Hu, Z.; Vechtomova, O.; Mihalcea, R. Deep Learning for Text Style Transfer: A Survey. *Comput. Linguist.* **2022**, *48*, 155–205. [CrossRef]
11. Zhou, G.; Luo, P.; Cao, R.; Lin, F.; Chen, B.; He, Q. Mechanism-aware neural machine for dialogue response generation. In Proceedings of the 31st AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017.
12. Reddy, S.; Knight, K. Obfuscating Gender in Social Media Writing. In Proceedings of the First Workshop on NLP and Computational Social Science, Austin, TX, USA, 5 November 2016; pp. 17–26. [CrossRef]
13. Rao, S.; Tetreault, J. Dear Sir or Madam, May I Introduce the GYAFC Dataset: Corpus, Benchmarks and Metrics for Formality Style Transfer. In Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), New Orleans, LA, USA, 1–6 June 2018; pp. 129–140. [CrossRef]
14. Yang, C.; Sun, M.; Yi, X.; Li, W. Stylistic Chinese Poetry Generation via Unsupervised Style Disentanglement. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, 31 October–4 November 2018; pp. 3960–3969.
15. Hu, Z.; Yang, Z.; Liang, X.; Salakhutdinov, R.; Xing, E.P. Toward Controlled Generation of Text. In Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; Precup, D., Teh, Y.W., Eds.; International Convention Centre: Birmingham, UK, 2017; Volume 70, pp. 1587–1596.
16. Shen, T.; Lei, T.; Barzilay, R.; Jaakkola, T. Style transfer from non-parallel text by cross-alignment. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 6830–6841.
17. Fu, Z.; Tan, X.; Peng, N.; Zhao, D.; Yan, R. Style transfer in text: Exploration and evaluation. In Proceedings of the 32nd AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018.
18. Li, J.; Jia, R.; He, H.; Liang, P. Delete, Retrieve, Generate: A Simple Approach to Sentiment and Style Transfer. In Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), New Orleans, LA, USA, 1–6 June 2018; pp. 1865–1874. [CrossRef]
19. Sudhakar, A.; Upadhyay, B.; Maheswaran, A. "Transforming" Delete, Retrieve, Generate Approach for Controlled Text Style Transfer. In Proceedings of the Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Hong Kong, China, 7 November 2019; pp. 3269–3279. [CrossRef]
20. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.u.; Polosukhin, I. Attention is All you Need. In *Advances in Neural Information Processing Systems 30*; Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2017; pp. 5998–6008.
21. Wu, X.; Zhang, T.; Zang, L.; Han, J.; Hu, S. Mask and Infill: Applying Masked Language Model for Sentiment Transfer. In Proceedings of the 28th International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, 10–16 August 2019; Kraus, S., Ed.; pp. 5271–5277. [CrossRef]
22. Radford, A.; Narasimhan, K.; Salimans, T.; Sutskever, I. Improving Language Understanding by Generative Pre-Training. 2018. Available online: https://www.cs.ubc.ca/~amuham01/LING530/papers/radford2018improving.pdf (accessed on 8 May 2023).
23. Wang, Y.; Wu, Y.; Mou, L.; Li, Z.; Chao, W. Harnessing Pre-Trained Neural Networks with Rules for Formality Style Transfer. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, 3–7 November 2019; Inui, K., Jiang, J., Ng, V., Wan, X., Eds.; Association for Computational Linguistics: Cedarville, OH, USA, 2019; pp. 3571–3576. [CrossRef]
24. Chen, L.; Dai, S.; Tao, C.; Shen, D.; Gan, Z.; Zhang, H.; Zhang, Y.; Carin, L. Adversarial Text Generation via Feature-Mover's Distance. *arXiv* **2020**, arXiv:1809.06297.
25. Yin, D.; Huang, S.; Dai, X.; Chen, J. Utilizing Non-Parallel Text for Style Transfer by Making Partial Comparisons. In Proceedings of the International Joint Conference on Artificial Intelligence, Macao, China, 10–16 August 2019.
26. Luo, F.; Li, P.; Zhou, J.; Yang, P.; Chang, B.; Sun, X.; Sui, Z. A Dual Reinforcement Learning Framework for Unsupervised Text Style Transfer. In Proceedings of the 28th International Joint Conference on Artificial Intelligence, Macao, China, 10–16 August 2019; pp. 5116–5122. [CrossRef]
27. He, J.; Wang, X.; Neubig, G.; Berg-Kirkpatrick, T. A Probabilistic Formulation of Unsupervised Text Style Transfer. In Proceedings of the 8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, 26–30 April 2020.

28. Lee, J. Stable Style Transformer: Delete and Generate Approach with Encoder-Decoder for Text Style Transfer. In Proceedings of the 13th International Conference on Natural Language Generation, INLG 2020, Dublin, Ireland, 15–18 December 2020; pp. 195–204.
29. Krishna, K.; Wieting, J.; Iyyer, M. Reformulating Unsupervised Style Transfer as Paraphrase Generation. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), Online, 12 November 2020; pp. 737–762. [CrossRef]
30. Dale, D.; Voronov, A.; Dementieva, D.; Logacheva, V.; Kozlova, O.; Semenov, N.; Panchenko, A. Text Detoxification using Large Pre-trained Neural Models. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, Punta Cana, Dominican Republic, 7–11 November 2021; pp. 7979–7996. [CrossRef]
31. Krause, B.; Gotmare, A.D.; McCann, B.; Keskar, N.S.; Joty, S.; Socher, R.; Rajani, N.F. GeDi: Generative Discriminator Guided Sequence Generation. In Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2021, Punta Cana, Dominican Republic, 16–20 November 2021; pp. 4929–4952. [CrossRef]
32. Logacheva, V.; Dementieva, D.; Ustyantsev, S.; Moskovskiy, D.; Dale, D.; Krotova, I.; Semenov, N.; Panchenko, A. ParaDetox: Detoxification with Parallel Data. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Dublin, Ireland, 22–27 May 2022; pp. 6804–6818. [CrossRef]
33. Wieting, J.; Gimpel, K. ParaNMT-50M: Pushing the Limits of Paraphrastic Sentence Embeddings with Millions of Machine Translations. *arXiv* **2018**, arXiv:1711.05732.
34. Lewis, M.; Liu, Y.; Goyal, N.; Ghazvininejad, M.; Mohamed, A.; Levy, O.; Stoyanov, V.; Zettlemoyer, L. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020; pp. 7871–7880. [CrossRef]
35. Feng, S.; Wallace, E.; Grissom II, A.; Iyyer, M.; Rodriguez, P.; Boyd-Graber, J. Pathologies of Neural Models Make Interpretations Difficult. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, 31 October–4 November 2018; pp. 3719–3728.
36. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv* **2019**, arXiv:1907.11692.
37. Jigsaw. Toxic Comment Classification Challenge. 2018. Available online: https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge (accessed on 1 March 2021).
38. Jigsaw. Jigsaw Unintended Bias in Toxicity Classification. 2019. Available online: https://www.kaggle.com/c/jigsaw-unintended-bias-in-toxicity-classification (accessed on 1 March 2021).
39. Jigsaw. Jigsaw Multilingual Toxic Comment Classification. 2020. Available online: https://www.kaggle.com/c/jigsaw-multilingual-toxic-comment-classification (accessed on 1 March 2021).
40. Wiegand, M.; Ruppenhofer, J.; Schmidt, A.; Greenberg, C. Inducing a Lexicon of Abusive Words-–A Feature-Based Approach. In Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), New Orleans, LA, 1–6 June 2018; pp. 1046–1056. [CrossRef]
41. Plaza-Del-Arco, F.M.; Molina-González, M.D.; Ureña-López, L.A.; Martín-Valdivia, M.T. Detecting Misogyny and Xenophobia in Spanish Tweets Using Language Technologies. *ACM Trans. Internet Technol. TOIT* **2020**, *20*, 1–19.
42. Palomino, M.; Grad, D.; Bedwell, J. GoldenWind at SemEval-2021 Task 5: Orthrus—An Ensemble Approach to Identify Toxicity. In Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021), Online, 5–6 August 2021; pp. 860–864. [CrossRef]
43. Diaz, G. Stopwords-ISO/Stopwords-en: English Stopwords Collection.
44. Papineni, K.; Roukos, S.; Ward, T.; Zhu, W.J. Bleu: A Method for Automatic Evaluation of Machine Translation. In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, Philadelphia, PA, USA, 6–12 July 2002; pp. 311–318. [CrossRef]
45. Banerjee, S.; Lavie, A. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization, Ann Arbor, MI, USA, 29 June 2005; pp. 65–72.
46. Lin, C.Y. ROUGE: A Package for Automatic Evaluation of Summaries. In Proceedings of the Text Summarization Branches Out, Barcelona, Spain, 25–26 July 2004; pp. 74–81.

47. Wieting, J.; Berg-Kirkpatrick, T.; Gimpel, K.; Neubig, G. Beyond BLEU: Training Neural Machine Translation with Semantic Similarity. In Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, 28 July–2 August 2019; Volume 1: Long Papers; Korhonen, A., Traum, D.R., Màrquez, L., Eds.; Association for Computational Linguistics: Cedarville, OH, USA, 2019; pp. 4344–4355. [CrossRef]
48. Warstadt, A.; Singh, A.; Bowman, S.R. Neural Network Acceptability Judgments. *Trans. Assoc. Comput. Linguist.* **2019**, *7*, 625–641. [CrossRef]
49. Sennrich, R.; Haddow, B.; Birch, A. Neural Machine Translation of Rare Words with Subword Units. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, Volume 1: Long Papers, Berlin, Germany, 7–12 August 2016; pp. 1715–1725. [CrossRef]
50. Demšar, J. Statistical Comparisons of Classifiers over Multiple Data Sets. *J. Mach. Learn. Res.* **2006**, *7*, 1–30.