Information Fusion 99 (2023) 101887

Contents lists available at ScienceDirect

Information Fusion

journal homepage: www.elsevier.com/locate/inffus

Full length article

Enhancing social network hate detection using back translation and GPT-3 augmentations during training and test-time

Seffi Cohen [*], Dan Presil, Or Katz, Ofir Arbili, Shvat Messica, Lior Rokach

Department of Software and Information Systems Engineering, Ben Gurion University, Beer-Sheva, 8410501, Israel

ARTICLE INFO

Dataset link: https://github.com/OrKatz7/parler-hate-speech

Keywords:
Hate-detection
TTA
Back-translation
GPT

ABSTRACT

Social media platforms have become an essential means of communication, but they also serve as a breeding ground for hateful content. Detecting hate speech accurately is challenging due to factors such as slang and implicit hate speech. In response to these challenges, this paper presents a novel ensemble approach utilizing DeBERTa models, integrating back-translation and GPT-3 augmentation techniques during both training and test time. This method aims to address the complexities associated with detecting hate speech, resulting in more robust and accurate results. Our findings indicate that the proposed approach significantly enhances hate speech detection performance across various metrics and models in both the Parler and GAB datasets. For reproducibility and further exploration, our code is publicly available at https://github.com/OrKatz7/parler-hate-speech.

1. Introduction

The widespread use and adoption of social media platforms such as Twitter, Facebook, and Reddit have made them a primary means of communication for a substantial portion of the world population. While social media platforms can facilitate meaningful and productive conversations among users, they have also become a breeding ground for abusive language and hate-based activities. The anonymity and ease of access to these online platforms have contributed to their exploitation for spreading harmful language and the organization of violent acts. Due to calls for social media providers to screen comments before they are released to the public [1,2], mainstream platforms such as Twitter [3], Facebook [4,5], and Reddit [6] have implemented more stringent moderation measures. As a result, social media networks like Gab and Parler, which promote themselves as champions of "free speech", have experienced significant growth in their user base. These platforms attract a diverse group of users, including those who have been banned from mainstream platforms, conspiracy theorists, extremists, and other individuals who hold unorthodox views, as well as supporters of free speech. The proliferation of hate speech on these social media platforms often spills over into real-life actions, such as hate crimes. For instance, Gab was linked to a mass shooting at a synagogue in 2018 [7], and Parler was implicated in the January 6, 2021 insurrection attempt at the US Capitol building [8].

Despite the promising results achieved by state-of-the-art transformer-based models such as BERT [9] and GPT-3 [10] in text classification and hate speech detection, researchers have identified several challenges: One major challenge is the widespread use of slang in social media posts, which can alter the meaning of words and phrases and make it challenging for automated systems to identify hateful content accurately. The diversity of slang across regions, cultures, and demographics complicates the creation of a comprehensive slang database, while slang's ability to hide abusive messages makes it difficult for both automated systems and human moderators to detect hate speech.

Detecting implicit hate speech presents another challenge for statistical and neural text classifiers. Implicit hate speech refers to subtle or indirect language that aims to insult individuals or groups based on protected characteristics including race, gender, and cultural identity. The linguistic complexity and diversity of implicit hate speech encompass sarcastic and humorous expressions, euphemisms, circumlocution, and metaphorical language. Extremist groups have exploited this coded language to incite acts of aggression and domestic terrorism while retaining the ability to deny their involvement.

This study aims to improve the performance of hate speech detection by implementing a classification model based on DeBERTa [11] combined with back-translation (BT) augmentation and rephrasing synthetic augmentations using GPT-3 pre-train model, on training and inferencing. This approach leverages diverse strategies to address the challenges associated with detecting hate speech mentioned above by

[*] Corresponding author.
E-mail address: seffi@post.bgu.ac.il (S. Cohen).

https://doi.org/10.1016/j.inffus.2023.101887
Received 17 April 2023; Received in revised form 24 May 2023; Accepted 9 June 2023
Available online 13 June 2023

improving model robustness by using BT on training and inference performance using Test-Time-Augmentation (TTA).

We neutralize slang and reduce bias by using BT, which involves converting target language sentences into the source language and augmenting the training data by merging the sentences and back-translated sentences and rephrasing using GPT-3. This technique reduces the impact of cultural, historical, and other forms of bias that may be present in the original data and thus help to improve the accuracy of hate speech detection models to minimize over-fitting. We use EasyNMT[1] and a neural machine translation model for BT and OpenAI API[2] for rephrasing the sentences. We also leverage embedding techniques to select the translation languages that best retain the meaning of the original sentences. Our innovative approach also implements BT and GPT-3 rephrasing as augmentation techniques during test time. We generate predictions for the original and selected back-translated/rephrased sentences during the testing phase. The classification for a given sentence is obtained by combining the predictions for all sentences. We effectively reduce bias and enhance the model's ability to generalize across various linguistic styles and social media platforms by utilizing BT and GPT-3 augmentations at both the training and testing stages.

We assessed our methodology using two datasets. The Parler dataset [12] comprised 10,000 posts from the Parler platform with 3224 hateful posts, and the Gab hate dataset [13] consisted of 27,665 social media posts from the Gab platform, with 2563 hateful posts. Multiple experiments were conducted using three DeBERTa-based models with different configurations of BT and two pre-trained GPT-3 models (Ada and Babbage), using TTA, and utilizing an augmentation selection. We can summarize our main novelty and contributions as follows:

- **Novelty - (1) TTA** - No previous studies have applied TTA for hate speech detection. In this paper, we propose a novel approach that utilizes pre-trained language models DeBERTa, which benefits from pre-training on massive data and integrating BT and GPT-3 rephrasing techniques during training and test time. **(2) Training Augmentation Selection** - An augmentation selection technique for training was utilized to enhance performance by selecting appropriate languages to augment. **(3) TTA Selection** - An augmentation selection technique for test time was also utilized to enhance performance by selecting appropriate GPT-3 rephrased and back-translated sentences for each inferring instance.
- **Enhancing performance** - Our extensive empirical experiments show performance improvement, as demonstrated by a reduction in RMSE from 0.84 to 0.82 and an increase in AUC from 81.05 to 82.58. These results highlight our technique's potential for enhancing hate speech detection performance.
- **Code Availability and live demo** - Our reproducible code is publicly available on GitHub,[3] and live demos are available on Kaggle,[4] Colab,[5] and HuggingFace[6]

The remainder of this paper is organized as follows. Related work is given in Section 2. The proposed method is explained in detail in Section 3. Our experiments, datasets, and evaluation set are detailed in Section 4. Section 5 summarizes the results. Finally, Section 6 provides a conclusion for this paper and suggests possible future directions.

2. Related work

With the increase in online communication on social media platforms, the development of techniques for automatically identifying toxic and abusive speech has become a crucial issue for regulators, companies, and researchers [1,2,14]. Regulators of social media platforms are particularly focused on finding a balance between protecting the right to free speech and ensuring respect for the dignity of all users. While deep learning-based methods for detecting offensive content have shown promise, a major challenge is obtaining sufficient amounts of high-quality labeled data for training these models [15].

2.1. Hate speech detection

Addressing the issue of hate speech in online communication is a crucial but challenging endeavor that can be facilitated by utilizing Natural Language Processing (NLP) methods. In a comparative study of transformer-based architectures for hate speech detection, Kui [16] evaluated six prominent models: ALBERT, BERT, DeBERTa, mBERT, XLNet, and SqueezeBERT. Their findings highlighted DeBERTa as the superior model in terms of performance in English, underscoring its effectiveness for this task. Cao et al. [17] introduced HateGAN, a deep generative reinforcement learning model, to tackle the challenge of imbalanced classes in the data by augmenting it with hateful tweets. The authors conducted extensive experiments to enhance two widely used hate speech detection datasets by incorporating the tweets generated by HateGAN. The results of their experiments demonstrate that the use of HateGAN enhances the performance of hate speech detection, regardless of the specific classifiers and datasets used in the detection task. Unlike prior studies that primarily concentrate on the cross-dataset generalization abilities of hate speech classifiers. Chiu et al. [10] used GPT-3 to identify sexist and racist text passages with zero-, one-, and few-shot learning. The work of Ludwig et al. [18] focuses on the generalization abilities of hate speech classifiers to new targets within a single dataset. Their findings reveal that hate speech classifiers that are simply trained exhibit a target group-specific bias and that utilizing unsupervised domain adaptation can enhance the generalization abilities of models across various target groups of hate. Ahmed et al. [19] have tackled the bias issue in automated hate speech detection and have demonstrated a method for achieving fairer detection. Malik et al. [20] have shown that in hate speech detection, there is a large gap between the same-domain performance and the cross-domain performance. Pérez et al. [21] assessed the impact of contextual information in hate speech detection. Their work highlights the importance of considering the broader context in which a message is posted, a factor that our own method also takes into account. Utku et al. [22] proposed a novel approach to detecting hateful Twitter users using a graph convolutional network model. Their work demonstrates the potential of advanced machine learning models in identifying hate speech, even in complex and dynamic social media environments. Nagar et al. [23] focused on creating a more robust hate speech detection system by using social context and user data. Their approach aligns with our own emphasis on considering a wide range of data points in hate speech detection. Israeli et al. [12] introduced the first public hate speech dataset of posts from the social media website Parler. They provided baseline post-level classification results of various hate detection models: Jigsaw perspective [24], deHateBERT [25], Twitter-roBERTa [26], and HateBase [27]. One of the main challenges in hate speech detection is dealing with excessively noisy datasets and improving performance. Our method combines BT, GPT-3 rephrasing augmentations, and test-time augmentation to address the gap.

[1] https://github.com/UKPLab/EasyNMT.

[2] https://platform.openai.com/docs/api-reference.

[3] https://github.com/OrKatz7/parler-hate-speech.

[4] https://www.kaggle.com/code/orkatz2/parler-hate-speech-demo/.

[5] https://colab.research.google.com/github/OrKatz7/parler-hate-speech/blob/main/colab_demo.ipynb.

[6] https://huggingface.co/OrK7/parler_hate_speech.

2.2. Text data augmentation

Previous studies have shown that in the training process, words that are difficult to be predicted on the target language side tend to be more accurate after adding pseudo data [28,29]. Sharifirad et al. [30] used ConceptNet [31] to find and replace similar words and Wikidata knowledge graphs for word definitions to improve the classification of sexist tweets. Balkus et al. [32] suggested data augmentation using GPT-3 for improving text classification. Azam et al. [33] have explored Different data augmentation techniques for improving hate speech classification in Roman Urdu. Mao et al. [34,35] proposed metaphor processing methods focusing on pre-processing slang and complex language structures. However, these methods may struggle with the diversity and complexity of language on social media platforms. Another approach is self-training, as presented by He et al. [36], which uses the model's own predictions to generate additional training data. While this can be effective in some contexts, it may not handle the nuances of hate speech as robustly as back translation. Knowledge-based systems, like the one proposed by Li et al. [37], leverage external knowledge sources to enhance text understanding. Despite providing valuable context, they may not be as effective as back translation in handling diverse and complex language. In contrast to these methods, our work combines back translation, GPT-3 rephrasing, and Test time augmentation that introduces slight variations in sentences while preserving their original meaning, effectively increasing training data diversity in training and test time. We argue that our work is particularly suitable for hate speech detection due to its effectiveness in handling diverse and complex language used on social media platforms.

2.2.1. Back-translation

Back-translation (BT) is a widely utilized method for expanding data sets in natural language processing tasks. It involves using machine translation to translate monolingual target data into source language data automatically. The technique was first proposed by Sennrich et al. [38] as a training method for neural machine translation models.

Fadaee et al. [28] has shown that BT can be a useful approach for expanding data sets, which does not require any changes to the training process for language translation models. Additionally, the study found that incorporating synthetic data is beneficial for words with high prediction loss during training. Sugiyama et al. [39] have conducted experiments with BT on four language directions (English to French and Japanese, Japanese to English, and French to English) and have shown the advantages it has on context-aware machine translation. Yu et al. [40] have trained a machine reading comprehension model with augmented data that was generated by BT from French-to-English-to-French and English-to-German-to-English.

Utilizing language models pre-trained on a large text corpus and refined on a downstream task is the standard training technique for various natural language processing tasks. However, due to insufficient data to retrain the pre-trained language model, current pre-training approaches exhibit underfitting on the task distribution. Retraining the pre-trained language model with task-relevant back-translated data using an adaptive pre-training strategy has recently demonstrated notable performance gains [41]. Additionally, it was observed that pre-training with BT is more efficient for small datasets and robust to noises. Similarly, according to Shleifer et al. [42], BT significantly improves scenarios with limited resources. However, these improvements become negligible when the model has access to the entire dataset.

Jong et al. [43] have improved an automated essay scoring model by using BT of essays. The original writing essays were transformed with BT using two languages. French and Chinese BT transformations were used to add diversity to the data. Determining the corresponding target scores for the transformed essay text was necessary because it directly impacted the effectiveness of data augmentation.

Table 1
Five examples of BT.

Language	Text
Original text	Seattle BLM protesters demand white people *give up their homes*
BT German	Seattle BLM protesters call on white people to ***preserve*** *their houses*
BT French	BLM protesters in Seattle call on the White to *give up their homes*
BT Spanish	Seattle BLM protesters call for white people to *leave their homes*
BT Dutch	Seattle BLM protesters call for white people to *leave their homes*
BT Norwegian	Seattle BLM demonstrators call for white people to *give up their homes*

The widespread implementation of automated hate speech detection on social media networks has made it difficult to obtain examples of online hate speech, as the system automatically deletes such content. Using BT for dataset expansion could yield similar representations of the original text that are useful for hate speech detection. Mishra et al. [44] have used BT with English and French to expand their multi-lingual (English, German, and Hindu) hate speech detection model. For low-resource binary hate speech classification, Beddiar et al. [15] have expanded their training dataset by 20 times and have improved the classification score significantly with German and French BT and paraphrasing. BT was used in our work as a training and teat-time augmentation method. The translations were carried out using EasyNMT, a leading machine translation tool that supports over 100 languages. The original English texts are back-translated using the neural machine translation model m2m_100_418M [45–47], a 100-language multilingual encoder–decoder with 418 million parameters.

2.2.2. Test-time-augmentation

Test-time augmentation (TTA) is a well-established technique in computer vision that can significantly improve model performance. TTA generates multiple augmented copies of each test sample, and the results are combined with the original sample's prediction to produce a final output. The benefits of TTA are particularly apparent in image data, where it can produce multiple viewpoints of the same image, reducing errors and improving performance without requiring changes to the network's architecture [48]. The diversity of the augmented instances is crucial for error reduction, according to the independence principle of ensemble methods [49]. Recent studies have applied TTA to text classification [42], anomaly detection [50], and tabular data [51], showing impressive improvements in performance. However, there have been no studies that have investigated the application of TTA to hate speech detection. In this study, we applied TTA with GPT-3 rephrasing augmentation and BT augmentation selection to improve hate speech detection performance. Our approach builds on previous research by leveraging TTA's benefits to enhance hate speech detection.

3. Methods

The proposed method is based on two main components as illustrated in Fig. 1:

(1) The Training Component produces GPT-3 rephrasing augmentations and BT augmentations, selects the suitable language for BT for training and balances the training data to avoid an imbalanced model.

(2) The Test-Time Component aims to generate GPT-3 rephrasing augmentations and BT augmentations for test time, infer the augmentations, select the best subset of augmentations for each test instance, and combine them. A detailed description of each component is provided below.

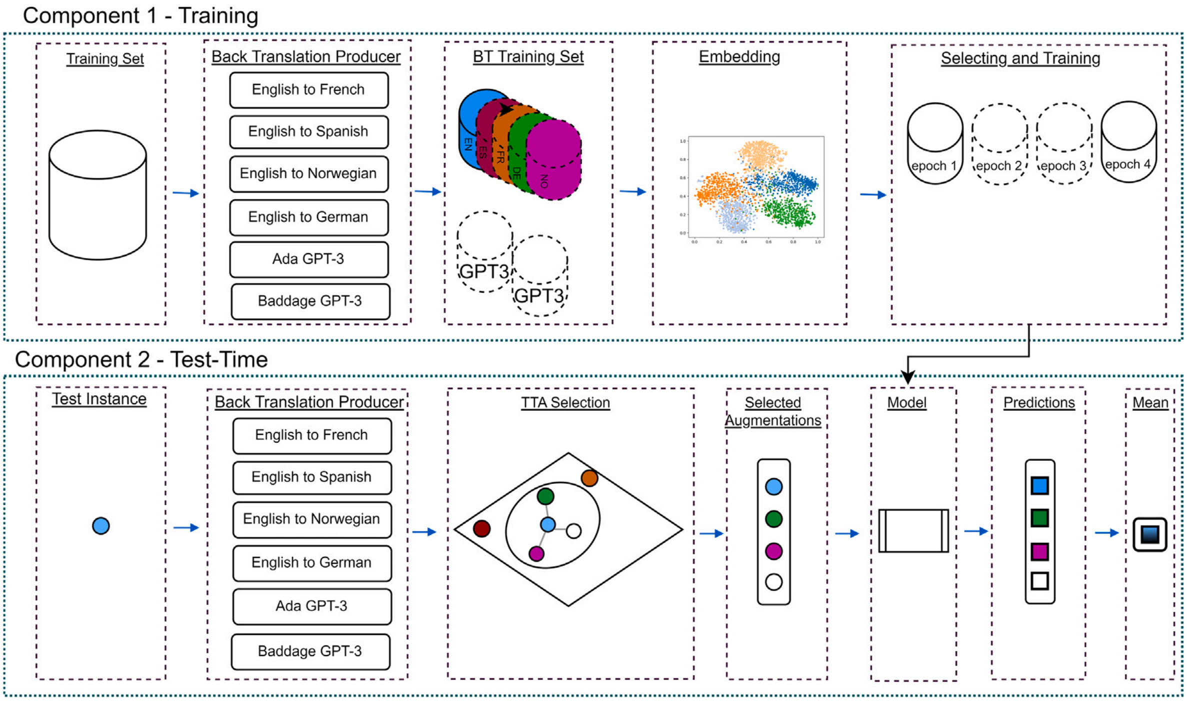

Fig. 1. An overview of our method. **Component 1** - The GPT-3 expands the training set by including rephrased sentences along with back-translated sentences. Then, based on the embedded distance, select the languages closest to the source language and exclude those far from the source language. We then trained the model using the original data in the initial epoch, the augmented data in subsequent epochs, and the original data in the final epoch. **Component 2** - Each instance in the test set is back-translated from various languages and rephrased using GPT-3. The closest augmented instances are selected based on their vector distance from the original instance, and the original and the selected augmentations are predicted. The average of all predictions is the final prediction.

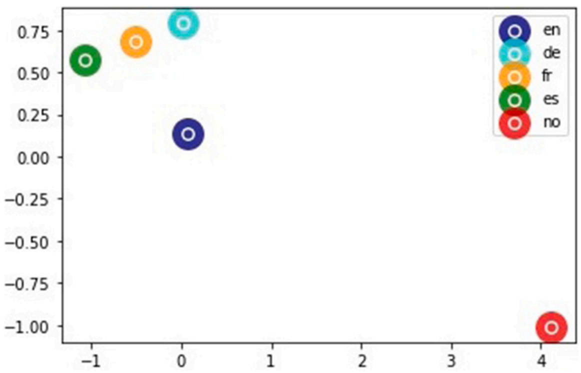

Fig. 2. A visual representation of the embedded languages using t-SNE. English (en), German (de), French (fr), Spanish (es), Norwegian (no).

3.1. The training component

The training component aims to optimize the use of GPT-3 and BT during training. There are two main weaknesses in the implementation of BT augmentation during training. The first challenge is poor translations that do not fit the original label, and the second challenge is overfitting the model with the augmented data.

Table 1 is an example of 4 successful and 1 unsuccessful augmentation of the original English text with BT. The BT output from French, Spanish, Dutch, and Norwegian preserves the meaning of the original sentence while the German BT reverses its meaning and turns the original hate-speech message into non-hate-speech. We utilize a BT selection and training sampler mechanism, described below, to enhance the quality of BT in training and address these challenges.

3.1.1. BT language selector

Choosing the optimal languages for BT involved selecting a set of initial languages and then selecting only those that produced the most beneficial augmented data that matched the original label.

The use of languages with close-embedded vectors is more suitable than the use of languages with far-embedded vectors [52]. Therefore, we selected languages closest to the source language and excluded those that are distant. This was achieved by translating the original sentences into a variety of languages. Then, we ran the augmented sentences through a DeBERTa-base pre-trained model and used t-SNE as a dimensionality reduction technique to find the distance between the original and the augmented sentences. Finally, the embeddings were used to determine which of the chosen language translations generated sentences that are most similar to the original sentences. The BT language selection method is described in algorithm 1.

After the languages have been embedded, we can observe in Fig. 2 that there are languages that are closer and farther from the source language.

3.1.2. BT training sampler

Incorporating BT into the initial epoch enhances model results by promoting the learning of diverse sentences. However, since transformer models are pre-trained on large volumes of text data, fine-tuning training can result in overfitting to the BT data. In order to take advantage of BT and avoid the risk of overfitting, the data sampler will begin training with the original data, then with BT data, and finally with only the original data for the last epoch.

3.2. Test-time augmentation selector

The Test-Time component aims to optimize the use of BT and GPT-3 augmentations during test time. Although TTA is applicable to any data type, its adoption in NLP is limited due to challenges in finding label-preserving transformations and the potential negative

Algorithm 1 BT Language Selector

```
Require: Original English sentences $S$, BT French sentences $S_{fr}$,
    BT Spanish sentences $S_{es}$, BT Dutch sentences $S_{nl}$, BT Norwegian
    sentences $S_{no}$, distance threshold $\delta = 0.7$
Ensure: Languages filtered by the Euclidean distance of their embedded sentences are less than or equal to $\delta$
    Convert $S \rightarrow v$, $S_{fr} \rightarrow v_{fr}$, $S_{es} \rightarrow v_{es}$, $S_{nl} \rightarrow v_{nl}$, $S_{no} \rightarrow v_{no}$ using
    DeBERTa-base embeddings
    Apply t-SNE to all vectors: $v_{tSNE} \leftarrow t\_SNE(v)$, $v_{tSNE_i} \leftarrow t\_SNE(v_i)$
    for $i \in \{fr, es, nl, no\}$
    for $i \in \{fr, es, nl, no\}$ do
        Compute Euclidean distance $d_i \leftarrow v_{tSNE} - v_{tSNE_i}$
    end for
    Initialize filtered set of Languages $F \leftarrow \emptyset$
    for $i \in \{fr, es, nl, no\}$ do
        if $d_i \leq \delta$ then
            Add $S_i$ to filtered set $F$
        end if
    end for
    return $F$
```

impact of poor translations, similar to the training component. During inference, in addition to the original sentence, we included all relevant back-translated and rephrased sentences, filtering out non-related augmentation using the embedding layer. During test-time, we evaluated which augmentation yielded the most accurate results. We used PCA as a dimensionality reduction technique to find the distances from the original sentence to the back-translated and rephrased sentences. This technique can determine which augmentations will be included in the TTA per row.

Algorithm 2 describes the Test Time Augmentation Selection method.

Algorithm 2 TTA Selector

```
Require: Original English sentence $S$, BT French sentence $S_{fr}$, Spanish
    sentence $S_{es}$, BT Dutch sentence $S_{nl}$, BT Norwegian sentence $S_{no}$,
    GPT-3 rephrased sentences $S_{gpt1}$, $S_{gpt2}$, distance threshold $\delta = 0.25$
Ensure: Augmentations filtered by the Euclidean distance of their
    embedded sentences are less than or equal to $\delta$
    Convert $S \rightarrow v$, $S_{fr} \rightarrow v_{fr}$, $S_{es} \rightarrow v_{es}$, $S_{nl} \rightarrow v_{nl}$, $S_{no} \rightarrow v_{no}$,
    $S_{gpt1} \rightarrow v_{gpt1}$, $S_{gpt2} \rightarrow v_{gpt2}$ using DeBERTa-base embeddings
    Apply PCA to all vectors: $v_{PCA} \leftarrow PCA(v)$, $v_{PCA_i} \leftarrow PCA(v_i)$ for
    $i \in \{fr, es, nl, no, gpt1, gpt2\}$
    for $i \in \{fr, es, nl, no, gpt1, gpt2\}$ do
        Compute Euclidean distance $d_i \leftarrow v_{PCA} - v_{PCA_i}$
    end for
    Initialize filtered set of augmentations $F \leftarrow \emptyset$
    for $i \in \{fr, es, nl, no, gpt1, gpt2\}$ do
        if $d_i \leq \delta$ then
            Add $S_i$ to filtered set $F$
        end if
    end for
    return $F$
```

This method aims to include for each tested instance the augmentations that yield the most accurate results. This approach enables us to determine which augmentations should be included in the TTA for each instance, ultimately enhancing the model's performance in detecting hate speech.

Fig. 3 provides a visual representation of the Test-Time Augmentation (TTA) selection process, specifically illustrating real data point distribution-based visualization. The figure displays the embedded sentences after PCA reduction, with 'En' representing the original sentence and other points signifying the augmentations. The selection behavior of TTA is guided by the distribution of these points, where the boundary around the original sentence serves as a filter for the augmentations. For instance, in "sample 1" the augmentations considered during the test-time were all the augmentations, except Norway BT and a rephrased sentence based on Ada's GPT-3. However, in Sample 4, all of the augmentations except the rephrased sentence based on Baddage's GPT-3 were very far from the original language, which could indicate problems with the input sentence, such as the use of multiple emojis or slang.

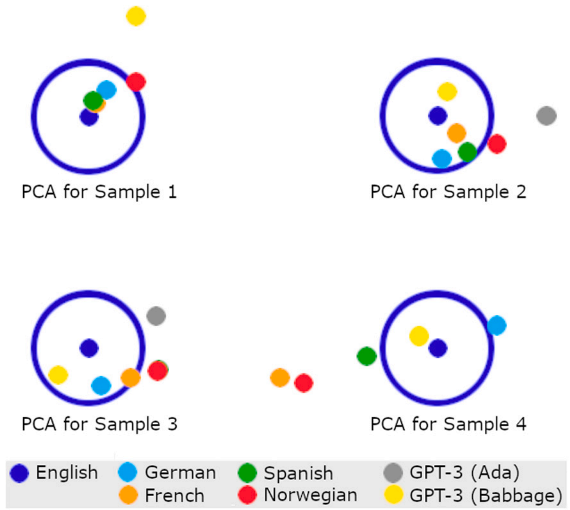

Fig. 3. Augmentation selection for TTA using embedding distance. The points represent the embedded sentence after PCA reduction. En represents the original sentence, while the other points represent the augmentation. Filtering the augmentation is based on the boundary around the original sentence.

4. Experiments

This section introduces the evaluation datasets, the experimental settings, and the evaluation set in our experiments. Our experiments were conducted on three different model architectures detailed in the experimental setup. Additionally, we examined all the components, including the training phase, the test phase, and the entire method, namely:

- small - pre-trained model based on DeBERTa-small architecture.
- base - pre-trained model based on DeBERTa-base architecture.
- large - pre-trained model based on DeBERTa-large architecture.
- Baseline - pre-trained model fine-tuned on the original data (without augmentation).
- BT - using BT augmentations.
- GPT - using GPT-3 rephrasing augmentations.
- TTA - utilizing Test Time Augmentation.

Additionally, to examine our method for regression and multi-class classification tasks, two datasets were evaluated: the Parler hate speech dataset [12], and the GAB Hate Corpus (GHC) [53] dataset [13].

4.1. Data

4.1.1. Parler

The Parler hate speech dataset is made up of 10K posts from the social media platform Parler. The posts were selected from the large unlabeled Parler dataset [54] to avoid an imbalanced dataset, using a pre-trained hate speech prediction model. Then, the selected posts were labeled by 112 annotators, and three annotators labeled each post. Posts' label scores range from 1 (no-hate) to 5 (extreme or explicit

Table 2
The Parler hate speech dataset label distribution.

Parler - Hate speech		
Hate score (4–5)	1,855	(18.3%)
Hate score (3–4)	2,213	(21.9%)
Hate score (2–3)	2,523	(24.9%)
Hate score (1–2)	3,530	(34.9%)
Total	10,121	

Table 3
The GAB Hate Corpus dataset label distribution.

GAB Hate Corpus [13]		
Hate	2,563	(9.26%)
Not-hate	25,102	(90.74%)
Total	27,665	
Top-level categories		
Human degradation	2,349	
Calls for violence	155	
Vulgar or offensive	1,748	
Targeted populations and framing		
Religious identity	480	
Racial/ethnic identity	629	
Sexual orientation	140	
Gender identity	156	
Ideology	235	
Nationality	209	
Political identity	335	
Mental/physical health	34	
Explicit rhetoric	1,201	
Implicit rhetoric	313	

hate). A post's average score is considered hateful if it exceeds three. The dataset label distribution is detailed in Table 2

4.1.2. GAB Hate Corpus

The GAB Hate Corpus (GHC) dataset contains 2563 hate posts out of 27,665 social media posts collected from gab.com and annotated to indicate the presence or absence of hate speech. According to Kennedy et al. [55], hate speech is defined as language that dehumanizes, undermines human dignity, belittles, incites violence, or promotes hateful ideologies such as white supremacy. Each instance in the GHC dataset was annotated by a minimum of three annotators chosen from a pool of 18. The number of annotations for each instance varies, and the total number of annotations for the entire dataset is 86,529. GHC is a multi-labeled dataset that includes labels for top-level categories, targeted populations, and framing as detailed in Table 3.

4.2. Experimental setup

These methods are applicable to any text-based dataset and pre-trained text classification model. We examine our method on DeBERTa-small, DeBERTa-base, and DeBERTa-large models.

The following subsections provide details on the specific setup for the models, the embedding for the augmentation selection, and the hyperparameters.

4.2.1. Neural network architecture and hyper-parameters tuning

We utilized DeBERTa-small, DeBERTa-base, and DeBERTa-large models, accompanied by a customized mean pooling layer to average the embedding model and an additional fully connected layer for prediction purposes. The DeBERTa models are trained with a cosine learning rate scheduler. Mean-Squared-Error was chosen as the loss function for regression and binary cross-entropy for classification. All experiments were run for 5 epochs with the cyclic cosine scheduler. The BT module was integrated during the initial two epochs. The embedding methods were tested using PCA, LDA, and t-SNE algorithms. Upon testing these three methods, it was determined that t-SNE yielded the most accurate results for the training phase, and PCA was the most accurate for the TTA and was therefore selected for the final configuration.

Table 4
Parler dataset regression results.

Configuration		Regression		
Model	Setting	MAE ↓	RMSE ↓	R^2 ↑
Small	Baseline	0.6503 ± 0.0153	0.8382 ± 0.0074	0.5201 ± 0.0562
	Baseline+TTA	0.6476 ± 0.0151	0.8314 ± 0.0074	0.5279 ± 0.0567
	BT	0.6374 ± 0.0167	0.8273 ± 0.0070	0.5326 ± 0.0515
	BT+TTA	0.6349 ± 0.0167	0.8209 ± 0.0073	0.5398 ± 0.0554
	GPT	0.6367 ± 0.0110	0.8220 ± 0.0069	0.5385 ± 0.0429
	GPT+TTA	0.6531 ± 0.0113	0.8363 ± 0.0078	0.5222 ± 0.0530
	BT+GPT	0.6344 ± 0.0146	0.8243 ± 0.0073	0.5325 ± 0.0471
	BT+GPT+TTA	**0.6341 ± 0.0160**	**0.8198 ± 0.0084**	**0.5409 ± 0.0597**
Base	Baseline	0.6414 ± 0.0143	0.8291 ± 0.0068	0.5306 ± 0.0553
	Baseline+TTA	0.6402 ± 0.0141	0.8246 ± 0.0061	0.5356 ± 0.0474
	BT	0.6312 ± 0.0164	0.8133 ± 0.0089	0.5482 ± 0.0562
	BT+TTA	0.6290 ± 0.0158	0.8086 ± 0.0086	0.5534 ± 0.0533
	GPT	0.6313 ± 0.0111	0.8165 ± 0.0057	0.5447 ± 0.0328
	GPT+TTA	0.6431 ± 0.0142	0.8263 ± 0.0075	0.5336 ± 0.0438
	BT+GPT	0.6285 ± 0.0162	0.8105 ± 0.0072	0.5513 ± 0.0415
	BT+GPT+TTA	**0.6283 ± 0.0144**	**0.8069 ± 0.0068**	**0.5552 ± 0.0389**
Large	Baseline	0.6187 ± 0.0171	0.7941 ± 0.0085	0.5694 ± 0.0721
	Baseline+TTA	0.6170 ± 0.0167	0.7903 ± 0.0083	0.5734 ± 0.0703
	BT	0.6109 ± 0.0142	0.7911 ± 0.0074	0.5726 ± 0.0511
	BT+TTA	**0.6097 ± 0.0135**	0.7864 ± 0.0070	0.5776 ± 0.0501
	GPT	0.6097 ± 0.0096	0.7909 ± 0.0054	0.5727 ± 0.0279
	GPT+TTA	0.6259 ± 0.0100	0.8043 ± 0.0063	0.5581 ± 0.0286
	BT+GPT	0.6137 ± 0.0138	0.7899 ± 0.0073	0.5737 ± 0.0525
	BT+GPT+TTA	0.6135 ± 0.0132	**0.7858 ± 0.0070**	**0.5782 ± 0.0518**

4.3. Evaluation set

In our study, we evaluated the performance of the compared classification methods using the AUC metric, which is commonly used for imbalance classification tasks. The regression methods were evaluated using R-Squared, Mean-Absolute-Error, and Root-Mean-Squared-Error.

A stratified 4-fold cross-validation method was performed to maintain a balanced distribution of label combinations across folds. In each fold, the GAB dataset was divided into 6916 posts, while the Parler dataset was divided into 4 folds of 2530 posts each fold for the same purposes.

5. Results

The results of our experiments on the evaluation datasets are summarized in this section, followed by an analysis of the results.

5.1. Parler dataset results

The results of our method with and without TTA compared to the baseline across the evaluated models are summarized in Table 4. The results showed that BT and GPT-3 augmentations enhanced the performance of all of the regression models across all evaluation metrics while incorporating the argumentations on training an on test time produced even better results.

5.2. GAB Hate Corpus results

The results of our method incorporating TTA, compared to the baseline across the evaluated models are summarized in Table 5. Similarly to the Parler results, our method significantly improved the performance of the classification. BT Augmentation incorporating TTA achieves the highest results on DeBRETa-small and DeBRETa-base models, while BT+GPT-3 Augmentation incorporating TTA achieves the best results on DeBRETa-large.

Table 5
GAB Hate Corpus results (AUC). Top-level categories: Violating human dignity (HD), calls for violence (CV), vulgarity and/or offensive language (VO). Targeted populations and framing: Religious identity (REL), racial/ethnic identity (RAE), sexual orientation (SXO), gender identity (GEN), ideology (IDL), nationality (NAT), political identity (POL), mental/physical health (MPH), explicit (EX) and implicit rhetoric (IM).

Configuration		Mean	Hate	Top-level			Targeted populations and framing									
Model	Setting			HD	CV	VO	REL	RAE	SXO	GEN	IDL	NAT	POL	MPH	EX	IM
Small	Baseline	80.16 ± 1.10	79.61	79.74	74.28	82.64	85.12	89.32	88.33	82.91	78.88	75.95	76.83	74.27	82.24	72.13
	Baseline+TTA	80.29 ± 1.11	79.72	79.84	74.29	82.73	85.57	89.72	88.54	82.94	79.03	76.08	77.02	73.96	82.28	72.30
	BT	81.56 ± 1.07	80.63	80.60	78.74	82.77	86.53	89.38	89.38	84.90	79.88	77.38	78.05	**77.66**	82.88	73.04
	BT+TTA	**81.73 ± 1.01**	**80.74**	**80.71**	**78.88**	**82.79**	87.22	89.62	**89.72**	85.08	80.17	**77.51**	78.31	77.30	**82.93**	**73.23**
	GPT	80.93 ± 1.15	80.10	80.16	76.54	82.13	86.95	89.00	88.34	84.62	80.10	75.44	78.63	75.83	82.43	72.70
	GPT+TTA	81.45 ± 1.17	80.38	80.45	77.00	82.34	87.87	89.53	89.05	**85.31**	**80.78**	76.21	**79.28**	76.44	82.85	72.81
	BT+GPT	81.20 ± 1.20	80.06	80.17	77.28	82.57	87.21	89.30	88.90	84.64	79.93	76.26	78.66	76.79	82.56	72.43
	BT+GPT+TTA	81.63 ± 1.20	80.27	80.38	77.59	82.78	**87.94**	**89.89**	89.30	**85.31**	80.57	76.88	79.23	77.38	82.90	72.46
Base	Baseline	80.63 ± 0.61	81.51	81.16	78.37	**83.42**	86.80	87.85	87.84	78.99	79.19	74.24	76.20	75.73	84.19	73.35
	Baseline+TTA	80.66 ± 0.63	81.52	81.16	78.34	83.41	87.03	88.14	87.96	78.99	79.06	74.44	76.21	75.41	**84.19**	73.37
	BT	81.62 ± 0.76	81.93	81.94	77.39	82.93	86.43	89.32	89.50	81.48	80.33	**76.95**	78.59	77.89	83.95	74.05
	BT+TTA	81.83 ± 0.76	**82.09**	**82.10**	77.70	82.95	86.80	89.56	90.02	81.82	80.40	77.14	78.97	77.83	84.01	**74.31**
	GPT	81.45 ± 1.01	80.10	80.16	76.54	82.13	86.95	89.00	88.34	**84.62**	80.10	75.44	78.63	75.83	82.43	72.70
	GPT+TTA	81.79 ± 1.02	81.56	81.47	79.16	83.29	87.69	89.15	89.44	82.33	**81.68**	75.91	79.75	76.24	83.84	73.48
	BT+GPT	82.01 ± 0.76	81.39	81.18	79.99	83.04	87.87	89.08	89.95	83.96	81.13	76.05	79.64	77.83	84.03	72.94
	BT+GPT+TTA	**82.33 ± 0.76**	81.62	81.39	**80.70**	83.13	**88.11**	**89.72**	**90.21**	84.48	81.42	76.34	**80.12**	**78.00**	84.16	73.23
Large	Baseline	81.05 ± 0.72	80.94	80.97	76.68	83.08	87.62	89.14	89.83	81.38	79.36	75.46	76.25	77.41	83.44	73.18
	Baseline+TTA	81.11 ± 0.66	80.96	81.01	76.71	83.03	88.00	89.38	90.10	81.32	79.37	75.35	76.37	77.21	83.42	73.26
	BT	81.23 ± 0.94	80.34	80.49	77.75	82.53	89.07	90.33	88.85	81.82	80.16	75.74	75.53	**78.71**	83.32	72.66
	BT+TTA	81.13 ± 0.87	80.11	80.29	77.62	82.33	**89.25**	**90.47**	88.94	81.74	80.08	75.52	75.39	78.41	83.06	72.59
	GPT	81.89 ± 0.82	81.09	81.05	79.29	83.17	88.45	88.91	88.97	84.20	81.02	77.25	78.29	77.86	82.98	73.95
	GPT+TTA	82.20 ± 0.79	81.18	81.09	79.91	83.23	88.67	89.44	89.41	84.58	81.52	77.68	78.66	78.17	83.04	**74.19**
	BT+GPT	82.31 ± 0.79	81.48	81.28	80.87	83.57	87.75	89.24	90.32	84.19	81.34	77.66	78.81	78.50	83.59	73.75
	BT+GPT+TTA	**82.58 ± 0.69**	**81.64**	**81.41**	**81.21**	**83.62**	88.27	89.60	**90.76**	**84.58**	**81.78**	**77.92**	**79.11**	78.57	**83.65**	74.04

5.3. Sensitivity analysis

We investigate the impact of the hyperparameter δ on the TTA Selection method's performance. In this method, δ represents the distance threshold for filtering augmentations. Our experiments evaluated the effects of different δ values between 0 and 0.4 with a step size of 0.01 using the Parler dataset and DeBERTa-small architecture.

5.3.1. TTA selection sensitivity analysis

Fig. 4 presents the RMSE results when the TTA Selection δ parameter ranges from 0.0 to 0.4 in increments of 0.01.

We find that optimal results are achieved when the δ parameter is set to 0.15.

In general, we observe that performance improves significantly as the distance from the original sentences increases. This is up to the 0.15 point where valuable augmentations are missed. Overall, we conclude that performance increases as the distance from the original sentences increases. This is until important augmentations are filtered out.

5.4. Result analysis and discussion

Our study demonstrates that BT and GPT-3 rephrasing augmentation in training and test-time enhances hate speech detection performance. Here we examine the impact of each component of our method and how it affects hate speech detection.

5.4.1. Quantitative analysis

For the Parler dataset, our method incorporating BT and GPT-3 augmentations during training and test time significantly enhanced the performance of all the regression models across all evaluation metrics. This is evident in Table 4, which summarizes the results of our method with and without Test-Time Augmentation (TTA) compared to the baseline across the evaluated models.

Similarly, for the GAB Hate Corpus, our method incorporating TTA significantly improved the performance of the classification models. BT Augmentation incorporating TTA achieves the highest results on DeBRETa-small and DeBRETa-base models, while BT+GPT-3 Augmentation incorporating TTA achieves the best results on DeBRETa-large. These results are summarized in Table 5.

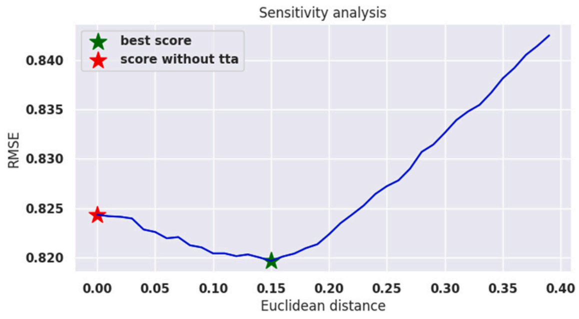

Fig. 4. TTA selection sensitivity analysis.

The Impact of BT Augmentation BT augmentation consistently improves training and test-time augmentation (TTA) results across all model architectures on both datasets. The improvements in the AUC scores for top-level categories, targeted populations, and framing are significant. For instance, in the DeBERTa-small model with BT, the mean of all categories' AUC scores improved from 80.16 in the baseline to 81.56. These improvements suggest that BT augmentation contributes to a more comprehensive understanding of the text, enabling the model to better identify hate speech across various categories and contexts.

The Impact of GPT-3 Augmentation The GPT-3 augmentation demonstrated a consistent improvement in training and TTA across all model architectures on the Parler dataset. However, the augmentation could slightly decrease performance in specific classes on the GAB dataset due to the augmentation changing labels between hate speech labels. This suggests that while GPT-3 augmentation generally enhances the model's performance, careful consideration is needed when applying it to different datasets to ensure label consistency.

The Impact of TTA TTA consistently achieves better results in the Parler dataset as well as the GAB dataset using all model architectures (DeBERTa-small, DeBERTa-base, DeBERTa-large) and augmentation sets (BT, GPT-3, BT+GPT-3). For example, in the DeBERTa-base model, TTA improved the mean of all categories' AUC scores from

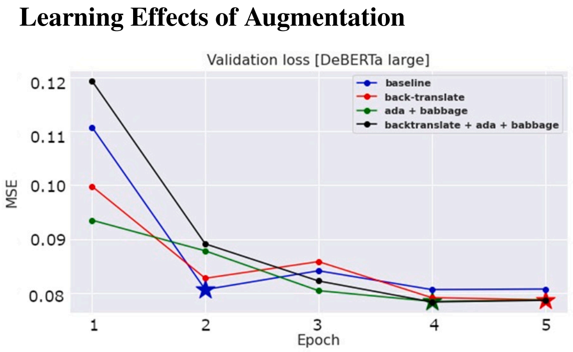

Fig. 5. Comparison of validation loss across the learning epochs.

80.63 in the baseline to 82.33 in the BT+GPT+TTA setting. This indicates that TTA can effectively enhance the model's performance by providing a more diverse set of data for the model to learn from during testing.

Learning Effects of Augmentation

Fig. 5 displays the effect of augmentation during training. With augmentations, the model converges in 4–5 epochs, whereas without it, it converges in 1–2 epochs on average, resulting in less overfitting. This graph illustrates the effect of augmentations on the learning score of the model. As a result of fine-tuning with augmented data, we demonstrate that the model can converge to later epochs and postpone over-fitting. The baseline model (without augmentations) converged after 2 epochs, while the BT-GPT (with augmentations) models converged in epochs 3 and 5 with a better validation score.

5.4.2. Qualitative analysis

To provide a qualitative analysis of our results, we examined several instances where our method correctly identified hate speech that was missed by the baseline models. We found that our method was particularly effective at identifying implicit hate speech and slang, which are often challenging for traditional models to detect. This can be attributed to the use of back-translation and GPT-3 augmentations, which allowed our models to better understand the nuances of language used in social media platforms.

Furthermore, we observed that our method was robust against common issues in social media text analysis such as misspellings and abbreviations. This is likely due to the robustness of the GPT-3 model, which has been trained on a diverse range of internet text and is thus capable of understanding these variations in language.

5.4.3. Discussion

Our results can be attributed to several factors related to the unique features of our method: The improvement in our results can be attributed to several factors related to the unique features of our method:

Back-Translation Augmentation: Back-translation Augmentation involves translating a sentence into a different language and then translating it back into the original language. This process can introduce slight variations in the sentence while preserving its original meaning, effectively increasing the diversity of our training data. This increased diversity likely helped our models to better generalize and perform more accurately on unseen data.

GPT-3 Augmentation: GPT-3 is a state-of-the-art language model that has been trained on a diverse range of internet text. By using GPT-3 to augment our training data, we were able to introduce additional variations and complexities that are representative of real-world language use. This likely helped our models to better understand the nuances of language used in social media platforms, including slang and implicit hate speech, thereby improving their performance.

Test-Time Augmentation (TTA): TTA involves creating multiple augmented versions of the test data and averaging the predictions of the model on these versions. This technique can help to increase the robustness of the model's predictions and reduce the impact of random variations in the test data. The use of TTA in our method likely contributed to the observed improvements in our results.

6. Conclusions

In this paper, we have demonstrated the benefits of applying BT and rephrasing sentences using GPT-3 augmentation at both training and test time to considerably improve the performance of hate speech detection models that were trained on relatively small labeled textual datasets.

As part of our research, we have also introduced two methods of augmentation selection: (1) BT Language Selector - selects relevant BT languages for training, based on the embedding layer of a pre-trained model; and (2) TTA Selector - selects appropriate augmentations for test time augmentation.

Our training approach has demonstrated that smaller language models can perform better than larger ones. One of the major limitations of our method is the difficulty of improving the performance of large models using a small amount of data. One way to address this issue is to produce more diverse augmentations and prevent overfitting. Future work should explore more augmentation techniques, such as the use of Large Language models to rephrase sentences in diverse dialects and styles.

Code availability and live demo

Our reproducible code is publicly available on:

- https://github.com/OrKatz7/parler-hate-speech

Our live demos are publicly available on:

- https://www.kaggle.com/code/orkatz2/parler-hate-speech-demo ,
- https://colab.research.google.com/github/OrKatz7/parler-hate-speech/blob/main/colab_demo.ipynb, and
- https://huggingface.co/OrK7/parler_hate_speech

CRediT authorship contribution statement

Seffi Cohen: Conceptualization, Investigation, Methodology, Formal analysis, Writing – original draft, Writing – review & editing. **Dan Presil:** Conceptualization, Software, Visualization, Data curation, Writing – original draft. **Or Katz:** Conceptualization, Software, Visualization, Data curation, Writing – original draft. **Ofir Arbili:** Software, Visualization, Data curation, Writing – original draft. **Shvat Messica:** Software, Visualization, Data curation, Writing – original draft. **Lior Rokach:** Supervision, Writing – review & editing.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Seffi Cohen reports equipment, drugs, or supplies and writing assistance were provided by Ben-Gurion University of the Negev.

Data availability

The data and reproducible code are publicly available at https://github.com/OrKatz7/parler-hate-speech.

References

[1] P. Fortuna, S. Nunes, A survey on automatic detection of hate speech in text, ACM Comput. Surv. 51 (2018) 1–30.

[2] A. Guterres, United Nations Strategy and Plan of Action on Hate Speech, Tech. Rep, United Nations, New York, NY, USA, 2019.

[3] P. Singh, How SPRINKLR Helps Identify and Measure Toxic Content with AI, Sprinklr, 2023.

[4] S. Shead, Facebook Claims A.I. Now Detects 94.7% of the Hate Speech that Gets Removed from its Platform, CNBC, 2020.

[5] M. Schroepfer, Update on Our Progress on AI and Hate Speech Detection, Meta, 2021.

[6] Reddit, Understanding Hate on Reddit, and the Impact of Our New Policy, Reddit Security, 2020.

[7] L. Hensley, Right-Wing Platform Gab Taken Down After Pittsburgh Shooting, Says It's been 'Smeared' by Media- National, Oct 2018, GlobalNews, Canada, 2018.

[8] A. Bajak, J. Guynn, M. Thorson, When Trump Started his Speech Before the Capitol Riot, Talk on Parler Turned to Civil War, Feb 2021, Usatoday, 2021.

[9] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, 2018, arXiv preprint arXiv:1810.04805.

[10] K.-L. Chiu, A. Collins, R. Alexander, Detecting hate speech with gpt-3, 2021, arXiv preprint arXiv:2103.12407.

[11] P. He, X. Liu, J. Gao, W. Chen, Deberta: Decoding-enhanced bert with disentangled attention, 2020, arXiv preprint arXiv:2006.03654.

[12] A. Israeli, O. Tsur, Free speech or free hate speech? Analyzing the proliferation of hate speech in Parler, in: Proceedings of the Sixth Workshop on Online Abuse and Harms, WOAH, 2022, pp. 109–121.

[13] B. Kennedy, M. Atari, A.M. Davani, L. Yeh, A. Omrani, Y. Kim, K. Coombs, S. Havaldar, G. Portillo-Wightman, E. Gonzalez, et al., The Gab Hate Corpus: A Collection of 27k Posts Annotated for Hate Speech, PsyArXiv, 2018, July 18.

[14] N. Lomas, Facebook, Google, Twitter Commit to Hate Speech Action in Germany, TechCrunch, 2015.

[15] D.R. Beddiar, M.S. Jahan, M. Oussalah, Data expansion using back translation and paraphrasing for hate speech detection, Online Soc. Netw. Media 24 (2021) 100153.

[16] Y. Kui, Detect hate and offensive content in english and indo-aryan languages based on transformer, in: Forum for Information Retrieval Evaluation (Working Notes)(FIRE), CEUR-WS. org, 2021.

[17] R. Cao, R.K.-W. Lee, Hategan: Adversarial generative-based data augmentation for hate speech detection, in: Proceedings of the 28th International Conference on Computational Linguistics, 2020, pp. 6327–6338.

[18] F. Ludwig, K. Dolos, T. Zesch, E. Hobley, Improving generalization of hate speech detection systems to novel target groups via domain adaptation, in: Proceedings of the Sixth Workshop on Online Abuse and Harms, WOAH, 2022, pp. 29–39.

[19] Z. Ahmed, B. Vidgen, S.A. Hale, Tackling racial bias in automated online hate detection: Towards fair and accurate detection of hateful users with geometric deep learning, EPJ Data Sci. 11 (1) (2022) 8.

[20] J.S. Malik, G. Pang, A.v.d. Hengel, Deep learning for hate speech detection: A comparative study, 2022, arXiv preprint arXiv:2202.09517.

[21] J.M. Pérez, F.M. Luque, D. Zayat, M. Kondratzky, A. Moro, P.S. Serrati, J. Zajac, P. Miguel, N. Debandi, A. Gravano, et al., Assessing the impact of contextual information in hate speech detection, IEEE Access 11 (2023) 30575–30590.

[22] A. Utku, U. Can, S. Aslan, Detection of hateful twitter users with graph convolutional network model, Earth Sci. Inform. (2023) 1–15.

[23] S. Nagar, F.A. Barbhuiya, K. Dey, Towards more robust hate speech detection: using social context and user data, Soc. Netw. Anal. Min. 13 (1) (2023) 47.

[24] H. Hosseini, S. Kannan, B. Zhang, R. Poovendran, Deceiving google's perspective api built for detecting toxic comments, 2017, arXiv preprint arXiv:1702.08138.

[25] S.S. Aluru, B. Mathew, P. Saha, A. Mukherjee, Deep learning models for multilingual hate speech detection, 2020, arXiv preprint arXiv:2004.06465.

[26] F. Barbieri, J. Camacho-Collados, L. Neves, L. Espinosa-Anke, Tweeteval: Unified benchmark and comparative evaluation for tweet classification, 2020, arXiv preprint arXiv:2010.12421.

[27] T. Davidson, D. Warmsley, M. Macy, I. Weber, Automated hate speech detection and the problem of offensive language, in: Proceedings of the International AAAI Conference on Web and Social Media, Vol. 11, 2017, pp. 512–515.

[28] M. Fadaee, C. Monz, Back-translation sampling by targeting difficult words in neural machine translation, 2018, arXiv preprint arXiv:1808.09006.

[29] N. Xu, Y. Li, C. Xu, Y. Li, B. Li, T. Xiao, J. Zhu, Analysis of back-translation methods for low-resource neural machine translation, in: CCF International Conference on Natural Language Processing and Chinese Computing, Springer, 2019, pp. 466–475.

[30] S. Sharifirad, B. Jafarpour, S. Matwin, Boosting text classification performance on sexist tweets by text augmentation and text generation using a combination of knowledge graphs, in: Proceedings of the 2nd Workshop on Abusive Language Online (ALW2), 2018, pp. 107–114.

[31] R. Speer, J. Chin, C. Havasi, Conceptnet 5.5: An open multilingual graph of general knowledge, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 31, 2017.

[32] S. Balkus, D. Yan, Improving short text classification with augmented data using GPT-3, 2022, arXiv preprint arXiv:2205.10981.

[33] U. Azam, H. Rizwan, A. Karim, Exploring data augmentation strategies for hate speech detection in roman urdu, in: Proceedings of the Thirteenth Language Resources and Evaluation Conference, 2022, pp. 4523–4531.

[34] R. Mao, C. Lin, F. Guerin, Word embedding and WordNet based metaphor identification and interpretation, in: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics (ACL), 2018.

[35] R. Mao, X. Li, M. Ge, E. Cambria, MetaPro: A computational metaphor processing model for text pre-processing, Inf. Fusion 86 (2022) 30–43.

[36] K. He, R. Mao, T. Gong, C. Li, E. Cambria, Meta-based self-training and re-weighting for aspect-based sentiment analysis, IEEE Trans. Affect. Comput. (2022).

[37] W. Li, L. Zhu, R. Mao, E. Cambria, SKIER: A symbolic knowledge integrated model for conversational emotion recognition, 2023.

[38] R. Sennrich, B. Haddow, A. Birch, Improving neural machine translation models with monolingual data, 2015, arXiv preprint arXiv:1511.06709.

[39] A. Sugiyama, N. Yoshinaga, Data augmentation using back-translation for context-aware neural machine translation, in: Proceedings of the Fourth Workshop on Discourse in Machine Translation (DiscoMT 2019), 2019, pp. 35–44.

[40] A.W. Yu, D. Dohan, M.-T. Luong, R. Zhao, K. Chen, M. Norouzi, Q.V. Le, Qanet: Combining local convolution with global self-attention for reading comprehension, 2018, arXiv preprint arXiv:1804.09541.

[41] J. Lee, J. Kim, P. Kang, Back-translated task adaptive pretraining: Improving accuracy and robustness on text classification, 2021, arXiv preprint arXiv:2107.10474.

[42] S. Shleifer, Low resource text classification with ulmfit and backtranslation, 2019, arXiv preprint arXiv:1903.09244.

[43] Y.-J. Jong, Y.-J. Kim, O.-C. Ri, Improving performance of automated essay scoring by using back-translation essays and adjusted scores, Math. Probl. Eng. 2022 (2022).

[44] S. Mishra, S. Prasad, S. Mishra, Exploring multi-task multi-lingual learning of transformer models for hate speech and offensive speech identification in social media, SN Comput. Sci. 2 (2021) 1–19.

[45] A. Fan, S. Bhosale, H. Schwenk, Z. Ma, A. El-Kishky, S. Goyal, M. Baines, O. Celebi, G. Wenzek, V. Chaudhary, N. Goyal, T. Birch, V. Liptchinsky, S. Edunov, E. Grave, M. Auli, A. Joulin, Beyond english-centric multilingual machine translation, 2020, arXiv preprint.

[46] H. Schwenk, G. Wenzek, S. Edunov, E. Grave, A. Joulin, Ccmatrix: Mining billions of high-quality parallel sentences on the web, 2019, arXiv preprint arXiv:1911.04944.

[47] A. El-Kishky, V. Chaudhary, F. Guzman, P. Koehn, A massive collection of cross-lingual web-document pairs, 2019, arXiv preprint arXiv:1911.06154.

[48] L. Perez, J. Wang, The effectiveness of data augmentation in image classification using deep learning, 2017, arXiv preprint arXiv:1712.04621.

[49] L. Rokach, Ensemble-based classifiers, Artif. Intell. Rev. 33 (1–2) (2010) 1–39.

[50] S. Cohen, N. Goldshlager, L. Rokach, B. Shapira, Boosting anomaly detection using unsupervised diverse test-time augmentation, Inform. Sci. (2023).

[51] S. Cohen, N. Dagan, N. Cohen-Inger, D. Ofer, L. Rokach, ICU survival prediction incorporating test-time augmentation to improve the accuracy of ensemble-based models, IEEE Access 9 (2021) 91584–91592.

[52] J. Wang, Y. Dong, Measurement of text similarity: a survey, Information 11 (9) (2020) 421.

[53] B. Kennedy, M. Atari, A.M. Davani, L. Yeh, A. Omrani, Y. Kim, K. Coombs, G. Portillo-Wightman, S. Havaldar, E. Gonzalez, et al., The Gab Hate Corpus, 2022, http://dx.doi.org/10.17605/OSF.IO/EDUA3, URL osf.io/edua3.

[54] M. Aliapoulios, E. Bevensee, J. Blackburn, B. Bradlyn, E. De Cristofaro, G. Stringhini, S. Zannettou, An early look at the parler online social network, 2021, arXiv preprint arXiv:2101.03820.

[55] B. Kennedy, M. Atari, A.M. Davani, L. Yeh, A. Omrani, Y. Kim, K. Coombs, S. Havaldar, G. Portillo-Wightman, E. Gonzalez, et al., Introducing the Gab Hate Corpus: defining and applying hate-based rhetoric to social media posts at scale, Lang. Resour. Eval. (2022) 1–30.