Engineering Universe for Scientific Research and Management
ISSN (Online): 2319-3069 Vol. XVI Issue VI
June 2024

A Comprehensive Framework of Deep Learning Algorithm for Effective Hate Speech Detection on Twitter

Rohit Kumar Srivastva[1], Chetan Agrawal[2], Rashi Yadav[3]
Dept. of CSE, Radharaman Institute of Technology & Science, Bhopal, India[1, 2, 3]
rohitbuxar03@gmail.com[1], chetan.agrawal12@gmail.com[2], rashi6yadav@gmail.com[3]

Abstract

Hate speech is an undesirable phenomenon with severe psychological and physical consequences. The emergence of mobile computing and Web 2.0 technologies has increasingly facilitated the spread of hate speech. The speed, accessibility and anonymity afforded by these tools present challenges in enforcing measures that minimize the spread of hate speech. The continued dissemination of hate speech online has triggered the development of various machine learning techniques for its automated detection. However, current approaches are inadequate because of further challenges such as the use of domain-specific language and language subtleties. Recent studies on automated hate speech detection have focused on the use of deep learning as a possible solution to these challenges. Although some studies have explored deep learning methods for hate speech detection, there are no studies that critically compare and evaluate their performance. This work investigates the use of deep learning algorithms as possible solutions to hate speech detection on Twitter. Three taxonomic classes of deep learning algorithms, namely, Traditional deep learning algorithms, Traditional algorithms with partial attention mechanism and Transformer models, which are entirely based on the attention mechanism, are evaluated for performance, using two publicly available corpora. One of the datasets contained 24786 tweets annotated into three different classes, while the other dataset contained 2300 tweets annotated into two different classes. The algorithms were tested on a wide spectrum of tweets containing different forms of hate speech. The efficacy of the deep learning algorithms was objectively evaluated using six state-of-the-art statistical evaluation metrics: precision, F- measure, recall, accuracy, Mathew’s correlation coefficient and area under the curve.

Keywords: *Hate Speech, Web 2.0, Twitter, Attention Mechanism, BERT.*

1. Introduction

Hate speech is a collective term for utterances or statements which disseminate, trigger, encourage or justify hatred, segregation and violence against an individual or group of individuals [1]. Typical forms of hate speech include racism, tribalism, sexism, xenophobia, and islamophobia. No single hate speech definition has been unanimously accepted as the gold standard by the research community. However, various researchers concur that it targets underprivileged persons in a way that may be deemed harmful to them [2]. Hate speech promotes prejudice, which can undermine people, sow seeds of discord between different societal groups and eventually lead to deeper social cohesion problems [3]. Divisions in societal cohesion and attacks on the egos of hate speech victims have the potential to fuel social unrest and hate crimes [4]. For example, hate speech fueled xenophobic attacks in the KwaZulu-Natal province of South Africa, where seven immigrants died and approximately 5000 others displaced between March 2015 and May 2015 [5].

In the past, the propagation of hate speech has been achieved mainly through the use of traditional electronic and print media such as newspapers, radio, and television. For example, the holocaust, which resulted in mass killings of Jews, also had its roots in hate speech propaganda, which was propagated using the technologies of those days. Furthermore, hate speech leading to the Rwandan genocide in 1994 was spread through radio and print media [6]. Since then, communication technologies have evolved to include the Internet and mobile devices, allowing rapid exchange of information.

The emergence of Web 2.0 tools such as Twitter and Facebook has transformed communication by allowing users in different parts of the world to seamlessly compile, collaborate, and share their content with others. Given the meteoric rise of user-generated content on platforms such as Twitter, the volume of online hate speech is growing [7]. Platforms such as Twitter enable users to instantaneously post different kinds of messages in different formats such as text, images, videos, and metadata, sometimes in the form of emojis, mentions, emoticons, uniform resource locators, and hashtags for social media users to view, comment, and share with other users [8]. Tweets are generally rife with idioms, acronyms, phonemes, homophones, and figures of speech like onomatopoeia, which can complicate the understanding of hateful speech. Moreover, the Twitter restrictions on the number of allowable characters encourage the usage of unconventional and incomprehensible abbreviations, misspellings, grammatical errors, and slang terminologies. Millions of tweets are generated daily, enabling the creation of datasets large enough for analysis [9]. In 2017 alone, Twitter had 330 million active users per month, and 157 million of the users were active daily, sharing approximately 500 million tweets each day [8]. It is projected that at least one-third of the world population will be using social media by the end of 2021 [10].

The large volumes of harmful messages posted on Twitter necessitate the development of techniques to curb their continued dissemination. To address this, some governments in the developed world have instituted laws to prohibit hate speech in face-to-face conversations and on the internet media [11]. Although such legislation acts as a deterrent, it does not entirely stop determined individuals from posting content containing hate speech.

Besides the broader societal implications of hate speech, uncontrolled propagation also negatively impacts the reputation of online host platforms such as Twitter and Facebook [12]. In response to this challenge, organizations such as Facebook and Twitter currently have employees dedicated to the task of manually deleting content perceived to contain hate speech. In addition, Facebook and Twitter users are advised to label and report content they deem unsuitable or harmful to society. However, such interventions are laborious for human annotators, and they are also prone to subjective human judgment [13]. These methods are stressful for human annotators, and they have been linked to post-traumatic stress disorders [14]. Critics have argued that the use of human annotators is insufficient since the messages are only deleted after they have been posted and possibly after the messages have inflicted harm already [15]. Hate speech may also be expressed in slang or other languages that the annotator may not understand. There are 7117 distinct languages used for communicating verbally and in written form [16]. Given this high number of languages, it is not practical for human annotators to understand all the languages used in social media.

Machine learning-based hate speech recognition models have been proposed in response to the shortcomings of human annotators and legislation. Classical machine learning algorithms and deep learning algorithms are the two taxonomic subclasses of machine learning algorithms. Classical algorithms make use of handcrafted features, which consume much time and are ordinarily insufficient [17]. As a result, classical algorithms fail to capture semantic and syntactic representations of text effectively. Deep learning algorithms, on the other hand, carry out end-to-end training, allowing the model to encode salient feature representations. Deep Neural Networks have been proven to outperform classical models based on n-gram features [18]. Furthermore, deep learning algorithms such as Recurrent Neural Networks (RNN) are capable of preserving sequential information over periods of time, which allows easier integration of contextual information in text classification tasks [19]. Although context helps distinguish hate from non-hate texts, it has largely been excluded from detection models [20]. Deep learning models can capture complex data representations making them applicable to the identification of hate speech, where the language used is highly ambiguous. However, no studies have focused on objective comparative evaluations of deep learning algorithms, making it difficult to understand the most appropriate algorithms in addressing the hate speech phenomenon in online spaces [21].

This work, therefore, was aimed at finding the best performing deep learning algorithm for detecting hate speech. To achieve this, an experimental comparison of deep learning algorithms for hate speech detection was carried out. Ten deep learning algorithms representing traditional deep learning algorithms and recent transformer-based algorithms were selected for investigation, namely, Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), Long Short Term Memory (LSTM), Gated Recurrent Unit (GRU), Multiplayer Perceptron (MLP), Bidirectional Encoder Representations from Transformers (BERT), Distilled Bidirectional Encoder Representations from Transformers (DistilBERT), Robustly optimized Bidirectional Encoder Representations from Transformers approach (RoBERTa) and XLNet.

2. Literature Review

In recent years, detecting hate speech in online text has become a significant focus in NLP research. Initially, studies relied on conventional machine learning algorithms like SVM, KNN, Random Forest, and Decision Tree, using various feature types (for example, syntactic, semantic, sentiment, and lexicon) to identify hate speech [22]. However, the rise of deep neural networks has prompted extensive exploration into their effectiveness for NLP-related problems [23]. Notably, Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs) have emerged as prominent options and are frequently assessed for hate speech detection.

Researchers often choose different deep learning models tailored to the text's characteristics. For shorter texts where capturing detailed context matters less, CNNs have become popular due to their adeptness at grasping local patterns across various text classification tasks [24], [25],

[26]. On the other hand, when dealing with longer text sequences that demand a better grasp of semantic features and context, RNNs like Long Short-Term Memory (LSTM) networks and Bidirectional LSTMs (BiLSTMs) shine [27], [28], [29]. These models efficiently capture contextual information and word dependencies, proving advantageous in tasks like sentiment analysis and document classification.

In the realm of hate speech detection, Warner et al. [30] conducted a seminal study concentrating on identifying anti-Semitic language as a form of hate speech. Alshalan and Al-Khalifa [31] delved into classifying Arabic hate tweets using CNNs, RNNs, and bidirectional encoder representations from transformers (BERT). Employing word2vec as embedding layers via the Continuous Bag of Words (CBOW) method, their findings revealed that BERT didn't perform well for this task, resulting in an approximate 10% drop in performance, while the CNN achieved an f-score of 0.79. Another notable exploration by Waseem and Hovey [32] targeted hate speech on Twitter, particularly racism and sexism. They investigated features, including user demographics, lexical usage, geographic information, and character n-grams. Their study emphasized that using character n-grams with a maximum length of four proved to be the most effective approach. Furthermore, integrating gender as an additional feature led to a slight improvement in the obtained results.

Vashistha and Zubiaga [33] examined six publicly available datasets to identify hate speech in English and Hindi text. They constructed a logistic regression-based model, incorporating Term Frequency - Inverse Document Frequency (TF-IDF) and Part-of-Speech (POS) features. This base model's performance was compared with a hierarchical neural network, which utilized several CNN filters and the BiLSTM model. The base model achieved an accuracy rate of 85%, while the neural network attained an accuracy rate of 83%. In Khan et al. [34], a proposed neural network architecture called BiCHAT combines BERT-based embedding, BiLSTM, and deep CNN with a hierarchical attention mechanism. The attention layers will apply on word and sentence levels, allowing focus on the most important words and phrases in the text while ignoring irrelevant information. The proposed approach was evaluated on several popular Twitter hate speech datasets and performed better than the base model.

Modha et al. [35] proposed a real-time model to identify and visualize hate comments from Facebook and Twitter. This model can be used as a plug-in tool in web browsers to monitor online hate speech effectively. Initially, the authors used traditional machine learning algorithms such as SVM and logistic regression as a baseline model. Subsequently, they experimented with more advanced models such as CNN, BiLSTM, and BERT transformers. The experimental results showed that the proposed models achieved an F1-score of 0.64 on the Facebook dataset and 0.58 on the Twitter dataset. Kapil and Ekbal [36] introduced a multi-task learning framework designed to identify multiple interconnected categories of hate speech, including offensive language, racism, and sexism. Multiple neural networks were developed, encompassing architectures such as CNNs, LSTM networks, and a combination of CNN and GRU. These networks were trained for both single-task and multi- task learning scenarios. The initial training of the models was carried out for individual classes, and subsequently, a shared neural network was developed to perform the combined classification task. Rodriguez-Sanchez et al. [37] conducted an experimental study to assess the effectiveness of deep learning, machine learning, and transformer learning approaches in detecting hate speech specifically in Spanish language text. The results indicated that the transformer approach outperformed the other methods, achieving the highest F1-score of 0.75 for hate classification.

Mossie and Wang [38] introduced a method targeting the recognition of vulnerable communities through hate speech detection techniques. They utilized word2vec word embedding and n-grams for feature extraction, followed by classification using machine learning and deep learning algorithms. Moreover, they expanded the hate word lexicon by integrating co-occurring word vectors with the highest similarity, enabling the identification of the target ethnic community based on matched hate words. Ameur et al. [39] presented a dataset of 10,828 Arabic tweets addressing hate speech related to COVID-19. They performed fundamental analyses using pre-trained models, highlighting the efficacy of these models in detecting hate speech and false information in the complex Arabic language context. Meanwhile, Khanday et al. [40] investigated hate speech detection on Twitter during the COVID-19 pandemic, employing various feature extraction methods such as TF/IDF, bag of words, and word length. Decision tree classifiers notably emerged as the most effective, achieving a remarkable 97% accuracy in hate speech detection.

Del et al. [41] introduced Social HaterBert, a model tailored for hate speech identification in English and Spanish tweets, showcasing improvements over the earlier HaterBert model. Employing Bert For Sequence Classification and 'BERT' for hate speech classification, the model demonstrated performance gains ranging from 3% to 27% compared to HaterBert. Additionally, the authors proposed a method to construct a hate speech user graph using user profile attributes, potentially enhancing hate speech detection in multilingual social media discussions. Furthermore, Fortuna et al. [42] conducted an extensive study using a dataset for hate speech, toxicity, abusive language, and offensive content classification. They experimented with various models, including BERT, ALBERT, fasttext, and SVM, trained on nine publicly available datasets, evaluating both intra-dataset and inter-dataset model performance to gauge their generalizability across different hate speech categories and datasets.

Overall, while progress has been made in detecting hate speech, many studies have mainly used small datasets from single platforms like Twitter, Facebook. Relying on these limited sources might affect how well these methods work in the real world, especially across different languages or platforms. To make these methods more reliable, future research should consider using more diverse and larger datasets from various sources.

3. Proposed Method

This secstion details the steps taken to meet the set objectives. The systematic approach employed in this study is known as experimentation. Firstly, the hate speech dataset acquisition process is discussed. Thereafter, the preprocessing of the acquired datasets is clearly explained, followed by a discussion of the selected feature representation method. Lastly, the training and classification process of the selected deep artificial neural networks is described. The subsequent sections elaborate on each of the methodological steps involved in this study.

3.1 Dataset

The dataset used in this study include the hate speech and offensive language dataset (HSO). We propose DistilBERT a streamlined version of BERT that uses only half the number of parameters of BERT but retains the performance of BERT in many text processing tasks while making the inference 60% faster than BERT. DistilBERT was created by removing token type embeddings and pooler from the default architecture of BERT. DistilBERT further reduced the number of layers by 50%, thereby significantly reducing the footprint of the model

3.1.1 The Hate Speech and Offensive Language (HSO) Dataset

The multiclass hate speech and offensive language dataset originally created by Davidson et al. (2017) was chosen for the experimental comparisons reported in this research because it contains different types of hate speech, and it has a comparatively high number of instances.

The dataset, as well as the results achieved by [11], provides a platform to measure improvements that could be achieved with the dataset and compare results, using various deep learning-based models developed by researchers who used the same dataset. This dataset had 24783 Twitter text messages categorized and labeled into three classes: neutral speech, offensive language and hate speech. 77.4% of the instances are labeled as neutral, 16.8% as offensive and 5.8% as hate. The tweets in the dataset were manually annotated by Crowd Flower (CF) employees. The employees were asked to label each tweet as either containing hate or not. In libeling the datasets, they were guided by the definition of Davidson et al. (2017, p.512), which describes hate speech as "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 ". Annotators were advised not only to look at the presence of certain words in a given tweet but also to consider the context surrounding words or phrases. A minimum of three annotators was assigned to code each tweet. The intercoder-annotator agreements score provided by Crowd Flower is 92%.

3.2 Data Pre-processing

Text preprocessing is an essential part of NLP tasks which transforms text into a form ready for input into text classification algorithms. Due to the conversational nature of Twitter texts, preprocessing was applied to convert the Tweets into a format that is more predictable and analyzable for the task of automated hate speech detection. Preprocessing also minimized feature sparsity in the feature representations. Preprocessing is a proven technique for improving the predictive capability of classifier algorithms [43]. Furthermore, preprocessing reduces the computational resources needed by a classifier while minimizing the overall training time [44]. The processes involved in preprocessing the tweets in the datasets include tweet cleaning, text normalization, stop word removal and removal of null values, as explained in the next section.

Machine learning algorithms accept features in numerical form only. Therefore, it was necessary to convert word features into a numerical format for input into classifier algorithms. Conversion of word features into numerical form can be achieved using different techniques such as word embeddings and the Bag of Words approach. Traditional feature representation methods such as the Bag of Words (BOW) approach suffer from several disadvantages as compared to word embeddings. Tuning deep neural networks can be challenging. Careful selection of parameters in neural networks can be the difference between superior and inferior performance. Poorly selected hyperparameters may lead to poor learning by algorithms. Figure 1 illustrates the major steps followed in implementing each of the deep learning algorithms.

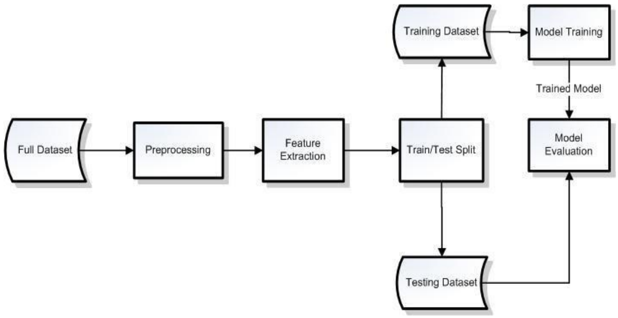

Figure 1: Illustration of hate speech detection using traditional deep learning algorithms

3.3 Attention Mechanism

The attention mechanism was implemented in conjunction with the Bidirectional LSTM at the word level. At every phase t, the Bidirectional-LSTM receives as input word vector containing semantic and syntactic information, known as word embedding. Thereafter, an attention layer was applied over each hidden state hˆ t. The model's attention weights are learned through the joining past hidden state of the Post-Attention LSTM (Pos-Att-LSTM and the current hidden state of the Bidirectional LSTM. The Post-Att-LSTM network detects the presence or absence of hate within a text.

The bidirectional LSTM operates basically the same way as the vanilla LSTM, but the processing of the incoming text is from both the left and right as opposed to one way. The Bidirectional-LSTM was investigated with the aim of capturing long-range and backwards dependencies based on its success in earlier studies. Two fundamental building blocks of the attention-based LSTM for hate speech detection are the attention layer and the Post attention Layer.

3.3.1 Attention Layer

The attention mechanism allows the Bidirectional LSTM to decide parts of the tweet on which the model should focus. The model learns what to focus on based on the input tweet and what it will have produced to date. The goal of the attention layer is to get a context vector capable of capturing salient information and input it to the subsequent level.

3.3.2 Post Attention Layer

The Post-Attention-LSTM is responsible for assigning tweets to either the hate or neutral category. At each time step, the network receives the context vector, propagated until the final hidden state.

The full architecture of the bidirectional LSTM with attention used in this study is summarized in Figure 2. As Illustrated, the model includes the bidirectional LSTM layer, the attention layer and the post attention layer.

3.4 Transformer Algorithms

Transformers mirror the standard text classification, which includes preprocessing the text, model training and predictions on unseen data. The transformer methods are selected in this study due to their built-in self-attention feature, which facilitates the capture of long-term dependencies while enabling parallel processing of input features. The capture of long-term dependencies enables anaphora resolution, which has been identified as a major limitation encountered when classifying subjective text. Nevertheless, the majority of transformers are resource-intensive, making them less applicable in environments with scarce resources. Given this background, in this study, full transformer methods were investigated alongside transformer methods that have been streamlined and customized for resource-constrained environments, for example, DistilBERT, which is a streamlined version of the BERT architecture. Other transformer methods explored in this study are RoBERT, XLNet, and BERT. Figure 3 shows the architecture of the transformer-based model hate speech detection models explored in this study.

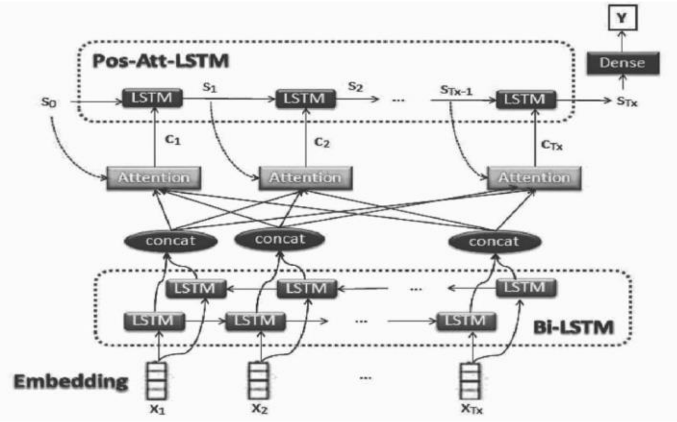

Figure 2: Architecture of the bidirectional LSTM with attention mechanism

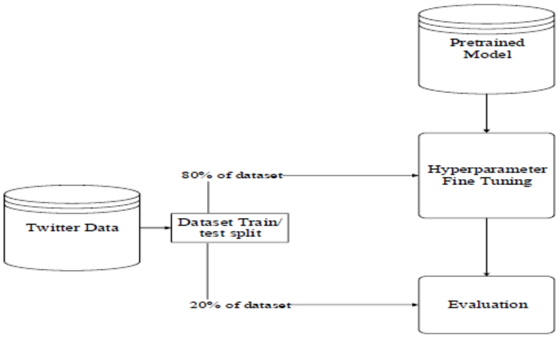

Figure 3: Transformer architecture for hate speech detection

Evaluation metrics were used to measure the quality and performance of machine learning models. Evaluating machine learning models or algorithms are crucial for any study because they present an objective way of assessing model performance. However, the evaluation was based on widely used metrics, which are accuracy, precision, recall, F-measure and area under the curve. Accuracy, precision, recall and F- measure can be calculated from values of the confusion matrix, which are True Positives (TP), True Negatives (TN), False Positives (FP) and False Negatives.TP represents actual positives predicted as positive, TN represents Actual Negatives that are predicted correctly as Negative, FP represents actual negatives that are wrongly predicted as positives, and FN represents actual positives that are wrongly predicted as negatives.

4. Result Analysis

Results of the proposed DistilBERT method was compared against results computed by BERT, XLNet, RoBERTa and attention-based LSTM. We split the dataset in the ratio of 80:20 for model training and testing, respectively. The algorithms were analyzed in terms of six standard functional metrics of accuracy, precision, recall and F-measure, Mathews correlation coefficient (MCC) and evaluation loss. The results are presented based on the ability of the models to detect hate tweets.

A. Analysis of Accuracy

The experimental results of the proposed DistilBERT method, along with five baseline algorithms are presented in Table II and Fig. 2. It can be observed that the proposed DistilBERT method recorded the highest average accuracy of 92%. It is worth mentioning that the differences in accuracy scores for all transformer-based methods were negligible. This may be attributed to the fact that they all use standard extensively tested pre-trained models. Expectedly all transformer-based algorithms performed better than the LSTM with Attention. The least performing transformer method had an accuracy of 89%, which is superior to LSTM with attention which had 66% accuracy. This trend is because of the ability of the transformers to capture long-term dependencies better than LSTM with Attention.

B. Analysis of Precision

It can be observed from Table III and Fig. 3 that the DistilBERT (base-uncased) and XLNet algorithms jointly recorded the highest precision score of 75% whilst LSTM with attention recorded the least precision score of 65.9%. Although the LSTM with attention recorded the least result, it should be noted that this score is higher than scores recorded by methods using classical machine learning [29]. This result confirms the literature position that attention improves performance in NLP tasks [19].

TABLE 1: Accuracy of Four Hate Speech Detection Algorithms and DistilBERT on the HSO Dataset

Algorithm	Method Name	Accuracy
BERT	bert-base-uncased	0.90
RoBERTa	robert-base	0.91
RoBERTa	robert-base-openai-detector	0.90
XLNet	xlm-mlm-en-2048	0.91
LSTM with Attention		0.66
DistilBERT	DistilBERT-base-uncased	**0.92**

TABLE 2: Precision of Four Hate Speech Detection Algorithms and DistilBERT Model on the HSO Dataset

Algorithm	Method Name	Precision
BERT	bert-base-uncased	0.74
RoBERTa	robert-base	0.74
RoBERTa	robert-base-openai-detector	0.72
XLNet	xlm-mlm-en-2048	**0.75**
LSTM with Attention		0.66
DistilBERT	DistilBERT-base-uncased	**0.75**

C. Analysis of Recall

Results from Table IV and Fig. 4 show that DistilBERT and XLNet recorded the average recall score of 75% to demonstrate its superior over other algorithms explored in this study. LSTM with attention had the least recall score of 66%. Although the LSTM with attention performed inferior in our experiments, it should be noted that it performed superior to an earlier study on the same dataset for the task of hate speech detection [29].

D. Analysis of MCC Scores

Table I lists the MCC scores calculated for the overall test tweets selected from the experimental dataset. It can be observed that our proposed method recorded the highest MCC score of 75%. Fig. 5 shows that the difference in MCC scores for all algorithms explored in this study is negligible. The worst performing algorithm was RoBERTa (robert-base- openai-detector) which recorded a MCC score of 71% while the best performing algorithm was DistilBERT (distilbert-base- uncased) which recorded a MCC score of 75%.

TABLE 3: Recall of Four Hate Speech Detection Algorithms and DistilBERT Model on the HSO Dataset

Algorithm	Method Name	Recall
BERT	bert-base-uncased	0.72
RoBERTa	robert-base	0.65
RoBERTa	robert-base-openai-detector	0.63
XLNet	xlm-mlm-en-2048	0.69
LSTM with Attention		0.66
DistilBERT	DistilBERT-base-uncased	**0.75**

TABLE 4: MCC of Four Hate Speech Detection Algorithms and DistilBERT Model on the HSO Dataset

Algorithm	Method Name	MCC
BERT	bert-base-uncased	0.73
RoBERTa	robert-base	0.73
RoBERTa	robert-base-openai-detector	0.71
XLNet	xlm-mlm-en-2048	0.74
LSTM with Attention		0.72
DistilBERT	DistilBERT-base-uncased	**0.75**

E. Analysis of Evaluation Loss

Table VI shows evaluation loss recordings for the experiments carried out in this study. Fig. 6 clearly shows that our proposed method recorded the best (lowest) evaluation loss of 28% while the LSTM with attention recorded the worst evaluation loss of 36%. This shows that our proposed method maximized predictive capability while minimizing the misclassification error rate more than any of the baseline algorithms.

Table 5: Evaluation Loss of Four Hate Speech Detection Algorithms And DistilBERT Model on The HSO Dataset

Algorithm	Method Name	Eval loss
BERT	bert-base-uncased	0.32
RoBERTa	robert-base	0.32
RoBERTa	robert-base-openai-detector	0.33
XLNet	xlm-mlm-en-2048	0.31
LSTM with Attention		0.36
DistilBERT	DistilBERT-base-uncased	**0.28**

F. Analysis of F-Measure

Table VII and Fig. 7 show the F-measure scores of the algorithms explored in this study. It can be observed that the DistilBERT (DistilBERT-base-uncased) recorded the best F- measure score of 75% while LSTM with attention recorded the lowest F-measure score of 66%. Although DistilBERT has fewer layers and parameters, it outperformed all other transformer algorithms explored in this study. The superior performance of DistilBERT may be attributed to the chosen hyperparameters during experimentation. The same hyperparameters were used to train all the models. It can be argued that the used parameters are not necessarily the optimal combination of hyperparameters for each model explored in this study. Careful selection of the best hypeparameters may improve performance of models such as BERT and RoBERTa.

TABLE 6: F-Measure of Four Hate Speech Detection Algorithms and DistilBERT Model on the HSO Dataset

Algorithm	Method Name	F-Measure
BERT	bert-base-uncased	0.73
RoBERTa	robert-base	0.69
RoBERTa	robert-base-openai-detector	0.67
XLNet	xlm-mlm-en-2048	0.72
LSTM with Attention		0.66
DistilBERT	DistilBERT-base-uncased	0.75

Comparative results based on five different metrics from this work show that the transformer models consistently outperform the LSTM with attention. The superior performance of transformer demonstrates that limitations of LSTM, which are inefficient sequence transduction and lengthy processing time have been adequately addressed by the transformer method in hate speech detection.

5. Conclusion

Given the societal implications of hate speech, it is crucial that systems that can accurately distinguish between hate speech, offensive language and neutral speech are developed. Despite concerted efforts from social media companies, governments, and academia, hate speech detection remains a challenging problem in the society of today. In this paper, we have explored several transformer-based methods for hate speech detection. We have evaluated the effectiveness of our method using six state of the art metrics. The results showed that the DistilBERT, a distilled version of BERT, outperforms all transformer-based baseline methods and the attention-based LSTM explored in this study. We, therefore, conclude that the proposed method can be used to learn effective information for the classification of hate speech in resource-constrained environments because it is computationally inexpensive. In addition, transformers facilitate transfer learning, allowing them to be used where training data is limited. It is common for hate speech on social media to be expressed in more than one language.

For example, most people in Africa code switch their native languages with French, Portuguese, or English language. In future work, we plan to explore multilingual pre-trained models for the task of hate speech detection. The data used in this study were limited to textual Twitter texts only, whereas hate speech on Twitter may be expressed through different data formats such as images and videos. For example, a user may post a video inciting hate speech on Twitter and still go undetected. This limitation calls for the development of multimodal datasets that include other formats of data. Future study will develop methods that integrate both textual and image data for hate speech detection.

References

[1] Whillock, Rita Kirk, and David Slayden. *Hate Speech*. Thousand Oaks, CA: Sage Publications, 1995.

[2] MacAvaney, Sean, Hârun Yao, Eugene Yang, Kelly Russell, Nazli Goharian, and Ophir Frieder. "Hate Speech Detection: Challenges and Solutions." *Proceedings of the 2019 ACM SIGIR International Conference on Theory of Information Retrieval* (2019): 89-96.

[3] Pálmadóttir, Védís, and Anastasia Kalenikova. "The Impact of Hate Speech on Social Cohesion." *Journal of Social Issues* 74, no. 2 (2018): 456-472.

[4] Bleich, Erik. "The Rise of Hate Speech Laws in Liberal Democracies." *Journal of Ethnic and Migration Studies* 37, no. 6 (2011): 917-934.

[5] Aljazeera. "Xenophobic Attacks in South Africa." *Aljazeera News*, June 2021.

[6] Schabas, William A. *Hate Speech and the Rwandan Genocide*. Cambridge: Cambridge University Press, 2000.

[7] Schmidt, Anna, and Michael Wiegand. "A Survey on Hate Speech Detection Using Natural Language Processing." *Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media* (2017): 1-10.

[8] Kursuncu, Ugur, Manas Gaur, Carlos Castillo, Amit Sheth, and Viktor L. Shalin. "Modeling the Interplay of Hate and Counter Speech in Online Platforms." *ACM Transactions on Internet Technology* 19, no. 3 (2019): 1-22.

[9] Gaumont, Nicolas, Mehdi Panahi, and David Chavalarias. "A Study of the Social Media Distribution of News Content." *Proceedings of the Twelfth International Conference on Web and Social Media* (2018): 137-146.

[10] Pereira-Kohatsu, Juan Carlos, Luis Quijano-Sanchez, Federico Liberatore, and Mikel Camacho-Collados. "Detecting and Monitoring Hate Speech in Twitter." *Proceedings of the International Conference on Web Engineering* (2019): 256-267.

[11] Davidson, Thomas, Dana Warmsley, Michael W. Macy, and Ingmar Weber. "Automated Hate Speech Detection and the Problem of Offensive Language." *Proceedings of the Eleventh International Conference on Web and Social Media* (2017): 512-515.

[12] Yasseri, Taha, and Bertie Vidgen. "Detecting Hate Speech in Social Media." *Social Media + Society* 5, no. 3 (2019): 1-9.

[13] Pitsilis, Georgios K., Heri Ramampiaro, and Helge Langseth. "Detecting Offensive Language in Tweets Using Deep Learning." *Proceedings of the 2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining* (2018): 100-107.

[14] Miok, K., Q. Wang, S. Wang, X. Zheng, and J. Wang. "Addressing the Stress of Hate Speech Annotation." *Proceedings of the 2020 AAAI Conference on Artificial Intelligence* (2020): 136-142.

[15] Ullmann, Sara, and Marcus Tomalin. "Quarantining Online Hate Speech: Technical and Ethical Perspectives." *Ethics and Information Technology* 22 (2020): 69-80.

[16] Ethnologue Languages of the World. "Number of Living Languages." 2021. https://www.ethnologue.com/statistics/number-of-living-languages.

[17] Young, Tom, Devamanyu Hazarika, Soujanya Poria, and Erik Cambria. "Recent Trends in Deep Learning Based Natural Language Processing." *IEEE Computational Intelligence Magazine* 13, no. 3 (2018): 55-75.

[18] Holmes, David, and Anil K. Jain. *Text Mining: Applications and Theory*. Chichester: John Wiley & Sons, 2006.

[19] Wang, Gang, Shumin Li, and Wei Xu. "Analyzing and Detecting Hate Speech Using Machine Learning: A Survey." *Journal of Big Data* 5 (2018): 12-19.

[20] Gao, Lei, and Ruihong Huang. "Detecting Online Hate Speech Using Context-Aware Models." *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing* (2017): 2559-2569.

[21] Fortuna, Paula, and Sérgio Nunes. "A Survey on Automatic Detection of Hate Speech in Text." *ACM Computing Surveys* 51, no. 4 (2018): 85-89.

[22] N. S. Mullah and W. M. N. W. Zainon, ‘‘Advances in machine learning algorithms for hate speech detection in social media: A review,’’ IEEE Access, vol. 9, pp. 88364–88376, 2021.

[23] X. Sun, D. Yang, X. Li, T. Zhang, Y. Meng, H. Qiu, G. Wang, E. Hovy, and J. Li, ‘‘Interpreting deep learning models in natural language processing: A review,’’ 2021, arXiv:2110.10470.

[24] H. Wang, J. He, X. Zhang, and S. Liu, ‘‘A short text classification method based on N -gram and CNN,’’ Chin. J. Electron., vol. 29, no. 2, pp. 248–254, 2020.

[25] Y. Zhou, J. Li, J. Chi, W. Tang, and Y. Zheng, ‘‘Set-CNN: A text convolutional neural network based on semantic extension for short text classification,’’ Knowl.-Based Syst., vol. 257, Dec. 2022, Art. no. 109948.

[26] J. Xu, Y. Cai, X. Wu, X. Lei, Q. Huang, H.-F. Leung, and Q. Li, ‘‘Incorporating context-relevant concepts into convolutional neural networks for short text classification,’’ Neurocomputing, vol. 386, pp. 42–53, Apr. 2020.

[27] J. Du, C.-M. Vong, and C. L. P. Chen, ‘‘Novel efficient RNN and LSTM- like architectures: Recurrent and gated broad learning systems and their applications for text classification,’’ IEEE Trans. Cybern., vol. 51, no. 3, pp. 1586–1597, Mar. 2021.

[28] W. K. Sari, D. P. Rini, and R. F. Malik, ‘‘Text classification using long short-term memory with glove,’’ Jurnal Ilmiah Teknik Elektro Komputer dan Informatika (JITEKI), vol. 5, no. 2, pp. 85–100, 2019.

[29] M. Shi, K. Wang, and C. Li, ‘‘A C-LSTM with word embedding model for news text classification,’’ in Proc. IEEE/ACIS 18th Int. Conf. Comput. Inf. Sci. (ICIS), Jun. 2019, pp. 253–257.

[30] W. Warner and J. Hirschberg, ‘‘Detecting hate speech on the world wide web,’’ in Proc. 2nd Workshop Lang. Social Media, 2012, pp. 19–26.

[31] R. Alshalan and H. Al-Khalifa, ‘‘A deep learning approach for automatic hate speech detection in the Saudi Twittersphere,’’ Appl. Sci., vol. 10, no. 23, p. 8614, Dec. 2020.

[32] Z. Waseem and D. Hovy, ‘‘Hateful symbols or hateful people? Predictive features for hate speech detection on Twitter,’’ in Proc. NAACL Student Res. Workshop, 2016, pp. 88–93.

[33] N. Vashistha and A. Zubiaga, ‘‘Online multilingual hate speech detection: Experimenting with Hindi and English social media,’’ Information, vol. 12, no. 1, p. 5, Dec. 2020.

[34] S. Khan, M. Fazil, V. K. Sejwal, M. A. Alshara, R. M. Alotaibi, A. Kamal, and A. R. Baig, ‘‘BiCHAT: BiLSTM with deep CNN and hierarchical attention for hate speech detection,’’ J. King Saud Univ. Comput. Inf. Sci., vol. 34, no. 7, pp. 4335–4344, Jul. 2022.

[35] S. Modha, P. Majumder, T. Mandl, and C. Mandalia, ‘‘Detecting and visualizing hate speech in social media: A cyber watchdog for surveillance,’’ Expert Syst. Appl., vol. 161, Dec. 2020, Art. no. 113725.

[36] P. Kapil and A. Ekbal, ‘‘A deep neural network based multi-task learning approach to hate speech detection,’’ Knowl.-Based Syst., vol. 210, Dec. 2020, Art. no. 106458.

[37] F. M. Plaza-del-Arco, M. D. Molina-González, L. A. Ureña-López, and M. T. Martín-Valdivia, ‘‘Comparing pre-trained language models for Spanish hate speech detection,’’ Expert Syst. Appl., vol. 166, Mar. 2021, Art. no. 114120.

[38] Z. Mossie and J.-H. Wang, ‘‘Vulnerable community identification using hate speech detection on social media,’’ Inf. Process. Manage., vol. 57, no. 3, May 2020, Art. no. 102087.

[39] M. S. H. Ameur and H. Aliane, ‘‘AraCOVID19-MFH: Arabic COVID-19 multi-label fake news & hate speech detection dataset,’’ Proc. Comput. Sci., vol. 189, pp. 232–241, Jan. 2021.

[40] A. M. U. D. Khanday, S. T. Rabani, Q. R. Khan, and S. H. Malik, ‘‘Detecting Twitter hate speech in COVID-19 era using machine learning and ensemble learning techniques,’’ Int. J. Inf. Manage. Data Insights, vol. 2, no. 2, Nov. 2022, Art. no. 100120.

[41] G. D. Valle-Cano, L. Quijano-Sánchez, F. Liberatore, and J. Gómez, ‘‘SocialHaterBERT: A dichotomous approach for automatically detecting hate speech on Twitter through textual analysis and user profiles,’’ Expert Syst. Appl., vol. 216, Apr. 2023, Art. no. 119446.

[42] P. Fortuna, J. Soler-Company, and L. Wanner, ‘‘How well do hate speech, toxicity, abusive and offensive language classification models generalize across datasets?’’ Inf. Process. Manage., vol. 58, no. 3, May 2021, Art. no. 102524.

[43] Uysal, A. K. and Gunal, S. 2014. The impact of preprocessing on text classification. Information Processing & Management, 50 (1): 104-112.

[44] Naiyer, Vaseem, Jitendra Sheetlani, and Harsh Pratap Singh. "Software Quality Prediction Using Machine Learning Application." Smart Intelligent Computing

and Applications: Proceedings of the Third International Conference on Smart Computing and Informatics, Volume 2. Springer Singapore, 2020.

[45] Pasha, Shaik Imran, and Harsh Pratap Singh. "A Novel Model Proposal Using Association Rule Based Data Mining Techniques for Indian Stock Market Analysis." Annals of the Romanian Society for Cell Biology (2021): 9394-9399.

[46] Md, Abdul Rasool, Harsh Pratap Singh, and K. Nagi Reddy. "Data Mining Approaches to Identify Spontaneous Homeopathic Syndrome Treatment." Annals of the Romanian Society for Cell Biology (2021): 3275-3286.

[47] Vijay Vasanth, A., et al. "Context-aware spectrum sharing and allocation for multiuser-based 5G cellular networks." Wireless Communications and Mobile Computing 2022 (2022).

[48] Singh, Harsh Pratap, and Rashmi Singh. "Exposure and Avoidance Mechanism Of Black Hole And Jamming Attack In Mobile Ad Hoc Network." International Journal of Computer Science, Engineering and Information Technology 7.1 (2017): 14-22.

[49] Singh, Harsh Pratap, et al. "Design and Implementation of an Algorithm for Mitigating the Congestion in Mobile Ad Hoc Network." International Journal on Emerging Technologies 10.3 (2019): 472-479.

[50] Singh, Harsh Pratap, et al. "Congestion Control in Mobile Ad Hoc Network: A Literature Survey."

[51] Rashmi et al.. "Exposure and Avoidance Mechanism Of Black Hole And Jamming Attack In Mobile Ad Hoc Network." International Journal of Computer Science, Engineering and Information Technology 7.1 (2017): 14-22.

[52] Sharma et al., "Guard against cooperative black hole attack in Mobile Ad-Hoc Network." Harsh Pratap Singh et al./International Journal of Engineering Science and Technology (IJEST) (2011).

[53] Singh, et al., "A mechanism for discovery and prevention of coopeartive black hole attack in mobile ad hoc network using AODV protocol." 2014 International Conference on Electronics and Communication Systems (ICECS). IEEE, 2014.

[54] Harsh et al., "Design and Implementation of an Algorithm for Mitigating the Congestion in Mobile Ad Hoc Network." International Journal on Emerging Technologies 10.3 (2019): 472-479.

[55] Kadhim, A. I. 2018. An Evaluation of Preprocessing Techniques for Text Classification. International Journal of Computer Science and Information Security (IJCSIS), 16 (6).