International Journal of System Dynamics Applications
Volume 11 • Issue 2

BERT-BU12 Hate Speech Detection Using Bidirectional Encoder-Decoder

Shailja Gupta, Manav Rachna University, India
Manpreet Kaur, Manav Rachna University, India
Sachin Lakra, Manav Rachna University, India

ABSTRACT

Transfer learning models have been known to exhibit good results in the area of text classification for question-answering, summarization, and next word prediction, but these learning models have not been extensively used for the problem of hate speech detection. The authors anticipate that these networks may give better results in another task of text classification (i.e., hate speech detection). This paper introduces a novel method of hate speech detection based on the concept of attention networks using the BERT attention model. The authors have conducted exhaustive experiments and evaluation over publicly available datasets using various evaluation metrics (precision, recall, and F1 score). They show that the model outperforms all the state-of-the-art methods by almost 4%. They have also discussed in detail the technical challenges faced during the implementation of the proposed model.

KEYWORDS

Attention Networks, Classification, FusedAdam, Gelu, Hate, Machine Learning, Transfer Learning, Uncased

INTRODUCTION

The right to speak and the right to express oneself freely are two of the various rights provided by the constitution of countries. People have been enjoying these rights by expressing their sentiments, opinions and their feelings with each other. Modern technology provides humans with social networking sites and microblogging sites to understand each other's culture and emotions even while living in various parts of a country or a world. However, people have also started misusing these platforms by trying to oppose the opinions or thoughts of other users by using abusive language, offensive words, and aggressive sentences on these platforms, as part of their communication. These platforms have also been used in recent times by religious groups, political parties and bullies to oppose others and improve their image among the general public for their own interest by posting hateful, offensive and abusive contents to spoil the image of opposing parties or groups. The younger generation which is tech-savvy and has not developed the understanding of worldly ways, are highly affected by reading and viewing such content.

According to statistics related to Hate Crime, (2019), there have been 103,379 hate crimes recorded in the year 2018-19 in England and Wales, where the majority have been race-related (76%), 56% of hate crimes recorded by police have been for public offenses and (36%) have involved violence. 5% of these crimes have been recorded as criminal damage and arson. A campaign advisor of a non-profit organization has reported that 73% of people with learning disabilities and autism have

DOI: 10.4018/IJSDA.20220701.oa4

experienced hate crime. Based on Hate Crime Statistics, (2018), the statistics collected by the FBI reported 7036 hate incidents involving 8646 victims, where 59.6% of hate crime has been reported under the categories of race, ethnicity, and ancestry bias, 0.7% of hate crimes reported have been gender-related, while the contribution of hate crimes against individuals with disabilities has been reported as 2.1%. 2.2% of hate crimes have been found to be related to gender identity, 16.7% of hate crimes have been found to be related to sexual orientation while hate crimes falling into the category of relational border constitute 18.7%.

Social networking sites are also gaining a bad reputation due to the presence of such content. There are many challenges faced in implementing hate speech detection by researchers in the field of developing automated hate speech detection methods, which make it difficult to assess an individual's contribution towards the problem. The reasons for the challenges in the hate speech detection problem are varying definitions of hate speech, limitation of data or content availability for the training and testing of these systems, casual approach for framing of the sentences, lack of grammar correctness, syntactic structure and comparative evaluation among the datasets.

For these reasons, governmental and social networking sites are trying to find solutions for reducing and removing hateful content from these platforms. Deriving from an article of the Council on Foreign Relations & United Nations Strategy and Plan of action on hate speech, (2019), social media agencies are investing hundreds of millions of Euros, along with time, and staff known as content moderators to combat the issue of hate speech detection by manually reviewing content present online and by detecting material that is not fit to be viewed. The basic problem of the detection of hate speech has been the understanding of the definition of hate speech as it can vary from person to person. The authors have attempted to understand the definition of hate speech by understanding its different terminologies.

Hate Speech

Hate can be expressed in many forms. It is difficult to identify if a part of a sentence contains hateful content or not, merely by reading it. The understanding of hate in hateful sentences is important and has been explained by different sources like ILGA, Facebook, YouTube, Twitter and other European countries, which are responsible for maintaining a code of conduct. Twitter, (2019); Nobata et al. (2016) & ILGA, (2016) has termed "hate" as words that incite discrimination, hostility, violence and lead to threatening or direct attack towards a person, people or a group of people based on certain actual or perceived attributes like age, sexual orientation, race, disability, gender, ethnicity, religious affiliations, disease, national origin, veteran status or gender identity. Social networking platforms like Facebook differentiate hateful from not hateful content by allowing content like standup comedy, jokes, lyrics of songs that might be considered as an attempt to express hateful words among others, but perceived as the bad taste of the authors or speakers. The presence of hateful content that criticizes a nation on its views is also considered as non-hateful but if the hateful comments or content are for a certain community or a group of people, it is considered as hate, as stated by YouTube, (2019). In brief as termed by Fortuna & Nunes, (2018), "hate" has specific targets and is used to promote violence or hate. The only purpose of "hate" is to attack or diminish a particular group of people.

Till now the problem of hate speech has been tacked using various deep learning models but still a lot of scope can be seen in improving the performance of the model. The rest of this paper is outlined as follows. Section Related Work discusses the literature survey carried out for the task of hate speech detection. The methodology applied for the task of hate speech detection has been discussed in Methodology Section. The Methodology section discusses the libraries used in the experiments conducted, the pre-processing of the dataset, the fine-tuning of the BERT model for classification followed by training, compilation, and optimization of our model. Section Experiment Conducted elaborates the experimental settings and discusses different comparative models that have been considered while evaluating the proposed model. The experimental results obtained from our proposed model, based on different evaluation metrics on four publicly available datasets have been

elaborated in Error Analysis and Confusion Matrix section. The analysis of errors has been provided using graphs and confusion matrix respectively in Section 6 and the challenges faced during the experiments are explained in Challenges section. The conclusion and future work have been discussed in Conclusion section.

RELATED WORK

Machine learning has been extensively implemented in various domains due to digital transformation that has taken place in recent years. Some of the works by Bouzaida & Sakly, (2018), Majhi, (2018), Panda, (2019), Hirpara & Sharma, (2020), Bhardwaj, (2020), Tikhomirova, (2020) have shown improved results over existing models. These machine learning models have also been used for the task of natural language processing in areas like sentiment analysis (Subramaniyaswamy et.al., (2020)), abuse detection (Founta et al., (2020)), fake news (Zhang & Ghorbani, (2020)) and others.

This section discusses some of the earlier work that has been done in the task of identifying hate speech in online text. The work done by Kwok et al., (2013) includes implementation of the bag of words model to a set of tweets and classifying these tweets as "racist" and "non-racist". The authors have built a balanced dataset of 24,582 tweets considering that these sentences contain parts of words that fall in the category of "hate". The authors have applied Naïve Bayes machine learning algorithm to a balanced dataset and have concluded that the high error rate obtained while training the model is due to the bag of words model, which is inefficient in classifying hate words from sentences. The authors propose that by using the bi-gram model and elaborating hashtags, the task of determining "hate" words in tweets can improve the score obtained in performance metrics. Another significant work in the field of hate speech detection is Davidson et al., (2017), in which the authors have used a dataset consisting 25k tweets labelled as one of hate, offensive and neither. The authors have applied machine learning algorithms like Naïve Bayes, logistic regression (LR), support vector machine (SVM), decision trees and random forest over n-gram features to conclude that SVM and LR with L1 regularization perform better than any other supervised machine learning algorithms. The authors of Badjatiya et al., (2017) have implemented three neural networks (CNN, LSTM and FastText) with features comprising of char-n gram, tf-idf and glove embedding over the dataset consisting of 16k tweets labelled as "racist", "sexist" and "neither". The results obtained by using these neural networks for the task of hate speech detection have clearly indicated that these models lead to improved accuracy values over other baseline methods.

With the researchers working with different datasets for the task of hate speech detection, a model has been required that can solve this problem of handcrafted features and platform specific data. The requirement of a generalized model and to transfer learned information to a new dataset, Rizoiu et al., (2019) has proposed a transfer learning architecture, t-DeepHate for the detection of hate speech in social media. This architecture uses Elmo, a neural network-based embedding for vector representation of every word in a sentence. The embeddings are passed over to Bi-LSTM and max pooling to generate results that outperform all the baseline methods. The advantage of this architecture over other baseline methods is to construct a space where learning from different datasets can be combined to solve the common problem of hate speech detection.

With the need of pre-trained models to re-utilize resources for machine learning tasks, the concept of transfer learning was popularized in 2018. BERT, a pre-trained model proposed by Devlin et al., (2018) has become a new interest area for researchers, especially for the language related tasks, as this model can be fine-tuned for any language specific task like question answering systems, predicting the next word and have shown promising results in solving different problem statements. This section discusses some of the work done by using these pre-trained learning models to find an effective solution for various natural language processing tasks, particularly in hate speech detection.

BERT or Bidirectional Encoder Decoder Representation from the Transformation model has been used in language understanding by Pan et al., (2009). The authors have demonstrated that

transfer learning models outperform any standard machine learning model such as SVM and Logistic Regression models, in the task of natural language processing. The authors of the paper have employed transfer learning to various related tasks and have concluded that these pre-trained models can be fine-tuned with any number of additional layers to create a model for any natural language processing task. The pre-trained models discussed by the authors such as BERT by Devlin et al., (2018), SpanBert by Joshi et al., (2020), RoBert by Liu et al.,(2019), XLNet by Yang et al.,(2019), AlBert by Hinton et al., (2015) and DistilBert by Lan et al., (2019) are known to outperform any state-of-the-art methods in the field of natural language detection. It has been observed that these models have shown significant results in the task of natural language processing. BERT has been used for the task of hate speech detection which is one of the classification tasks of natural language processing in the significant work by Liu et al., (2019, June). The authors of the paper have implemented the LSTM and BERT pre-trained models and findings have shown that the BERT model outperforms the LSTM and the linear models with high margins. The authors have also discussed that for unlabeled datasets, BERT performs insignificantly better than any other model. The work done by Zhu et al., (2019) in the identification of hate speech have contributed to research by using a pre-trained BERT based classifier to find abusive and offensive content in tweets. They have implemented SVM character gram and have found that the BERT pre-trained model performs better than the SVM model with marginal improvement in F1 scores between both the models. Kumar et al., (2019) have implemented BERT and SVM for the hate speech detection task in HASOC 2019. The results obtained from this work clearly indicate that a BERT pretrained model was able to achieve a high recall score even for the datasets with a low number of training samples. They have also discussed that the customization of a pre-trained model can be done according to the problem statements and concluded that the BERT model has been capable of generalizing a problem better than the SVM model.

The model proposed by Mozafari et al., (2019) uses a BERT pre-trained model that has been trained on the English Wikipedia dictionary and Book Corpus. The model uses a 12-layered encoder decoder model with 512 max token length and a SoftMax activation function. The authors have applied the BERT model over standard datasets to identify hate speech. The results obtained in experiments outperform all the existing machine learning and deep learning models (as per our knowledge) by 2-3%.

In this research paper, the authors have tried to provide an improvement over the research work of Mozafari et al., (2019) by tuning the hyperparameters of the model, proposed by the author. The model obtained at the end of experiments shows an increase in the performance metrics by 4%, which, according to our knowledge outperforms the performance of available models till date.

In the next section the methodology used while training the model, will be discussed. The pre-processing steps, the libraries and optimization parameters, that have been used while implementing the models, have been described.

METHODOLOGY

This section discusses the BERT pre-trained model along with the preprocessing of the tweets, the libraries used in the model, conversion of tweets, optimizing step and different arguments used for the classification process on the dataset. BERT or Bidirectional Encoder-Decoder Representation of the transformer is a transfer learning method that reads the text from both the ends of the sentence for a better understanding of the text. It receives the input and gives the output at the other end of the decoder. Encoder architecture is represented using a transformer. Generally, Recurrent Neural Networks have been used widely for the task of Natural Language Processing (NLP) but using transfer learning for NLP has generated efficient results. Transfer learning is based on an encoder-decoder model that uses the concept of transformers. Transformers are combinations of attention (also called multi-head attention), normalization and masked attention in the decoder phase as explained in the work of Devlin et al., (2018).

Table 1. CPU time taken for the pre-processing steps for each dataset

Dataset	CPU time
D1-Hatebase Twitter Dataset, (2017)	4 min 45 sec
D2-Detecting Insults in Social Commentary | Kaggle Dataset, (2012)	2 min 6 sec
D3-Aitor-Garcia-p/hate-speech-Dataset, (2018)	5 min 12 sec
D4-Aggression, and Cyberbullying (TRAC - 1) Dataset, (2018)	7 min 43 sec

The traditional attention models have been using a flat attention structure over the hidden status of RNN while the BERT model uses multiple layers of attention (12 or 24 depending upon the model) and also incorporates multiple attention heads in every layer (i.e. 12 or 16). Since the model weights are not shared between the layers, a single BERT model effectively has up to 24X16=384 different attention mechanisms. A twelve-layer model is easier on resources as it takes a lot of space and memory. Using BERT as a base model, our model has been fine-tuned using the following steps:

Preprocessing: Given a tweet, Preprocessing is a step to normalize the inputs according to the desired format. The steps for preprocessing have been outlined as follows:

1. Supplementing the auxiliary dictionary with the words from the dataset using spacy tokenizer. Snowball Stemmer has been used as it is better than Porter Stemmer as discussed in the article (Stemmers-NLTK).
2. Analysis of existing hashtags, as hashtags carry serious information about emotion. To analyze them the authors of the paper have split them into words.
3. Splitting the hashtags into separate ones and replacing them in the sentences.
4. Removing duplicate letters as in "yeessss". The characters repeating more than three times were removed.
5. Removing all the numbers from the sentences.
6. Tokenization was carried out using spacy_tokenization to analyze the resulting tokens.
7. The process of lemmatization was carried out using spacy_lemmatization to reduce the diversity present in the tokens.

The steps (2- 5) were implemented using regular expressions while steps (6) and (7) used the Spacy library. Using the above-mentioned preprocessing step, Table 1 lists the CPU time taken for the preprocessing of each dataset.

Libraries Used: As part of experiments, the model was fine-tuned for the task of hate speech detection. The libraries that were imported for the classification model are numpy, pandas, BERT Tokenizer, Wordpiece Tokenizer, BERT for pretraining, BERT for pretraining model, BERT model, BERT for masked LM, torch, tensor, BERT for sequence classification, BCE with logits loss, roc curve, and sequential sampler.

Optimizing Phase: For the optimization task of the model, the "adam" optimizer, along with "FP16", was applied to the training dataset. The policy for the cyclic learning rate was used to set the values of the learning rate for all the parameter groups as discussed in the work of Smith, L. N. (2017). For each batch, the policy for the cyclic rate changes the rate of learning for that batch.

Training Phase: Four datasets were considered for experiments. The shape of a complete set along with the training and validation set is listed in Table 2. This table represents the number of tweets considered for training after grouping the sentences by class.

Table 2. The shape of the complete dataset, and the training and validation set

Dataset	Shape (total)	Shape (train)	Shape (valid)
D1	24783,13	18587,13	6196,13
D2	3947, 7	2960, 7	987, 7
D3	10703,7	8027,7	2676,7
D4	12000,8	9000,8	3000,8

Compiling Model: The model was compiled on a TPU enabled machine with a sequence length of 128, a learning rate of 3e-5, and the number of epochs as 2.

In the next section the experiments that were conducted on four publicly available datasets are discussed. In the end of the next section, the results obtained using some of the commonly used evaluation metrics are discussed, and an analysis of errors that were obtained during experiments is presented.

EXPERIMENTS Conducted

In this section the details of the experiments conducted by the authors to determine hate speech in four publicly available datasets are elaborated and the performance of the proposed model has been analyzed. This section discusses (i) the datasets that were used for the experimental setup, (ii) different comparative methods that was used for the task of hate speech detection, (iii) implementation process (iv) parameter tuning and (v) evaluation metrics for the proposed model.

Datasets: For our research work, four publicly available datasets were considered, namely, (i) Hatebase Twitter Dataset, (2017) (ii) Detecting Insults in Social Commentary| Kaggle Dataset, (2012) (iii) Aitor–Garcia–p/hate–speech Dataset (2018) (iv) Aggression and Cyberbullying (TRAC - 1) Dataset, (2018). The detailed information about these datasets is listed in Table 3.

Baseline Model: The works of Mozafari et al., (2019) was considered as a baseline model for the experiments in this research work. For the pre-training of the model, the authors of the paper have used the BERT base pre-trained model consisting of 12 attention heads and 12 transformer blocks to extract embeddings from tweets. BERT, which was used by the authors, was trained on BookCorpus (consisting of 800 million words) and Wikipedia (consisting of 2,500 million words). The pre-trained BERT model has been fine-tuned by applying an additional layer of deep neural networks over the existing model. The fine-tuned model takes the sequence length of tokens as 512 with a learning rate of 2e-5 for the experiments. The drop out probability has been considered as 0.1. For the experiments, the authors have trained the classifier in a batch size of 32 with the number of epochs as 3. The dataset used in this research work include Davidson et al., (2017) and Waseem and Hovy, (2016) datasets for experiments. The authors have used the Google Colaboratory research tool along with a GPU machine.

Comparative Models: The results obtained from the proposed model (Bert-BU_{12}), was compared with the following state-of-the-art methods:

1. The work of Zhang et al., (2018) used Naïve Bayes (NB), Logistic Regression, and Support Vector Machine for the task of text categorization. Naïve Bayes was used by the authors to determine the probability of a label with an assumption that features do not interact with each other. The support vector machine and logistic regression models predict the class

Table 3. Publicly available datasets along with classes

Datasets	Dataset Number	Year	Classes	Labels Provided for the experiment	#Tweets	Origin Source	Language
Hatebase Twitter	D1	2017	Hate Offensive Neither	Hate-4994 Offensive-21309 Neither-5892	24783	Twitter	English
Detecting Insults in Social Commentary | Kaggle	D2	2017	Insulting Non-Insulting	Hate-1052 Neither-2898	3948	Wikipedia Comments	English
Aitor-Garcia-p/hate-speech-Dataset	D3	2018	Hate Non-hate	Hate-1196 Neither-9507	10703	Online forum	English
Aggression and Cyberbullying (TRAC - 1)	D4	2018	Overtly Aggressive Covertly Aggressive Non-aggressive	Hate-4240 Offensive-2708 Neither-5052	12000	Twitter	English

labels based on the scores of the features. Analysis of existing hashtags, as hashtags carry serious information about emotion was also performed. To analyze them, the authors of the paper split them into words.

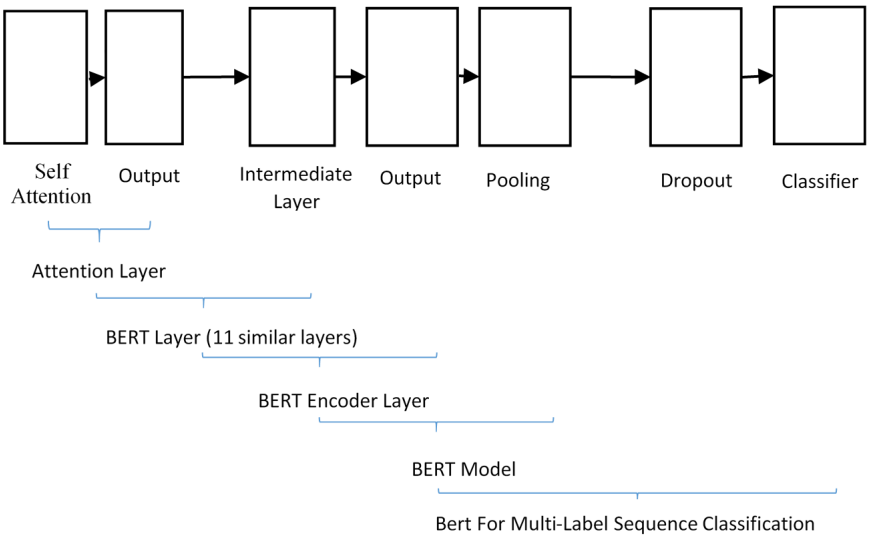
2. A Neural Ensemble method by Zimmerman et al., (2018) combines the decision of convolution neural network consisting of 10 layers of length three with random weight initializations pooled over the entire document length.
3. The BERT pre-trained model by MacAvaney et al., (2019) was used for the classification task where a linear layer was implemented on top of the classification token.
4. Multiview SVM was used by MacAvaney et al., (2019) for the classification task that uses Multiview stacked SVM using features that are fitted over linear SVM.

Implementation, Parameter Tuning and Evaluation Metrics: The model was pre-trained on text passages ignoring labels, lists, and headers. Pre-training provides the embeddings that capture contextual representation among words. These embeddings are used to train the model for the task in hand. The authors used the pytorch pre-trained BERT library with a TensorFlow backend that contained a pre-trained WordPiece dictionary, text tokenizer, and a BERT model. For the implementation, the authors used Google co-lab tool that is available for free for research purposes. The authors have considered 85% of the dataset for the training, 10% for the validation and 5% for the testing of the dataset. Training, validation and test sets were taken by shuffling the original dataset.

Training the model: For the fine-tuning of the BERT model, inputs and outputs were simply fed into the BERT model and all the parameters were fine-tuned. A batch size of 8 with the number of epochs as 2 was used and the best fine-tuning rate of 3e-5 was selected for our experiments. The following is the description of the parameters used during the fine-tuning of the model:

1. Sequence Length: It corresponds to the length of the processed sequence of tokens (words) in the message. In this research work, all Twitter posts that were considered correspond to the sequence length of 128.
2. Batch Size: It corresponds to the model weights that are adjusted after the error function and gradient are calculated. This adjustment of weights was done on each example from the training set, but it is possible to consider it on an average, having counted an average error

Figure 1. The Hate Speech Classification Bert Model: BERT-BU$_{12}$



on some samples, and adjusting afterward. The size of this set was termed as Batch Size. In the experiments, the authors found that a Batch Size of 8 is an optimal size as the model has shown worse results for the Batch Size of less than 8.

3. Learning Rate: It corresponds to how quickly the weights change in the direction of the gradient. Experiments showed best results with a learning rate of 3e-5.
4. Epoch: It corresponds to the number of epochs used during the training of the model. Subsequently, the number of epochs was selected as 2 in the research work, at which the maximum quality was achieved.

Figure 1 represents the hate speech BERT classification model, i.e. BERT-BU$_{12}$, used in this research work, illustrating different layers used for the model. It consists of eleven self-attention BERT layers with linear layers for key, value, and query along with a drop out of 0.1. The output obtained from each layer is in the form of a linear layer with BERT layer normalization and a drop out of 0.1. The intermediate layer is also a linear layer. A pooling layer was applied to the output of the BERT final layer using a linear layer with a sigmoid activation. A drop out of 0.1 on the linear layer provides three labels as output over the dataset. The training was done on a GPU machine. The model was trained with an early stop mode and the best results on the validation sample were saved.

Performance Evaluation Metrics: The authors used Precision (P), Recall (R) and F1 score for the evaluation of results in the research work. The results obtained after conducting experiments is presented in a tabular form in Tables 4-7. The best scores were obtained by the proposed model BERT-BU$_{12}$ and are highlighted in the tables.

RESULTS AND DISCUSSION

The results of the experiments in this research work showed that the BERT model learns better with pre-trained word embeddings. As per our knowledge, this model has shown the best score of

Table 4. Results on HateBase Twitter dataset (D1)

Models	Precision	Recall	F1
SVM	86.6	86.4	86.5
CNN+LSTM	94.2	93.9	94.1
Bert	-	-	89.2
Neural Ensemble	-	-	91.2
LR+L2 regularization	91	90	90
LR+ word-2-vec	91.2	91.2	91.2
LR+ Skip-Thought	89.3	90	75.6
BERT + CNN	92	92	92
BERT-BU$_{12}$	**96.1**	**96.1**	**96.1**

Precision, Recall and F1 score on all the four datasets that were used for the experiments over all the comparative methods.

Tables 4-7 list all the results obtained by the experiments applied on the following comparative models:

1. SVM, CNN+LSTM, LR+word2vec and LR+ skip-thought by Zhang et al., (2018)
2. Neural Ensemble and BERT by MacAvaney et al., (2019)
3. LR+L2 regularization by Yang et al., (2019)
4. BERT base + CNN by Mozafari et al., (2019)

From Table 4, it is clearly seen that there is almost a 4% increase in the Precision, Recall and F1 score over the HateBase twitter dataset (D1) representing the values of P, R and F1 as 96.1%.

The values of performance metrics P, R and F1 were obtained as 87.3% in Table 5 for Detecting Insults in Social Commentary | Kaggle dataset (D2).

In Table 6, BERT-BU$_{12}$ shows an increase of 8% in the performance metrics over the Aitor-garcia-p/hate -speech-dataset (D3).

A huge increase in the evaluation metrics (97.8%) was observed in Table 7 for Aggression and Cyberbullying (TRAC – 1) dataset (D4).

Graphs for loss function and accuracy metrics: The results obtained from the experiments conducted on the publicly available datasets are presented in Figure 2 – a, c, e, g, and show the training and the validation loss values for different epochs and the validation accuracy over the number of epochs using ROC-AUC values are depicted in Figure 2 - b, d, f, h.

Graph Setting: For the training and the validation loss graphs, the epochs are placed on the x-axis while the y-axis represents the loss between actual and predicted values. In the validation accuracy over the number of epochs using ROC-AUC values, the x-axis represents the different epochs

Table 5. Results on detecting insults in social commentary | Kaggle dataset (D2)

Models	Precision	Recall	F1
BERT-BU$_{12}$	**87.3**	**87.3**	**87.3**

Table 6. Results on Aitor-Garcia-p/hate-speech-dataset (D3)

Models	Precision	Recall	F1
BERT	-	-	82.01
Multiview SVM	-	-	80.31
BERT-BU$_{12}$	**90.5**	**90.5**	**90.5**

Table 7. Results on aggression and cyberbullying (TRAC - 1) dataset (D4)

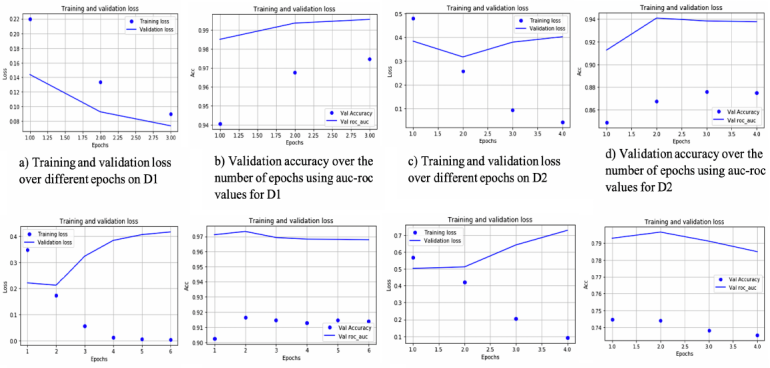
Models	Precision	Recall	F1
Multiview SVM	-	-	53.68
BERT	-	-	52.34
BERT-BU$_{12}$	**97.8**	**97.8**	**97.8**

that were used in the experiments and the y-axis represents the accuracy using the ROC-AUC scores obtained.

Graph Interpretation: From the training and validation loss graph, the loss graphs clearly indicate that the loss during the validation phase decrease at the starting of the epochs while after the second epoch it increases for datasets D2, D3, and D4. The authors suspect that the reason for this sudden increase might be due to the overfitting of the model after '2' epochs. A decrease in training and validation loss in dataset D1 was also observed. The authors suspect that the reason for the increase in validation loss can be due to the reason that the model was built for three classes "hate", "offensive" and "neither" while the other dataset having fewer labels and label names was trained using the same model. As a solution to this un-even number of labels, some standard datasets need to be designed for the problem of hate speech detection.

From the graphs obtained in Figure 2, the authors identified "2" as a suitable limit for the number of epochs for the hate speech classification problem. ROC-AUC or ROC curve was used to find the accuracy of the tests conducted in binary classification models. It uses the concept of true positives

Figure 2. Loss-Accuracy Graphs



a) Training and validation loss over different epochs on D1

b) Validation accuracy over the number of epochs using auc-roc values for D1

c) Training and validation loss over different epochs on D2

d) Validation accuracy over the number of epochs using auc-roc values for D2

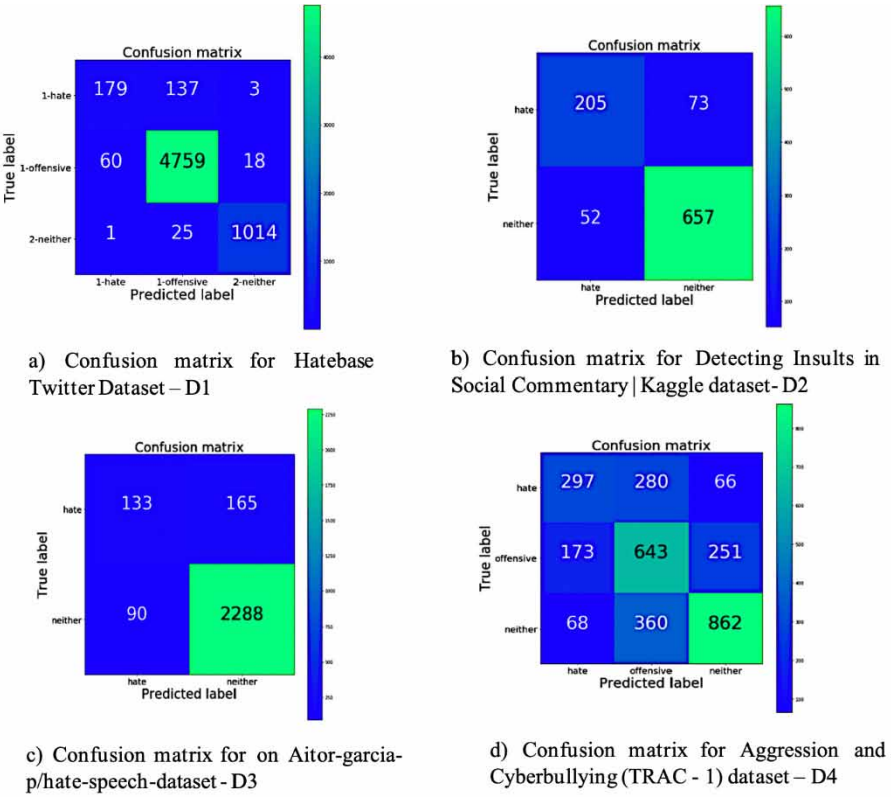
e) Training and validation loss over different epochs on D3

f) Validation accuracy over the number of epochs using auc-roc values for D3

g) Training and validation loss over different epochs on D4

h) Validation accuracy over the number of epochs using auc-roc values for D4

Figure 3. Confusion Matrix



a) Confusion matrix for Hatebase Twitter Dataset – D1

b) Confusion matrix for Detecting Insults in Social Commentary | Kaggle dataset- D2

c) Confusion matrix for on Aitor-garcia-p/hate-speech-dataset - D3

d) Confusion matrix for Aggression and Cyberbullying (TRAC - 1) dataset – D4

and false positives with a value of "0", "0.5" and "1" representing the bad, misclassified and good results, respectively. It was also seen that for dataset D1, the value of ROC reaches near to 1 with two epochs which represents a good classification of labels. Similarly, for datasets D2 and D3, the model shows good accuracy while classifying labels. In dataset D4, the value of ROC falls drastically after two epochs which may be due to overfitting of the data.

Error Analysis And Confusion Matrix

The confusion matrix was used to find true positives, true negatives, false positives and false negatives in the classification problems. Figure 3 illustrates the confusion matrix for datasets D1, D2, D3 and D4.

The results obtained are as follows:

1. In Dataset D1, 76.8% of "offensive" labels and 2.8% of "hate" labels were correctly classified while 2.2% of "hate" labels were misclassified as "offensive".
2. In Dataset D2, 25.32% of "hate labels" were classified correctly while the percentage of misclassified labels was 7.39%. Also, 66.5% of "neither" labels were correctly classified leaving a mere 5.2% as misclassified.
3. In Dataset D3, 85.5% and 4.9% of "neither" and "hate" labels were classified correctly.
4. In Dataset D4, many "offensive" and "neither" labels were classified correctly, i.e., 21.43% and 28.73% respectively while half of "hate" labels, i.e., 9.9% were correctly classified.

As the proposed model was pre-trained on a general corpus, it gained a general knowledge about text data without stressing on "hate", "no-hate", "offensive", "sexism" and so on. It is suspected that the reason for errors was due to misclassification in the dataset as the model was able to classify all the labels by its language. Moreover, some words might be misunderstood by the system as a pre-trained model was used in the experiments and the model might not have been able to understand the context. It is possible that the error occurred due to the bias during the collection of the data. The reason for misclassification can also be due to the bias among the annotators for the creation of rules while annotation and bias present at the time of data collection.

Challenges

In this section, different challenges that were faced while conducting the experiments are discussed.

1. Detecting hate speech is a difficult task as the datasets available were small in size and were extracted mostly from Twitter [49]. The character limitation on online platforms and usage of non-alphabetical characters (hashtags, the short form of words, emojis, etc.) also plays a role in affecting the meaning of the sentence, as the user tries to consolidate users' opinion in a short form of text thereby hindering the system to understand the context of the sentence.
2. Non-availability of curated datasets is another challenge in identifying "hate" and "non-hate" labels from the text. The results generated by the researchers were based on modifications on the datasets which sometimes have not been available publicly thereby hindering researchers to extract datasets for result analysis.
3. An imbalanced dataset, in terms of the number of hate sentences versus non-hate sentences, hinders result generation due to misclassification while executing a model. The authors assume that these imbalances are due to the availability of less data for the "hate" label while extracting tweets from Twitter. The reason for misclassification can be due to different understanding of the annotators for the meanings of labels (hate, no-hate, offensive, etc.).
4. A large number of false positives were generated while executing the model. The authors suspect that the reason for this can be the presence of irrelevant words or misclassification of labels in the datasets.
5. The structure of data is also a challenge for constructing a universal model as the datasets have different numbers and categories of labels. Labels like "sexism" and "racism" have been used by the models for hate speech detection while the authors were unable to find a suitable definition as to which among "sexism" or "racism" to consider as " hate", "offensive" or "no-hate".

CONCLUSION

The authors of the paper have implemented a fine-tuned model, namely, the BERT based model for the performance enhancement on a hate speech detection problem and generalized it on different publicly available datasets. For the implementation of the BERT-BU_{12}, a GPU machine was used instead of a CPU, as the memory requirement is high for the loading and running of the model and the usage of CPU takes a longer time for the execution of the model. The model is customized for the HateBase Twitter dataset, outperforming all the state-of-the-art methods present as per our knowledge. The same model with minor modifications has been executed for other three publicly available datasets and has shown improved results over them as well. Further, it has been emphasized that due to different labels and the number of classes, a universal model for hate speech detection tasks will not be a good choice and the BERT model has to be customized for every single problem. Some of the labels were relabeled for experiments, as the authors were unable to find any written proofs for what can be considered as "hate", "offensive" or "neither" which, the authors also suspect, can be the reason for low evaluation metrics. Although the model outperforms all the state-of-the-art

methods as per our knowledge, it is still felt that by resolving the issue of unbalanced datasets and the absence of sufficient training samples for different classes, a higher performance can be achieved, i.e., a more balanced dataset with large learning samples for each class might produce better results in these instances.

As the contribution of this research work over the baseline work of Mozafari et al., (2019), a jumble solver and snowball stemmer were used as an auxiliary dictionary and stemmer, respectively, for word embeddings. A "gelu" layer was applied to solve the problem of non-linearity of the model. The authors also used categorical cross-entropy for the calculation of loss and used max-pooling and dropout to avoid the problem of overfitting. Fused Adam and FP16 were used in this research work as optimizers with cyclical learning rate class. As a solution to deal with the multilabel classification problem, sigmoid and a softmax function were used. The parametric values of the learning rate, sequence length were modified to produce results that outperform the evaluation parameters by 4% over the baseline method.

Towards the end, the authors have provided the parameters used for the tuning of the model and the specifications of different layers of BERT, which are incorporated in the model. The results that were obtained by the experiment outperform all the state-of-the-art methods proposed (as per our knowledge) for the task of hate speech detection. The authors have also determined the reasons that can be rectified for the performance enhancement of the model. It is also suspected that an understanding of features that are closely related to the problem of hate speech can further improve the existing efficiency of the model.

REFERENCES

Badjatiya, P., Gupta, S., Gupta, M., & Varma, V. (2017, April). Deep learning for hate speech detection in tweets. In *Proceedings of the 26th International Conference on World Wide Web Companion* (pp. 759-760). doi:10.1145/3041021.3054223

Bhardwaj, A. (2020). Health Insurance Claim Prediction Using Artificial Neural Networks. *International Journal of System Dynamics Applications*, *9*(3), 40–57. doi:10.4018/IJSDA.2020070103

Bouzaida, S., & Sakly, A. (2018, April). Adaptive Neuro-Fuzzy Sliding Mode Controller. *International Journal of System Dynamics Applications*, *7*(2), 34–54. doi:10.4018/IJSDA.2018040103

Davidson, T., Warmsley, D., Macy, M., & Weber, I. (2017, May). Hatebase Twitter Dataset. Automated hate speech detection and the problem of offensive language data. In *Proceedings of the 11th International AAAI Conference on Web and Social Media* (pp. 512-515). https://github.com/t-davidson/hate-speech-and-offensive-language

de Gibert, O., Perez, N., Garcia, A., & Cuadros, M. (2018). Hate Speech Dataset from a White Supremacy Forum. In *Proceedings of the 2nd Workshop on abusive Language Online (ALW 2)* (pp. 11–20). Association for Computational Linguistics. doi:10.18653/v1/W18-5102

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Founta, A. M., Chatzakou, D., Kourtellis, N., Blackburn, J., Vakali, A., & Leontiadis, I. (2019, June). A unified deep learning architecture for abuse detection. In *Proceedings of the 10th ACM Conference on Web Science* (pp. 105-114). doi:10.1145/3292522.3326028

Fortuna, P., & Nunes, S. (2018). A survey on automatic detection of hate speech in text. *ACM Computing Surveys*, *51*(4), 1–30. doi:10.1145/3232676

Hate Crime Statistics. (2018). https://www.fbi.gov/news/stories/2018-hate-crime-statistics-released-111219

Hate Crime. (2019). *Hate Crimes Double in Five Years in England and Wales Ben Quinn*. https://www.theguardian.com/society/2019/oct/15/hate-crimes-double-england-wales

Hate Speech on Social Media. (2019). *Global Comparisons*. https://www.cfr.org/backgrounder/hate-speech-social-media-global-comparisons

Hinton, G., Vinyals, O., & Dean, J. (2015). Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Hirpara, R. H., & Sharma, S. N. (2020). An Analysis of a Wind Turbine-Generator System in the Presence of Stochasticity and Fokker-Planck Equations. *International Journal of System Dynamics Applications*, *9*(1), 18–43. doi:10.4018/IJSDA.2020010102

ILGA. (2016). *Hate crime and hate speech*. https://www.ilga-europe.org/what-we-do/our-advocacy-work/hate-crime-hate-speech

Impermium. (2012). *Detecting insults in social commentary dataset*. https://www.kaggle.com/c/ =detecting-insults-in-social-commentary

Joshi, M., Chen, D., Liu, Y., Weld, D. S., Zettlemoyer, L., & Levy, O. (2020). Spanbert: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, *8*, 64–77. doi:10.1162/tacl_a_00300

Kumar, R., & Ojha, A. K. (2019). KMI-Panlingua at *HASOC 2019*. SVM vs BERT for Hate Speech and Offensive Content Detection. In *FIRE* (pp. 285–292). Working Notes.

Kumar, R., Reganti, A. N., Bhatia, A., & Maheshwari, T. (2018, May). Aggression, and Cyberbullying (TRAC-1) Dataset. Aggression-annotated Corpus of Hindi-English Code-mixed Data. *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC)*. https://www.aclweb.org/anthology/W18-4401

Kwok, I., & Wang, Y. (2013, June). Locate the hate: Detecting tweets against blacks. *Twenty-seventh AAAI conference on artificial intelligence.*

Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., & Soricut, R. (2019). Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Liu, P., Li, W., & Zou, L. (2019, June). NULI at SemEval-2019 Task 6: transfer learning for offensive language detection using bidirectional transformers. In *Proceedings of the 13th International Workshop on Semantic Evaluation* (pp. 87-91). doi:10.18653/v1/S19-2011

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., . . . Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

MacAvaney, S., Yao, H. R., Yang, E., Russell, K., Goharian, N., & Frieder, O. (2019). Hate speech detection: Challenges and solutions. *PLoS One*, *14*(8), e0221152. doi:10.1371/journal.pone.0221152 PMID:31430308

Majhi, S. K. (2018). An efficient feed foreword network model with sine cosine algorithm for breast cancer classification. *International Journal of System Dynamics Applications*, *7*(2), 1–14. doi:10.4018/IJSDA.2018040101

Mozafari, M., Farahbakhsh, R., & Crespi, N. (2019, December). A BERT-based transfer learning approach for hate speech detection in online social media. In *International Conference on Complex Networks and Their Applications* (pp. 928-940). Springer.

Nobata, C., Tetreault, J., Thomas, A., Mehdad, Y., & Chang, Y. (2016, April). Abusive language detection in online user content. In *Proceedings of the 25th international conference on world wide web* (pp. 145-153). doi:10.1145/2872427.2883062

Pan, S. J., & Yang, Q. (2009). A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, *22*(10), 1345–1359. doi:10.1109/TKDE.2009.191

Panda, M. (2019). Software defect prediction using hybrid distribution Base balance instance selection and radial basis function classifier. *International Journal of System Dynamics Applications*, *8*(3), 53–75. doi:10.4018/IJSDA.2019070103

Rizoiu, M. A., Wang, T., Ferraro, G., & Suominen, H. (2019). Transfer learning for hate speech detection in social media. *arXiv preprint arXiv:1906.03829*.

Subramaniyaswamy, V., Logesh, R., Abejith, M., Umasankar, S., & Umamakeswari, A. (2020). Sentiment analysis of tweets for estimating criticality and security of events. In *Improving the Safety and Efficiency of Emergency Services: Emerging Tools and Technologies for First Responders* (pp. 293–319). IGI Global. doi:10.4018/978-1-7998-2535-7.ch013

Tikhomirova, O. (2020). Entrepreneurial innovative network and the design of socio-economic neural system. *International Journal of System Dynamics Applications*, *9*(2), 80–102. doi:10.4018/IJSDA.2020040105

Twitter. (2019). *The Twitter Rules*. https://help. twitter.com/en/rules-and-policies/twitter-rules

United Nations Strategy and Plan of action on hate speech. (2019). https://www.un.org/en /genocideprevention/documents/UN%20Strategy%20and%20Plan%20of%20Action%20on%20Hate%20Speech%2018%20June%20SYNOPSIS.pdf

Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R. R., & Le, Q. V. (2019). Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in Neural Information Processing Systems*, 5754–5764.

Youtube. (2019). *Hate speech policy*. https://support.google.com/youtube/answer/2801939?hl=en

Zhang, X., & Ghorbani, A. A. (2020). An overview of online fake news: Characterization, detection, and discussion. *Information Processing & Management*, *57*(2), 102025. doi:10.1016/j.ipm.2019.03.004

Zhang, Z., Robinson, D., & Tepper, J. (2018, June). Detecting hate speech on twitter using a convolution-gru based deep neural network. In *European semantic web conference* (pp. 745-760). Springer.

Zhu, J., Tian, Z., & Kübler, S. (2019). Um-iu@ ling at semeval-2019 task 6: Identifying offensive tweets using bert and svms. *arXiv preprint arXiv:1904.03450.*

Zimmerman, S., Kruschwitz, U., & Fox, C. (2018, May). Improving hate speech detection with deep learning ensembles. *Proceedings of the Eleventh International Conference on Language Resources and Evaluation.*

Shailja Gupta has about 3 years of experience in academics. She is currently pursuing Ph.D in Computer Science from Manav Rachna University. She has received his Master's degree in Computer Science from YMCAUST, Faridabad in 2013. She has subsequently been teaching in the Department of Computer Science and Technology, Manav Rachna University (formerly Manav Rachna College of Engineering (MRCE)), Faridabad, Haryana, India and presently holds the designation of Assistant Professor. Ms. Shailja has research interests in the areas of Natural Language Processing and Entailment.

Manpreet Kaur is currently serving as an Associate Professor in the department of Computer Science and Technology. She has a rich academic experience of working at esteemed institutions for 12 years. She holds her post graduate degree in Computer Science and Engineering from Punjabi University, Punjab. She has submitted her doctoral research work in computer engineering from Netaji Subhas Institute of Technology affiliated to Delhi University, New Delhi. Her research interests lie in the areas of natural language processing, information retrieval, text analytics and machine learning. She has many research papers published in esteemed international journals and conferences along with best paper award to her credit. She is the university coordinator of SWAYAM-NPTEL for promoting massive open online courses (MOOC) among students of university. She has received appreciation from NPTEL, IIT kanpur many times for her instrumental role as SPOC in university. She is also handling the responsibility of IEEE branch counselor of the university and is actively engaged in promoting various technical activities in the campus.

Sachin Lakra has about 14 years of experience in academics and has completed his doctoral degree from Koneru Lakshmaiah University, Andhra Pradesh, India in 2017. He has received his Master's degree in Information Technology from Allahabad Agricultural Institute-Deemed University in 2005. He was then awarded his Master of Philosophy degree in Computer Science from Chaudhary Devi Lal University, Sirsa, in 2008. He has subsequently been teaching in the Department of Computer Science and Technology, Manav Rachna University (formerly Manav Rachna College of Engineering (MRCE)), Faridabad, Haryana, India and presently holds the designation of Associate Professor. Dr. Lakra has research interests in the areas of soft computing, fuzzy theory, speech processing and green computing. He has about 35 research papers to his credit and has authored two books. He was the Co-Convener of the National Conference on Future Trends in Software Development organized in 2008 at MRCE and was the editor of the proceedings of the conference. He is presently guiding 5 PhD scholars. He is a Member of the Institute of Electrical and Electronics Engineers (IEEE) and a Life Member of the Indian Society for Technical Education (ISTE). At present he is an active member of the Research Cluster of Computing, a group of researchers under the aegis of Manav Rachna Research, Innovation, and Incubation Centre.