Topic Modeling for Enhancing Transformers Hate Speech Detection*

Filipe Santos, João M. C. Sousa, and Susana M. Vieira

IDMEC, Instituto Superior Técnico, University of Lisbon, Lisbon, Portugal
{filipempsantos,jmsousa,susana.vieira}@tecnico.ulisboa.pt

Abstract. This paper proposes TFusion, a novel text classification framework that integrates topic modeling from Latent Dirichlet Allocation (LDA) and deep contextual embeddings from Large Language Models such as BERT. LDA learns topic representations of samples, capturing word-frequency dependent features, while the transformer generates deep contextual embeddings, capturing context-based features. Model-level fusion is used to combine these complementary sets of features to enhance predictive performance. This paper addresses the topic of Hate Speech detection, a specific field of Natural Language Processing of high academic, governmental and corporate interest over the last decade, and applies TFusion to this problem. The framework was tested on the Stormfront Hate Speech Dataset, chosen for being one the most challenging in the field, in which state-of-the-art approaches achieve some of their lowest performances. The carried experiments used a Distil-BERT encoder inside TFusion due to its state-of-the-art performance in Hate Speech detection coupled with a lower computational demand. Moreover, 10-fold validation was conducted 10 times, totaling 100 tests, to validate the obtained results. The results show that the proposed framework outperforms both LDA coupled with a classifier and Distil-BERT, in terms of macro F1 score in the hate speech detection task (p-value=0.0046).

Keywords: Hate Speech Detection · Text Classification · Information Fusion · Model-Level Fusion · Contextual Embeddings · Distil-BERT · Latent Dirichlet Allocation.

1 Introduction

In recent years, the proliferation of hate speech has been an increasingly large societal concern, with its manifestations escalating beyond face-to-face interactions and expanding to online communication. The expansion of internet availability and user-friendliness encourages more than ever individuals to communicate and express their opinions online [13]. However, the anonymity afforded by the internet contributes to the adoption of aggressive behavior in virtual spaces [6]. Thus, online platforms serve as quick and highly accessible means to proliferate hate speech.

* Supported by LAETA and FCT

The gravity of this issue lies in the potential harm it poses to society. Governments, companies among other entities with public exposure acknowledge the need for prevention regarding this topic. In fact, they actively pursue hate speech detection and removal to safeguard their reputation, mitigating any additional related risks, such as security [9]. This heightened awareness is also reflected in the initiatives undertaken by the European Union Commission, which has introduced programs such as the No Hate Speech Movement by the Council of Europe, alongside legislative pressures on major platforms like Facebook, YouTube, Twitter, and Microsoft [13].

Hence, reliable and robust tools to both identify and block the spread of hate speech are in demand [6]. However, this task has many critical nuances, such as the definition of hate speech, where every entity has its own [19], the subjectivity of each particular language [16, 17] among other social biases [13].

This paper proposes TFusion (Topic modeling and Token-pooled contextual embeddings Fusion framework), a novel text classification framework, that integrates topic information from Latent Dirichlet Allocation (LDA) and deep contextual embeddings from Large Language Models (LLMs) such as BERT (Bidirectional Encoder Representations from Transformers). LDA learns topic representations of samples, capturing word-frequency dependent features, while the transformer generates deep contextual embeddings, capturing context-based features. Model-level fusion is used to combine these complementary set of features to enhance predictive performance.

The remainder of this paper is organized as follows. Section 2 delves into related work. The proposed framework is presented in Section 3. Section 4 details the empirical tests, including data description and preparation, hyperparameter tuning, results presentation, and respective discussion. Finally, conclusions are drawn in section 5 alongside some future work guidelines.

2 Related Work

Hate speech recognition has boomed as a research topic since 2014 [19]. As a Natural Language Processing (NLP) task, hate speech detection has had a development curve aligned with the remainder of the NLP field. This section provides an overview of existing literature, focusing on the methodologies employed and the current state-of-the-art.

Early works in the field resorted to techniques such as Bag-of-Words (BoG) to process text and obtain numerical features. Hence, the obtained models were highly based on word frequency. N-grams (with $N > 1$) and Part-of-Speech techniques were also incorporated in the data processing in some of those works so that results depend on how sentences were structured [14, 15].

Since 2014, a wider range of techniques to process text has been used. Text datasets would be processed with techniques such as topic modeling [18], word embeddings [27], and Stanford Sentence Parser [8]. Simultaneously, a wider variety of models were being tested such as Logistic Regression, Naive-Bayes, and Random Forests, among others [2, 6, 7, 12, 25]. The usage of Deep Learning also

surged during this period where Recurrent Neural Networks (RNNs) with Gated Recurrent Units (GRUs) and Long-Short Term Memories (LSTMs) [21], analogous to the remainder of the NLP field.

In more recent years, research has shifted to using transformer-based methods [5, 9], which use contextualized embeddings that make them effective in understanding the context of words and phrases within a given text. Pre-trained LLMs such as BERT, HateBERT, HateXplain, ALBERT, and Distil-BERT achieve state-of-the-art results [1, 22, 26].

Regarding performance metrics, since the surge of Hate Speech Detection as a topic of high academic interest, the utilized performance metrics remained in a larger part unchanged, preferring F1-score, Recall, and Precision for unbalanced datasets [2, 6, 7, 12, 25] and Accuracy for balanced datasets [16, 27].

3 Proposed Architecture

In this section, TFusion architecture is thoroughly presented. TFusion has 4 main components which are the Initial Text Normalization (see section 3.1), where raw text is preprocessed to suit the two next components, the Topic Modelling (see section 3.2), where the cleaned text is converted into topic memberships, the Contextual Embedding Encoding (see section 3.3), where the cleaned text is encoded using a transformer encoder, and the Classification Head (3.4), which outputs predictions based on both topic memberships and the encoded embeddings. Fig.1 shows schematically the framework proposed.

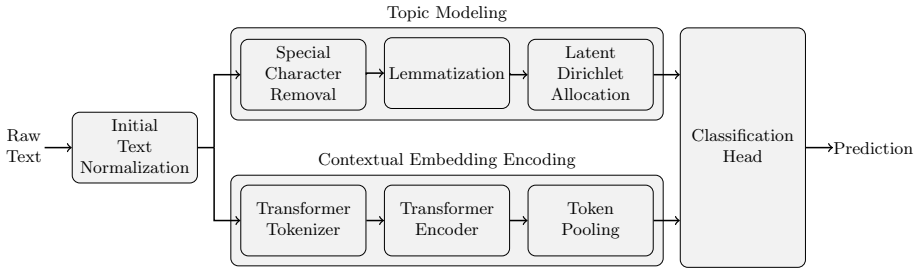


Fig. 1: TFusion

3.1 Initial Text Normalization

Initial Text Normalization functions as TFusion entry point, receiving raw text as input. In this component, data is preprocessed so that character sequences, such as URLs, email adresses, phone numbers, ID numbers and filenames, are detected to be then replaced by generic placeholders. As these sequences tend to have high variability within a specific structure, even though most of those

variations carry little to no additional meaning than the structure itself. Hence, replacing such sequences by generic placeholders results on less sparse data that is less noisy, while carrying little drawbacks [4].

3.2 Topic Modeling

Latent Dirichlet Allocation (LDA) was proposed in 2003 [3]. This technique assumes any document can be represented as mixtures of topics, where each topic is characterized by a distribution over words. Under these assumptions, LDA is a Bayesian network that aims to uncover these topics and their associated word distributions. TFusion uses LDA as a tool to create word-frequency-dependent features to be used by Classification Head.

Nevertheless, LDA particularly struggles with data sparsity. With sparse data, it becomes challenging for LDA to accurately estimate the probability distribution of words within each topic, leading to less well-defined topics [24]. Hence, in TFusion, two other preprocessing operations are performed prior to using LDA in order to diminish data sparsity.

Firstly, any uninterrupted sequence of characters with a special character is removed. This operation diminishes data sparsity by removing noise. Secondly, the text is lemmatized. Lemmatization is an NLP technique that involves reducing words to their canonical form, known as the lemma. By mapping words to their canonical form, the vocabulary reduces significantly, leading to a more concise representation of the data [20].

In sum, three operations are conducted to obtain topic memberships, two additional preprocessing operations, Special Character Removal and Lemmatization, which reduce data sparsity, and Latent Dirichlet Allocation which converts the cleaned text into topic memberships.

3.3 Contextual Embedding Encoding

In parallel to Topic Modelling, text is also processed by a pipeline inline with the current state-of-the-art LLMs for text classification. Firstly, text is tokenized, in others words, text is segmented into smaller units, tokens, in accordance to the transformer vocabulary. Tokens can correspond to words or sub-words. The tokens are then matched with their respective embeddings which are processed by the Transformer Encoder. The encoder output is reduced to a feature vector with fixed size. Different transformers use different methods for this reduction. In TFusion, Token Pooling is performed, this technique consists of extracting a token from the encoder output. For instance, models like BERT and DistilBERT extract the CLS token, the first token of the sequence also known as the classification token [11].

Multiple encoder transformers share this pipeline. Hence, several different transformers can be used for Contextual Embedding Encoding.

3.4 Classification Head

Classification Head is a neural network that receives as input both the topic representations from LDA, and the encoded embedding from the transfomer; and it outputs the prediction made by the framework regarding each sample. This network fuses both inputs according to the model-fusion framework. In other words, both inputs are fed through distinct input layers, processed in different sets of hidden layers, and the result of those sets of hidden layers is only then fused (in this case, concatenated) to be again processed by a third set of hidden layers. Hyperparameter-tuning for the classification head is conducted in Section 4.3.

4 Experimental Setup and Results

4.1 Dataset

The dataset used for all the tests conducted in this section is the Stormfront Hate Speech Dataset [10]. This dataset was created by extracting roughly 5,000 posts from a white supremacy forum, Stormfront, and having each post manually annotated at the sentence level. All in all, it is composed of 10,703 samples, where each sample is a sentence classified as hateful or not. In addition, it is also important to note that the dataset is highly unbalanced, having roughly 1 sample in every 9 labeled as conveying hate speech.

4.2 Experimental Setup

In this section the experimental setup is thoroughly described, including hardware, results validation process and TFusion operations specifications.

Firstly, the results presented in section 4.3 are 10-fold cross-validated. Nevertheless, results presented in section 4.4 result from performing 10-fold cross-validation ten times, totaling 100 tests. Additionally, all the conducted tests were run on a NVIDIA GeForce RTX 3050 GPU.

Regarding TFusion, more specifically the Initial Text Normalization, URLs, e-mail addresses, and file names were replaced by generic placeholders. Table 1 summarizes how the raw text was manipulated in this instance.

Raw Text	Processed Text
any.email@sth.sth.com	email
www.youtube.com	youtube
https://youtube.com	youtube
http://www.youtube.com	youtube

Raw Text	Processed Text
name.png	file png
name.pdf	file pdf
name.pdf	file pdf
name.jpeg	file jpeg

Table 1: URLs, email and file name normalization

Regarding Contextual Embbeding Encoding, all the tests conducted utilized Distil-BERT as the transformer of choice for TFusion. This transformer was chosen for achieving state-of-the-art performance on Hate Speech detection, namely on the Stormfront Hate Speech Dataset, while being computationally less demanding when compared to other transformers.

Regarding LDA, as stated in Section 3.2, it particularly struggles with very sparse input vector spaces [24]. This issue is diminished by lemmatization, nevertheless, for datasets where samples are very short such as the Stormfront Hate Speech dataset, such a solution is insufficient to reduce the sparsity of the input vector spaces.

To diminish this issue, a sample aggregation method was developed to create longer samples to train the LDA. This aggregation consists of concatenating training samples of the same label after Lemmatization. The method developed has two parameters n, the group size, and m, the number of times each sample will be concatenated with other samples. Having set m and n, training samples are divided by their labels, so that no two samples of distinct labels are concatenated. Then, each training sample is placed in groups of n samples with matching labels m times.

Finally, regarding lemmatization, spaCy library was utilized to perform rule-based lemmatization [23].

4.3 Hyperparameter Tuning

In this subsection, all the hyperparameter tuning conducted tests regarding TFusion are thoroughly explained.

Regarding the Topic Modeling, hyperparameter tuning was done in two stages. The first stage optimized the separation between labels on the document-topic matrix through a grid search. In the second stage, Bayesian optimization was used to optimize the hyperparameters of a neural network which predicted the label of the sample solely based on the output of LDA.

Firstly, a grid search was performed to maximize the separation between labels amongst samples. The silhouette score was chosen as a metric of separation. The hyperparameters used in this grid search were the number of topics and the data aggregation parameters, m and n. This procedure was performed on the 10 train sets obtained from the 10-fold cross-validation. These were then broken into two smaller sets, one smaller training set to train LDA using the data aggregation (Algorithm 1), and one validation set that was not subjected to data aggregation and was used to evaluate the results of LDA. The obtained results are presented in Fig.2.

The obtained results favor the use of a lower number of topics and higher values for m and n. Moreover, upon closer inspection, it is observable that the lower the number of topics the more sensitive LDA is concerning parameters m and n. As the usage of two topics also led to the lowest silhouette score of the grid search. Furthermore, it is noteworthy as well, that if samples are not aggregated ($n = 1$) before training LDA, both other parameters do not have much impact on the final result. That was expected for parameter m, as

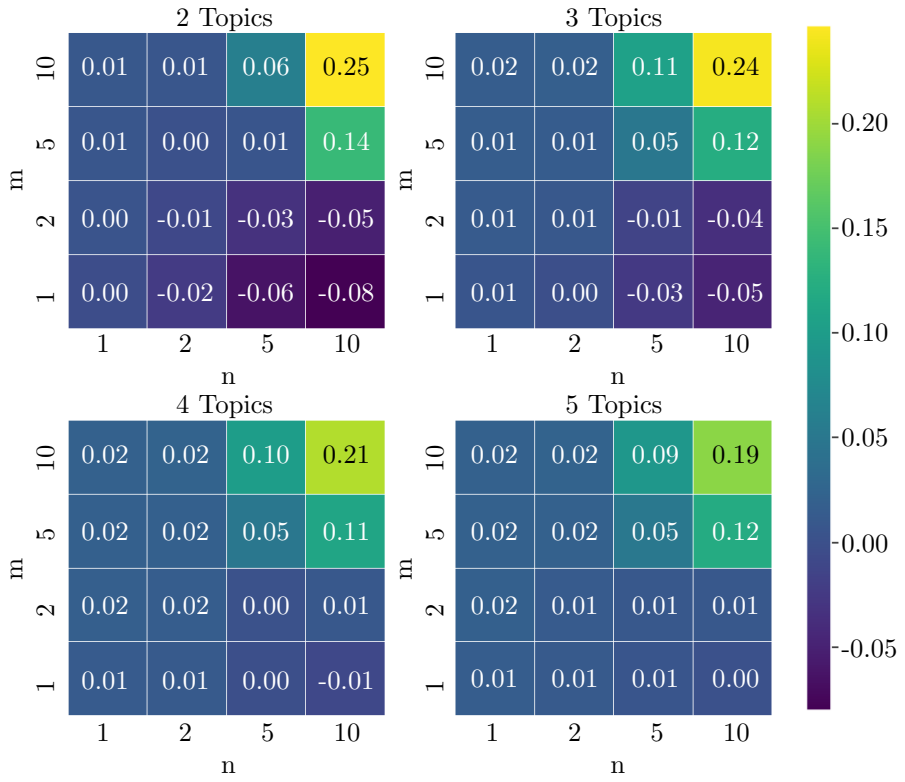


Fig. 2: Grid search silhouette score results, where n is group size and m is number of times each sample is grouped

increasing said parameter would only result in feeding repeated samples into LDA. Finally, the best obtained was 2 topics with both m and n equal 10. Hence, those are parameters in the remainder of this section.

Regarding the second stage of hyperparameter tuning, a dense neural network was designed to predict the label of each sample based on the output of LDA. The obtained network architecture will be used to help design the Classification Head Network as well as to draw a comparison to TFusion performance.

In this stage, Bayesian optimization was used to optimize the number of layers, the layer size, the dropout rate, and the learning rate. Table 2 shows the hyperparameter spaces used in this optimization under *Search Space*, it is important to note that the search performed on the learning was logarithmic. The data used for the Bayesian optimization followed the same procedure used in the gridsearch. Table 2 shows the obtained results under column *results*.

	Search Space	Results
Macro F1	-	62.4%
No. Hidden Layers	1, 2, 3, 4 or 5	3
Hidden Layers Size	8, 16, 32 or 64	64
Dropout Probability	[0.05, 0.2]	0.09
Learning Rate	[0.00001, 0.001]	0.0006

Table 2: Hyperparameter Search Spaces and Results from Bayesian Optimization

4.4 Results

Having concluded the hyperparameter tuning, the TFusion Classification Head network (see Fig.3) was designed taking into account both the results from the Bayesian optimization (left-upper block) and the Distil-BERT classification head (left-lower block). The common layers were built using characteristics from the other blocks.

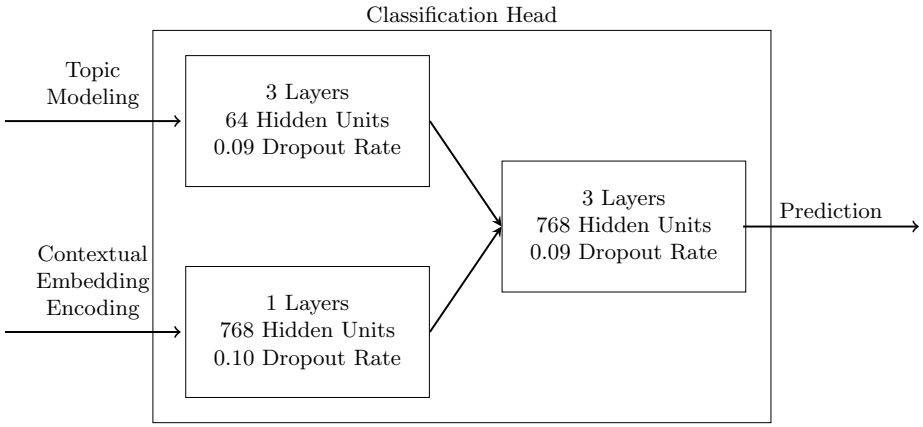


Fig. 3: TFusion Classification Head Architecture

Finally, 10-fold cross-validation was conducted 10 times to train TFusion, totaling 100 tests. In parallel, both Distil-BERT using its default classification head, and the LDA with network optimized in section 4.3 were trained following the same methodology to establish baselines. The results for these tests, as well as for both works in [22] and [1], are presented in Table 4. Additionally, Table 3 provides extra information regarding all the methods.

Regarding the LDA coupled with a classifier and Distil-BERT, the results are in line with what is expected, being consistent with the historical trends in both hate speech detection and NLP. The LDA coupled with a classifier obtained the worst results. Whereas Distil-BERT proved to be a superior hate speech

Method	Model Size	Expert Knowledge	Training Stages
BERT-large [1]	8.0GB	No	1
Distil-BERT+st[22]	1.5GB	Yes	3
LDA+classifier	∼0GB	No	1
Distil-BERT	1.5GB	No	1
TFusion	1.6GB	No	1

Table 3: Method Charateristics

	Macro F1	F1 (Hate)	F1 (No Hate)	Precision (Hate)	Recall (Hate)
BERT-large [1]	79.2	62.5	95.8	-	-
Distil-BERT+st[22]	78.4	-	-	-	-
LDA+classifier	63.2±2.6	37.2±3.9	89.3±1.8	30.6±4.0	48.1±6.1
Distil-BERT	76.7±2.4	58.1±4.5	95.2±0.5	64.9±5.8	53.5±7.6
TFusion	77.1±2.0	59.0±3.7	95.2±0.4	63.2±4.2	55.7±5.0

Table 4: Results in percentage

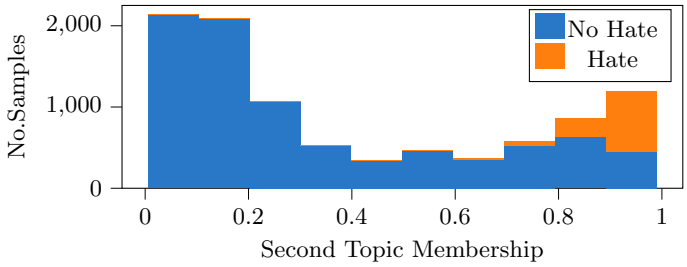
detector, achieving a 76.7% macro F1 score compared to the 63.5%, as well as outperforming it in all the other metrics detailed in Table 4.

The TFusion outperforms both individual models, improving 0.4% macro F1 score compared to the best-performing one (p-value=0.0046). A closer examination of this metric reveals that the proposed approach outperformed Distil-BERT in 64 out of 100 tests, tying in 4 instances and lagging in 32. Analyzing additional metrics indicates that enhancement in performance comes from a 0.9% increase in F1 (Hate), while maintaining comparable performance in F1 (No Hate) with only a 0.1% decrease on average. Moreover, the results suggest that the proposed approach tends to classify more samples as hateful, evidenced by an average increase in recall by 2.2%, coupled with a 1.7% average decrease in precision. Notably, the proposed approach also exhibits greater robustness in its predictions, as evidenced by lower standard deviations across all metrics when compared to Distil-BERT.

TFusion shows competitive results with both the Distil-BERT+st approach presented in [22], and BERT-large trained in [1] even though both of these methodologies show a higher macro F1. Regarding BERT-large, it is currently, to the authors' knowledge, the best-performing model in this dataset. Nevertheless, this transformer is much more computationally demanding, around 8 GB just to store the model structure, compared to 1.5 GB for Distil-BERT, or 1.6 GB for the entirety of TFusion. Regarding Distil-BERT+st, it requires the most complex training method from the compared methods, requiring 3 training stages, token debiasing, sample debiasing, and training on the task itself. In addition, the first two training stages require expert knowledge to indicate which words make samples biased. Nevertheless, both these methodologies can be seamlessly incorporated into TFusion.

Finally, it is also noteworthy that while the LDA by itself does not output features that can be used to create a hate detector, it is the only component in the framework that can be directly interpreted. The trained LDA can be used to analyze how both specific words and samples are distributed across the topic space. Fig.4 showcases one of the conducted tests.

(a) Words membership regarding each topic. The utmost left side of the colorbar represents 100% membership to the first topic, right represents 100% membership to the second topic



(b) Histogram of sample membership regarding the second topic

Fig. 4: Topic Modelling

From this example, it is observable in Fig.4b that most samples labeled as including hate speech have more than 70% membership to the second topic, and, hence, less than 30% membership to the first topic. In addition, resorting to Fig.4a gives insight into what belonging to the second means. In this case, from the highlighted words, it is observable that words such as "fun" and "good" are more common on the first topic, words such as "people" and "man" have a more similar distribution among topics, and "bad" and "woman" are more frequent in the second topic.

5 Conclusion and Future Work

This paper proposed a novel text classification framework, TFusion. The framework makes use of LDA for topic modeling with an LLM for creating deep contextual embeddings. Additionally, it utilizes model-level fusion to synergistically incorporate both word-frequency and context-based features into the framework predictions.

TFusion was tested on the hate speech detection use-case, more precisely, it was tested on the Stormfront Hate Speech Dataset. Additionally, Distil-BERT

was chosen to be the transformer incorporated into the framework for showcasing both state-of-the-art results as being a more computationally-light option. The experimental results demonstrated that the proposed framework outperforms LDA coupled with a classifier and Distil-BERT (p-value=0.0046), achieving a 77.1% macro F1 score. Additionally, the fusion approach gave comparable results with BERT-Large and Distil-BERT+st while being a computationally less demanding method.

In conclusion, TFusion achieved competitive results regarding hate speech detection state-of-the-art. Future works should concern testing this framework on other datasets and further exploring the interpretability of the framework.

Acknowledgements

The authors acknowledge Fundação para a Ciência e a Tecnologia (FCT) financial support via the projects LAETA Base Funding (DOI: 10.54499/UIDB/50022/2020) and LAETA Programatic Funding (DOI: 10.54499/UIDP/50022/2020) and Filipe Santos' work was supported by the Ph.D. Scholarship 2022.12077.BDANA from FCT.

References

1. Alatawi, H.S., Alhothali, A.M., Moria, K.M.: Detecting white supremacist hate speech using domain specific word embedding with deep learning and bert. IEEE Access **9**, 106363–106374 (2021)
2. Badjatiya, P., Gupta, S., Gupta, M., Varma, V.: Deep learning for hate speech detection in tweets. In: Proceedings of the 26th international conference on World Wide Web companion. pp. 759–760 (2017)
3. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. Journal of machine Learning research **3**(Jan), 993–1022 (2003)
4. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al.: Language models are few-shot learners. Advances in neural information processing systems **33**, 1877–1901 (2020)
5. Bugueño, M., Mendoza, M.: Learning to detect online harassment on twitter with the transformer. In: Machine Learning and Knowledge Discovery in Databases: International Workshops of ECML PKDD 2019, Würzburg, Germany, September 16–20, 2019, Proceedings, Part II. pp. 298–306. Springer (2020)
6. Burnap, P., Williams, M.L.: Cyber hate speech on twitter: An application of machine classification and statistical modeling for policy and decision making. Policy & internet **7**(2), 223–242 (2015)
7. Burnap, P., Williams, M.L.: Us and them: identifying cyber hate on twitter across multiple protected characteristics. EPJ Data science **5**, 1–15 (2016)
8. Chen, Y., Zhou, Y., Zhu, S., Xu, H.: Detecting offensive language in social media to protect adolescent online safety. In: 2012 international conference on privacy, security, risk and trust and 2012 international confernece on social computing. pp. 71–80. IEEE (2012)
9. Chhabra, A., Vishwakarma, D.K.: A literature survey on multimodal and multilingual automatic hate speech identification. Multimedia Systems **29**(3), 1203–1230 (2023)

10. De Gibert, O., Perez, N., García-Pablos, A., Cuadros, M.: Hate speech dataset from a white supremacy forum. arXiv preprint arXiv:1809.04444 (2018)
11. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
12. Djuric, N., Zhou, J., Morris, R., Grbovic, M., Radosavljevic, V., Bhamidipati, N.: Hate speech detection with comment embeddings. In: Proceedings of the 24th international conference on world wide web. pp. 29–30 (2015)
13. Fortuna, P., Nunes, S.: A survey on automatic detection of hate speech in text. ACM Computing Surveys (CSUR) **51**(4), 1–30 (2018)
14. Greevy, E.: Automatic text categorisation of racist webpages. Ph.D. thesis, Dublin City University (2004)
15. Greevy, E., Smeaton, A.F.: Classifying racist texts using a support vector machine. In: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval. pp. 468–469 (2004)
16. Kwok, I., Wang, Y.: Locate the hate: Detecting tweets against blacks. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 27, no.1, pp. 1621–1622 (2013)
17. Nobata, C., Tetreault, J., Thomas, A., Mehdad, Y., Chang, Y.: Abusive language detection in online user content. In: Proceedings of the 25th international conference on world wide web. pp. 145–153 (2016)
18. Oboler, A., Connelly, K.: Hate speech: A quality of service challenge. In: 2014 IEEE Conference on e-Learning, e-Management and e-Services (IC3e). pp. 117–121. IEEE (2014)
19. Paz, M.A., Montero-Díaz, J., Moreno-Delgado, A.: Hate speech: A systematized review. Sage Open **10**(4), 2158244020973022 (2020)
20. Piskorski, J., Wieloch, K., Sydow, M.: On knowledge-poor methods for person name matching and lemmatization for highly inflectional languages. Information retrieval **12**, 275–299 (2009)
21. Pitsilis, G.K., Ramampiaro, H., Langseth, H.: Detecting offensive language in tweets using deep learning. arXiv preprint arXiv:1801.04433 (2018)
22. Song, R., Giunchiglia, F., Li, Y., Shi, L., Xu, H.: Measuring and mitigating language model biases in abusive language detection. Information Processing & Management **60**(3), 103277 (2023)
23. Spacy lemmatizer, https://spacy.io/api/lemmatizer, last accessed 4 February 2024
24. Syed, S., Spruit, M.: Full-text or abstract? examining topic coherence scores using latent dirichlet allocation. In: 2017 IEEE International conference on data science and advanced analytics (DSAA). pp. 165–174. Ieee (2017)
25. Waseem, Z., Hovy, D.: Hateful symbols or hateful people? predictive features for hate speech detection on twitter. In: Proceedings of the NAACL student research workshop. pp. 88–93 (2016)
26. Wullach, T., Adler, A., Minkov, E.: Fight fire with fire: Fine-tuning hate detectors using large samples of generated hate speech. arXiv preprint arXiv:2109.00591 (2021)
27. Yuan, S., Wu, X., Xiang, Y.: A two phase deep learning model for identifying discrimination from tweets. In: EDBT. pp. 696–697 (2016)