Lisbora Likaj

Analysis of the Hate Speech Detection on Twitter

Seminar Report

submitted to

Graz University of Technology

Supervisor Roman Kern

Institute of Interactive Systems and Data Science
Head: Univ. Prof. Dipl.-Inf. Dr. Stefanie Lindstaedt

Graz, February 2023

Abstract

The detection of hate speech in online content has become an issue of increasing importance in recent years. Text classification for online content is challenging due to the complexity of natural language and the sheer number of posts being published, which often include many mistakes and abbreviations and are lacking context. In this report, we investigate hate speech detection as a text classification problem. Our analysis is focused on hate speech detection by utilizing Twitter sentiments and a machine learning model to classify tweets into hateful or not hateful. We make a comparative analysis of the results between state-of-the-art approaches by analysing the hate speech content before and after a particular event on time e.g Covid-19 Lockdown.

Contents

List of Figures

1 Introduction

Twitter advocates a set of rules to prevent hateful content online (Twitter). Even with specific guidelines on how one should behave on the site, hate speech is still a problem on Twitter (Wagner) and the majority of social media platforms (Laub). In order to avoid further harm to other users, hate speech should be continuously monitored, detected and removed from the site. With over 6000 tweets being published every second, this is a challenging mission (Sayce). To omit the need to manually read through every tweet and classify them as hateful or not, recent research has been leveraging machine learning techniques to automatically detect hate speech online (Badjatiya et al., 1 Jun 2017; Cao et al., October 2020; Davidson et al., 2017; Fortuna and Nunes, 2018; Mutangai et al., No. 9, 2020; Ziqi, October 2018).

In this report, we analyse automatic detection of publicly hateful content in Twitter (Kamble and Joshi, 2018). In section 1, we introduce and motivate our ideas behind hate speech detection. In section 2, we investigate different state-of-the-art models and their properties. In section 3, we look at available datasets for hate speech detection and propose different use case scenarios. In section 4, we briefly summarise our approach and experimental setup. In section 5, we look at the analysis of collected tweets. We showcase some experimental outcomes, which are then followed by a discussion and some concluding remarks.

1.1 Research Tasks and Hate Speech Terminology

In this report, we use existing state-of-the-art machine learning approach to detect hate speech on social media. Specifically, we use LSTM and Bi-LSTM as a mean to classify hateful content, considering it as a binary classification task. Furthermore, we provide insights on tweets related to ”Donald Trump” controversially. Our aim is to classify tweets into two categories: *hate* speech, and *non-hate*. We define *hate* as 1) “language that is used to expresses hatred towards a targeted group or is intended to be derogatory, to humiliate, or to insult the members of the group” (Davidson et al., 2017) and 2) spread hatred, or cause harm to others, although it may not contain offensive or abusive language. Moreover, by leveraging a pre-

trained machine learning model named "Detoxify" [1], we perform experiments on data collected from Twitter and compare the classification results with the prediction of our models. We offer further insights into the following:

- What makes detecting hate speech on Twitter difficult for machine learning models?
- How to properly collect data from Twitter?
- How did Covid-19 pandemic affect hate speech on Twitter, particularly towards Trump?

Why Donald Trump?

Whether or not you support Donald Trump, there is a huge debate going on on social media regarding his controversial decisions and his governmental acts over the past few years. We focus on the users' behaviour before and after Covid-19 pandemic. Hence, we gather tweets related to Trump before and after lockdown and vaccination enforcement in USA during 2020 (Chang; Ellis).

[1] Detoxify: Toxic Comment Classification with Pytorch Lightning and Transformers: `https://github.com/unitaryai/detoxify`

2 Related Work

In this section, we shed light on some state of the art classification approaches focusing on hate speech detection. Fortuna et al. categorise text mining models into two groups: general models and complementary models (Fortuna and Nunes, 2018). The general model focus on word frequency, bag-of-words and other straight-forward approaches, while complementary models combine classification techniques with an extra effort in pre-processing and feature engineering.

There are plenty of reasearch in mining textual information such as linguistic pre-processing, counting frequencies/occurrences of tokens, analysis of content e.g PCA or LDA, deep learning techniques e.g word embedding, transformation etc. In our context, word embeddings refer to representation of the textual facts as numrerical vectors. The idea is that similar words with similar meaning or given in same context receive similar vectorial representation. We can mention here a great deal of works paragraph2vec, fast text, GloVE, word2vec etc. On the other hand, transformation approaches such as sntiment and topic analysis, N-grams, TF-IDF models are based on the distance between words or dissimilarity degree, considering the minimum number of edits necessary to transform one string into another. Additionally, feature engineering relies on the characteristics or specific features of sequence, like number of words, syllables, characters, etc (Fortuna and Nunes, 2018; Schmidt and Wiegand, August 27, 2021; Xiang et al., 2012; Davidson et al., 2017; Cao et al., October 2020; Zampieri et al., 2019; Badjatiya et al., 1 Jun 2017; Mutangai et al., No. 9, 2020).

Deep learning models offer remarkable performance, by making use of word embeddings, while excavating the benefits of neural network. Such networks as LSTM, Bi-LSTM, CNN-attention have achieved high accuracy when dealing with long sequences. Recently, Transformers architecture gained a lot of popularity due to its attention mechanism. The attention is a feature which enables the network to focus on important aspects of the sequence. This mechanism applies different weights to the words depending on their context and the contribution to the sequence meaning. Among the above mentioned categories of methods (Fortuna and Nunes, 2018; Schmidt and Wiegand, August 27, 2021; Xiang et al., 2012; Davidson et al., 2017; Cao et al., October 2020; Zampieri et al., 2019; Badjatiya et al., 1 Jun 2017; Mutangai et al., No. 9, 2020).

TF-IDF model (Davidson et al., 2017), DeepHate, AngryBERT (Cao et al., October

2020), LSTM, or Bi-LSTM (Badjatiya et al., 1 Jun 2017; Bhalla; Bisht et al., 2018; Sarrace et al., 2018) are a few supervised models that apply a variety of feature extraction techniques, besides different text analysis methods. These models utilise several natural language pre-processing methods on tweets (Cao et al., October 2020; Badjatiya et al., 1 Jun 2017) in order to better classify textual content.

When labelling datasets, tweets written by African Americans are more likely to be labelled as offensive (Sap et al., 2019). This is partially due to words, which are offensive when used by someone not part of the community, being reclaimed by these communities. This leads to the voices of an already discriminated group, being subdued even further. To reduce racial bias, Sap et al. (Sap et al., 2019) showed that dialect and race priming helps, i.e. asking the labelers to consider whether a tweet is offensive to *anyone* and to consider the possible racial background of the author.

Most of the existing research focus on introducing new techniques or publishing new datasets related to hate speech detection. In contrary to existing works, in our report we do not propose a new deep learning model. We make use of LSTM and BiLSTM to analyse hate speech on recent tweets. Our goal is to see how different events affect use behaviour online.

3 Hate Speech Datasets and Twitter Crawling Process

3.1 Hate Speech Datasets

During our research, we observed a lack in a common available benchmark to compare different models. We see this as a major limitation when comparing different SOTA. Existing models make use of data that is collected in a private way (Schmidt and Wiegand, August 27, 2021). Waseem and Hovy (Waseem and Hovy, June 17, 2016) created a dataset of $16,914$ tweets, which is split into *sexist* content, *racist* content and *neither sexist or racist*. In addition, Kumar et al. (Kumar and Pranesh, August 27, 2021) have recently annotated a small dataset that is used to detect hate speech related to the Black Lives Matter movement. The dataset contains $3,084$ tweets labelled as *hate* and the remaining $6,081$ as *non-hate* tweets. Davidson et al. published another dataset which contains 24,783 tweets that are manually annotated as *hate*, *offensive* or *neither hate nor offensive* (Davidson et al., 2017).

In this report, we utilize a dataset found in Kaggle (Dataset). Tweets are classified as *racist/sexist* or *neither racist nor sexist*. Unfortunately, the dataset is heavily skewed with 29,695 tweets (93%) being classified as *neither racist nor sexist* and only 2,240 tweets (7%) are classified as *racist/sexist*. This heavily affect our model predictive ability and might induce perplexity when trying to classify hate speech content.

3.2 Data Collection Pipeline

Our data collection relies on two python packages, namely *tweepy* and *twarc*. Tweepy is used to crawl data from the Twitter API and to collect tweets based on a set of keywords. For a detailed insight into the keywords used, readers are encouraged to have a look at the source code. As Twitter constrains the collected data usage, most online datasets only store tweet-ids, resulting in the need to convert them into usable text. To remedy this issue, the *twarc* package can be used to hydrate tweets and to collect the respective text of each tweet based on the id.

Our collection approach is based on two sets of keywords. We have one set of keywords with topic related tweets e.g *Trump, election, USA president, Covid-19, vaccination* etc. The second set of tweets contains offensive words and words which are commonly associated with hate. For instance, in the context of "Trump" a word like "*misogyny*" may be considered hate, although the word itself is neither hateful nor offensive. We join the two sets using *Tweepy* and query the Twitter API based on these combined words, i.e. a tweet should contain at least one word from each set. We collected a total of **4611** tweets related to Trump and Covid pandemic, from which around 1826 tweets were collected before Covid-19 strike (i.e before 2020).

4 Supervised Approach on Detecting Hate Speech as a Binary Classification Problem

In this section, we introduce our method for hate speech detection. We describe a few pre-processing technique, and then dive into a pre-trained approach for detecting toxic content. We train LSTM and BiLSTM models using hate content crawled from Twitter and compare their accuracy with a pre-trained method named "Detoxify". We analyse which approach performs better in case of hate speech detection on Twitter.

4.1 Models Architecture

In our paper we make use of **LSTM** and **BiLSTM** to detect hate speech on our collected tweets. We use the Kaggle Dataset to train and fine-tune our models.

The Long Short Term Memory model, is inspired by different implementations Badjatiya et al. (1 Jun 2017); Bhalla; Bisht et al. (2018); Sarrace et al. (2018). Recurrent neural networks like LSTMs employ their internal memory to handle arbitrarily input sequences. Hence, we use LSTMs to capture long range dependencies in tweets, as we believe that they play a key role in hate speech detection. The model utilises an embedding layer which maps each token in the sequence into real vector representations while preserving semantic information (see Table 5.2) Mikolov et al. (2017). In addition, a single LSTM layer with 128 hidden units is used. Dropout is added to avoid overfitting. In both implementations, we use a dense layer whose number of units depend on the number of classes to be predicted.

Furthermore, we experiment with the bidirectional LSTM (BiLSTM). The idea is to traverse the input sequence twice *(i.e one from the beginning of the sequence till the end and vice versa).* Accordingly, this yields additional training parameters. Consequently, this results in the BiLSTM model generalising better for small datasets Ezen-Can (Sep 11, 2020).

4.2 Text Pre-Processing

Tweets often contain many different tenses, grammatical errors, unknown symbols, hashtags, and non-English characters. To tackle this problem, we employ different natural language processing techniques. Given a tweet, we use the following normalisation procedure:

1. Clean the text (e.g remove special characters, links, emojis etc.)
2. Lowercase all words
3. Remove stop-words and punctuation
4. Replace mentions (i.e @user) with default values
5. Normalize hashtags into words using a predefined look up dictionary
6. Use stemming to remedy a few issues on word inflections
7. Tokenisation of the input sentence (i.e tweets) into words
8. Pad or truncate the tokenised sequence to have a fixed length for each tweet

Note that we do not remove hashtags as they convey useful information when classifying Twitter text and are frequently used in a hate context.

4.3 Pre-trained Model on Toxic Content

For our pre-trained model we decided to try out Detoxify. This is a pre-trained model that classify toxic comments in online scenarios. The model is trained on the Jigsaw Unintended Bias in Toxicity Classification dataset found in Kaggle. The idea of the approach is to label hateful content by understanding context of visual elements in the sequence. The model is able to generate a toxicity label. A "Very Toxic" represents a hateful or aggressive comment, while "Not Toxic" label represents non-hateful content.

First, we consider a toxic tweet to be related to hate speech if the toxicity level is above 0.85. In order to have a better prediction furthermore, we take into account other labels provided by Detoxify such as "threat", "insult" and "identity_attack" labels. If one of these levels is highly combined with a high score in toxicity level, we mark the tweet as **Hate**. Otherwise the tweet is marked as **Non-Hate**.

5 Model Evaluation and Experimental Results

In this section we showcase evaluation results of training on the benchmark dataset. We analyse how LSMT and BiLSTM models perform compared to Detoxify Transformers pre-trained model. We provide further analysis on why the model might be biased on predicting hateful content. Lastly, we observe the trends in hateful tweets related to Donald Trump before and after Covid-19.

5.1 Experimental Setup

We use 80% of the data for training and the remaining 20% for testing. The LSTM and BiLSTM models are fine-tuned by adjusting the number of epochs (1-100) and batch sizes (32, 64, 128, 256). The results presented in this report are for models trained for 6 epochs with batch size 128. When detecting hate speech on the Kaggle dataset, LSTM and BiLSTM employ the sigmoid activation function. Additionally, the loss is defined as binary cross entropy, as the dataset only has two classes (i.e *sexist/racist* or *neither*).

5.2 Performance Evaluation on the Kaggle Dataset

In Table 5.1, we compare LSTM and BiLSTM training results. Both models score highly when predicting non-hate tweets, however, they perform less well when tested with tweets labelled as hate. One of the reasons is the unbalanced dataset that is used during training. As it only contains 7% hate content, it may not be enough for the model to exploit clear patterns of hate speech. Overall, BiLSTM achieves higher scores when compared to LSTM. We assume that the additional training propagation in BiLSTM and the usage of embeddings, might be a good indication as to why this model has a slightly higher accuracy than the other model.

In Table 5.2, we compare two different LSTM training configurations. Initially, we experimented with GloVE pre-trained word embeddings, but we found that there are a lot of out of vocabulary words due to typos and slang used in the

Table 5.1: Comparison of Model Training on the Kaggle Dataset - underlined are the best values for each metric for each class compared across the models

	LSTM			BiLSTM		
Class	**Precision**	**Recall**	**F1**	**Precision**	**Recall**	**F1**
Hate	0.64	0.43	0.52	0.57	0.45	0.50
Non-Hate	0.97	0.95	0.96	0.96	0.97	0.97
Accuracy	0.94			0.94		

Table 5.2: Comparison of LSTM model on the Kaggle Dataset. LSTM + WE refers to usage of non-trainable word embeddings, and LSTM + UE refers to usage of embeddings updated during training - underlined are the best values for each metric

	LSTM + UE			LSTM + WE		
Class	**Precision**	**Recall**	**F1**	**Precision**	**Recall**	**F1**
Hate	0.64	0.43	0.51	0.02	0.01	0.01
Non-Hate	0.96	0.97	0.97	0.93	0.98	0.97
Accuracy	0.94			0.93		

tweets Pennington et al. (2014). The results of the first model were given in the previous Table 5.1. To remedy this issue, we use embeddings trained on the corpus of training datasets, to set up the weights of the embedding layer. The model is then further trained, while updating these weights (LSTM + UE). During further experiments we also set the weights of this layer as fixed and non-trainable (LSTM + WE). As shown in Table 5.2, the LSTM + UE model is better at predicting hate speech, as it exploits different weights during training. Therefore, it learns better representations of the tweets. On the other hand, the LSTM + WE appears to underfit as it might not contain enough parameters to capture patterns pertinent to the tweets. As previously mentioned, the low precision in the hate class is also related to the lack of sizeable hate content in the dataset.

5.3 Analysis of Results and Observations

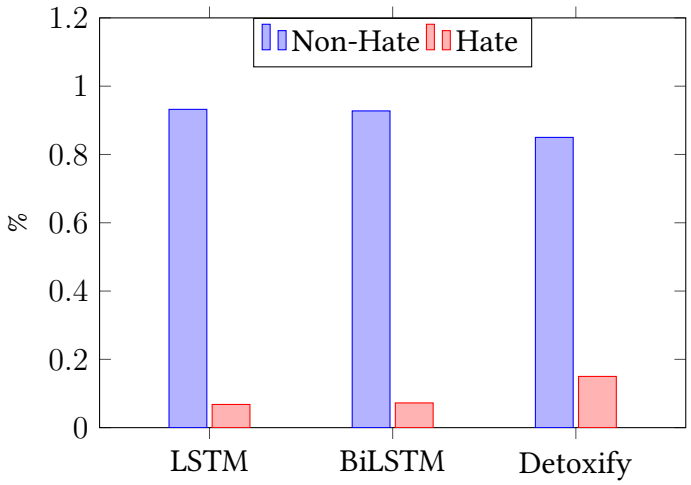
To label the collected tweets we use Detoxify, LSTM and BiLSTM. Results are summarised in Table 5.3. As it can be seen from the table, “Detoxify“ is more robust in finding hate speech compared to our trained models. This indicates that the Transformers architecture, and the dataset used during pre-training of Detoxify model provide better insights, hence enhanced prediction capabilities.

Table 5.3: Classifiction of LSTM, BiLSTM, and Detoxify of the collected tweets (All tweets).

	Tweets Counter	
Model	**Hate**	**Non-Hate**
LSTM	313	4298
BiLSTM	334	4277
Detoxify	653	3958

In Figure 5.1 we illustrate the percentage of hate and non-hate content classified by LSTM, BiLSTM and Detoxify respectively. Detoxify marks the tweets with a toxicity level which are then converted into a heuristic score to determine if a tweet is hateful or not. We can point out the slight increase in percentage of hate speech found by Detoxify, which support the finding of Table 5.3.

Figure 5.1: Labelling of tweets by LSTM, BiLSTM and Detoxify



Additionally, in Table 5.4, we observe and analyse the prediction of our LSTM model and the Detoxify model. We further include our own labeling and we list some deciding factors on why we classified a tweet in a such a way.

5.3.1 Discussion

Table 5.4: Comparison of labelled tweets from LSTM and Detoxify. In brackets of Detoxify label is the toxicity level

Raw Tweets (Unprocessed)	**Manual Label**	**LSTM**	**Detoxify**
missouri seriously needs to get their shit together stat this is not even remotely tolerable it is getting to the point we need to start looking at travel bans on states not countries	Non-Hate	Hate	Toxic(0.82) → Non-Hate
she's more of a whore than a slut. I'm certain Trump had to pay. Not any more because she's to old for him.??? prostitution isn't really profitable for women in their50.	Hate	Non-Hate	Very Toxic (0.92) → Hate
Why is #DonaldTrump from #MarALago, who has committed more #crimes and singlehandedly killed and maimed #Americans than anyone in world history free. Why the fuck is he free?	Non-Hate	Non-Hate	Very Toxic (0.97) → Hate

In Table 5.4 we compare how LSTM classified a few example tweets and how Detoxify has labelled the same tweets. We also marked our own labeling on the table above. For example, one may argue that last two tweets are quite similar in their offensiveness, hence we have labelled both of them as *Non-hate*. The LSTM model labelled the first tweet as hate. In the second row, we label the tweet as hate due to very offensive words. However LSTM model calssifies it as non-hate. This may be due to the word "whore" which we as humans can easily identify as a bad word, but if this phrase did not appear during the model training, a machine learning model would struggle with identifying it correctly.

5.3.2 How did Covid-19 lockdown and vaccination affected Trump campaign?

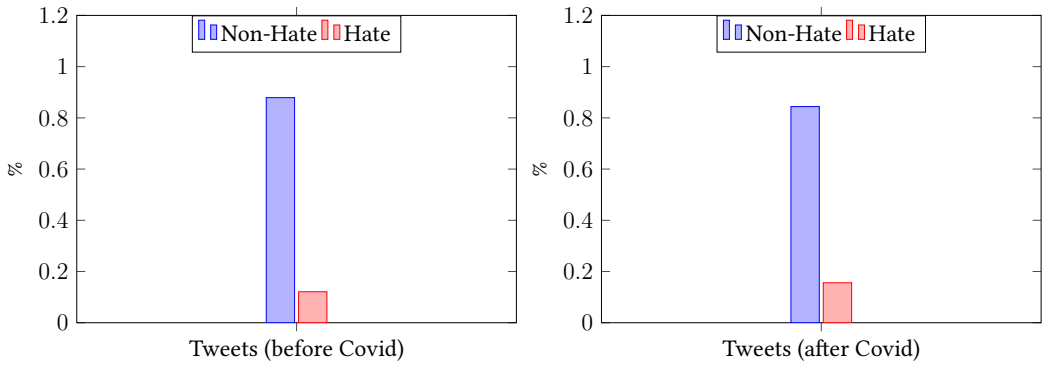


Figure 5.2: Labelling of tweets, left: labeled tweets before Covid-19 pandemic strike and right: the tweets after the Covid-19 pandemic quarantine laws and vaccinations.

In Figure 5.2, we illustrate how the tweets percentage related to Trump changed before and after Covid-19 pandemic. We notice a small increase in percentage of hate speech after 2020 lockdown and vaccination policies. There is around 1.5% hateful content in our collected tweets after 2020. In total we have 1826 tweets before Covid-19 and 2785 post-covid. The number of tweets labeled as hate before Covid is 221. The number rises to 432 tweets after the pandemic. This could be linked to the higher number of tweets collected after the pandemic (i.e 2785 tweeets post-covid). We additionally assume that the number of hateful content did not change with big leaps over time, although we believe that Covid-19 was a trigger for people to post more offensive and abusive comments online. This can also be linked with the lockdown as a vast number of people were stuck in their homes, browsing more online content.

However, the overall number of tweets detected as hate is very small. This might be associated with Twitter strict policies on hateful conduct. Most of tweets that contain inappropriate speech is probably deleted and already removed from the Twitter. A few remaining tweets might be linked to automatic bot posts or spam tweets.

6 Conclusions

The propagation of hate speech has been increasing significantly in recent years and has become a major concern for all users of social media. Despite substantial efforts from law enforcement departments, legislative bodies and social media companies, it is widely recognised that effective counter-measures rely on automated data mining techniques. This work makes several contributions to this problem.

During this work, we have identified how challenging detecting hate speech can be whilst labelling our own tweets. When issues such as missing context, sarcasm, humor, regional nuances and continuously changing abbreviations are added to the problem, the task becomes even more complicated. Particularly, new words and phrases could be considered as a cold start problem for hate speech detection models. For example, in the Covid related data, the phrase “Chinese Virus” can be easily understood as hate speech by humans, but the intelligent model which is trained by datasets that are developed before the times of Covid, will have very little chance to classify it correctly as a hate speech phrase.

Bibliography

Twitter, "The twitter rules," https://help.twitter.com/en/rules-and-policies/twitter-rules, accessed: 2022-01-04.

K. Wagner, "Twitter penalizes record number of accounts for posting hate speech," https://time.com/6080324/twitter-hate-speech-penalties/, accessed: 2022-01-04.

Z. Laub, "Hate speech on social media: Global comparisons," https://www.cfr.org/backgrounder/hate-speech-social-media-global-comparisons, accessed: 2022-01-17.

D. Sayce, "The number of tweets per day in 2020," https://www.dsayce.com/social-media/tweets-day/, accessed: 2022-01-04.

P. Badjatiya, S. Gupta, Manish, and V. Vasudeva, "Deep Learning for Hate Speech Detection in Tweets," in *Proceedings of ACM WWW17 Companion, Perth, Western Australia, Apr 2017*, 1 Jun 2017.

R. Cao, R. Ka-Wei Lee, and T.-A. Hoang, "DeepHate: Hate Speech Detection via Multi-Faceted Text Representations," in *WebSci '20: 12th ACM Conference on Web ScienceJuly 2020 Pages 11–20*, October 2020.

T. Davidson, D. Warmsley, M. Macy, and I. Weber, "Automated hate speech detection and the problem of offensive language," *CoRR*, vol. abs/1703.04009, 03 2017. [Online]. Available: http://arxiv.org/abs/1703.04009

P. Fortuna and S. Nunes, "A survey on automatic detection of hate speech in text," in *ACM Computing Surveys (CSUR)*, 2018.

R. T. Mutangai, N. Naicker, and O. Olugbarae, "Hate Speech Detection in Twitter using Transformer Methods," in *(IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 11*, No. 9, 2020.

Z. Ziqi, "Hate Speech Detection: A Solved Problem? The Challenging Case of Long Tail on Twitter," in *Information School University of Sheffield Regent Court 211 Portobello*, October 2018.

S. Kamble and A. Joshi, "Hate speech detection from code-mixed hindi-english tweets using deep learning models," *CoRR*, vol. abs/1811.05145, 2018. [Online]. Available: http://arxiv.org/abs/1811.05145

C. Chang, "Us election: The controversies that defined donald trump's four years in office," https://www.nzherald.co.nz/world/us-election-the-controversies-that-defined-donald-trumps-four-years-in-office/CYAPFNGQKNZTCMEMKRSNAV2M3M/, accessed: 2022-01-04.

N. T. Ellis, "'stand back and stand by': Rhetoric some call racist has marked trump's entire presidency," https://eu.usatoday.com/story/news/politics/elections/2020/10/13/hate-speech-common-theme-trumps-presidency/5873238002/, accessed: 2022-01-04.

A. Schmidt and M. Wiegand, "A Survey on Hate Speech Detection using Natural Language Processing," in *Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media*, August 27, 2021.

G. Xiang, B. Fan, L. Wang, J. Hong, and C. Rose, "Detecting offensive tweets via topical feature discovery over a large scale twitter corpus," in *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, ser. CIKM '12. New York, NY, USA: Association for Computing Machinery, 2012, p. 1980–1984. [Online]. Available: https://doi.org/10.1145/2396761.2398556

M. Zampieri, S. Malmasi, P. Nakov, S. Rosenthal, N. Farra, and R. Kumar, "Predicting the Type and Target of Offensive Posts in Social Media," in *Proceedings of NAACL*, 2019.

S. Bhalla, "Hate-speech-detection-on-twitter-data," https://github.com/srishb28, accessed: 2021-11-21.

A. Bisht, A. Singh, H. S. Bhadauria, and J. Virmani, "Detection of Hate Speech and Offensive Language in Twitter Data Using LSTM Model," in *Part of the Advances in Intelligent Systems and Computing book series (AISC, volume 1124)*, 2018.

G. L. D. l. P. Sarrace, R. G. Pons, C. E. C. Muniz, and P. Rosso, "Hate Speech Detection using Attention-based LSTM," in *CERPAMID, Cuba*, 2018.

M. Sap, D. Card, S. Gabriel, Y. Choi, and N. A. Smith, "The risk of racial bias in hate speech detection," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, Jul. 2019, pp. 1668–1678. [Online]. Available: https://aclanthology.org/P19-1163

Z. Waseem and D. Hovy, "Hateful Symbols or Hateful People? Predictive Features for Hate Speech Detection on Twitter," in *The 15th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, June 17, 2016.

S. Kumar and R. R. Pranesh, "TweetBLM: A Hate Speech Dataset and Analysis of Black Lives Matter-related Microblogs on Twitter," in *arXiv:2108.12521*, August 27, 2021.

K. Dataset, "Twitter sentiment analysis," https://www.kaggle.com/arkhoshghalb/twitter-sentiment-analysis-hatred-speech, accessed: 2022-01-02.

T. Mikolov, E. Grave, P. Bojanowski, C. Puhrsch, and A. Joulin, "Advances in pre-training distributed word representations," 2017.

A. Ezen-Can, "A Comparison of LSTM and BERT for Small Corpus," Sep 11, 2020.

J. Pennington, R. Socher, and C. Manning, "GloVe: Global vectors for word representation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 1532–1543. [Online]. Available: https://aclanthology.org/D14-1162