Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 219 (2023) 1527–1537

Procedia Computer Science

www.elsevier.com/locate/procedia

CENTERIS – International Conference on ENTERprise Information Systems / ProjMAN – International Conference on Project MANagement / HCist – International Conference on Health and Social Care Information Systems and Technologies 2022

Intelligent Identification of Hate Speeches to address the increased rate of Individual Mental Degeneration

Lamima Tabassum Ava[a], Asif Karim[b*], Md. Mehedi Hassan[c], Fahad Faisal[d], Sami Azam[e], A S M Farhan Al Haque[f], Sadika Zaman[g]

[a] *Computer Science and Engineering, Bangladesh University of Business and Technology Dhaka, Bangladesh, 17182103281@cse.bubt.edu.bd*
[b] *College of Engineering, IT and Environment, Charles Darwin University, NT, Australia, asif.karim@cdu.edu.au*
[c] *Department of Computer Science and Engineering, North Western University, Khulna, Bangladesh, mehedihassan@ieee.org*
[d] *Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh fahad.cse@diu.edu.bd*
[e] *College of Engineering, IT and Environment, Charles Darwin University, NT, Australia, sami.azam@cdu.edu.au*
[f] *Department of Electronic Systems, Aalborg University Copenhagen, Denmark, ahaque21@student.aau.dk*
[g] *Department of Computer Science and Engineering, North Western University, Khulna, Bangladesh, sadikazaman@yahoo.com*

Abstract

Hate speech is a public statement that demonstrates resentment or provokes disturbance towards a person or group often based upon race, age, religion, sexual orientation, minority group, psychical disability, political belief, etc. Such an act is a leading cause of mental degeneration in individuals observed throughout the world. We have witnessed an upsurge in the spreading of hateful speech through videos in recent times due to increased social media usage. Researchers are working on this issue because it has become more frequent on several social media platforms, and it leads to low self-esteem and has significant negative impacts on human life. In this work, we focus on collecting data from such videos as nowadays people are sharing numerous videos of this negative nature on platforms like Facebook and YouTube. The audio data of these videos were then converted into text to build the dataset, and we applied some classifier models to our dataset. In this paper, we utilized a transfer learning Bidirectional Encoder Representations from Transformers (BERT) model that gives state-of-the-art outcomes. More precisely, we fine-tuned our model based on transfer learning to evaluate BERT's capacity to capture hostile contexts inside YouTube videos. We examined Fine–Tuning BERT; with different learning rates and listed the outcomes. We train the BERT by freezing all the hyperparameters but with various random seeds to evaluate our suggested Fine-tuning approach. Compared to previous methodologies that used our dataset, our proposition fared better in terms of accuracy and execution time.

* Corresponding author. Tel.: +0-000-000-0000 ; fax: +0-000-000-0000 .
E-mail address: asif.karim@cdu.edu.au

1877-0509 © 2023 The Authors. Published by Elsevier B.V.
This is an open access article under the CC BY-NC-ND license (https://creativecommons.org/licenses/by-nc-nd/4.0)
Peer-review under responsibility of the scientific committee of the CENTERIS – International Conference on ENTERprise Information Systems / ProjMAN - International Conference on Project MANagement / HCist - International Conference on Health and Social Care Information Systems and Technologies 2022
10.1016/j.procs.2023.01.444

Keywords: Hate Speech; MP4 to Text; BERT; NLP; Random Seed; Hyperparameter

1. Introduction

We are human beings; we have the right to express our feelings and share our thoughts. It is our liberty to raise our voices and expressions. But sometimes with our voices and expressions we can also hurt someone, neglect someone's standing, involved in racism, and define gender biasness, state negative religious opinions and so many other things. As a result, freedom of speech is involved with so many negative actions also. Most of the publicly available online communication sources are traceable in some way; most users consider the internet as a medium to express their random opinions anonymously [1].

Numerous paper has been published based on Hate speech such as the work by Marzieh et al. (2020) [2], Xu et al. (2012) [3], Burnap et al. (2015) [4], and this work by Obadimu et al.(2019) [5] detecting hate speech from YouTube [6]comments. Childnet International conducted intensive research on more than 68 countries and came up with the following outcome - anonymous communication is most popular among young people who strongly believe that anonymity on the internet should be preserved, despite its potential adverse effect [1]. The research outcome also indicated that young people seriously suffer from low self-esteem, insecurities, and depression caused by hate speech that they face on the internet. In this paper, we have tried to detect hate speech from YouTube videos as people are watching them for one billion hours every day. According to 2021, YouTube is the second-most prevalent video-sharing platform, and it has over 2.3 billion users globally (Statista, 2021) [7]. There are some relevant papers where the authors applied binary classification (Nobata et al. (2016) [8] for detecting hate speech but our model performs multiclass classification as we created a dataset that contains three different labels. Nowadays researchers are applying machine learning Nurguhoet al. (2019) [9], MacAvaney et al. (2019) [10], Chetty et al. (2018) [11], Abro et al. (2019) [12], Kennedy et al. (2018) [13] and deep learning method Naseem et al. (2019) [14], Founta et al. (2019) [15], Badjatiya et al. (2017) [16], to detect hate speech. Recently NLP models Schmidt et al. (2017) [17], Sajjad et al. (2020) [18], Florio et al. (2020) [19] and transfer learning Abderrouaf et al. (2019) [20], Qasim et al. (2022) [21] transfer learning based models are also used for this issue. We have used a couple of models to detect hate speech including Support Vector Machine (SVM), Random Forest (RF), Naive Bayes (NB), Recurrent Neural Network (RNN), Decision Tree (DT), BERT, BERT-CNN from videos. The remaining paper is organized as follows. Section 2 includes the results and literature from the published paper. The remaining paper is organized as follows. Section 2 includes the results and literature from the published paper. In Section 3, we included our study's methodology, and model architecture that was employed in our work, and in the results analysis part, we examined our findings while Section 4 includes the experimental setting, comprising a brief description of the employed data set and presents an overview of related approaches, as well as theoretical aspects related to the deep learning model used in this study and a discussion about the obtained results. The last Section indicates the conclusions of our work alongside future aspects and limitation of the research.

2. Background

The section presents an overview of related approaches proposed in the literature for detecting hate speech from videos using machine learning models and deep learning models. Our novel approach is processing the videos, creating a dataset, and applying our proposed model for detecting hate speech.

There have been several types of research on computational methods to detect Hate Speech from social media published in the last few years. There is one paper where Schmidt et al. (2017) suggest a method for detecting hate speech in tweets. They utilized Natural Language Processing (NLP) techniques for their dataset that contains three labels: Hate speech, Offensive, and Neither. To categorize the data, the authors utilized a CNN. After preprocessing tweets by Word embeddings, they used the CNN model and achieved 91% accuracy. Biere et al. (2018) [22] used Natural Language Processing (NLP) techniques and proposed a method for detecting hate speech in tweets. The authors used CNN to sort each tweet into three categories: hate, strong language, or neither. The tweets are preprocessed to extract word embeddings before being fed into the CNN model which achieved a 91% accuracy rate. In another recent work by Marzieh et al. (2020) [2]

on Twitter hate speech, the authors used two open datasets one is Waseem dataset and the Davidson dataset. They have utilized the BERT model on these two datasets. They evaluated four strategies for BERT fine-tuning but achieved the best result in BERT+CNN. For the Waseem dataset, the F1 score is 88% and for the Davidson dataset best F1 Score is 92%.

Badjatiya et al. (2017) [16] for detecting hate speech using a deep learning approach where 16K annotated tweets were used for demonstrating that such deep learning methods outperform state-of-the-art char/word n-gram methods by F1 points. They utilized LSTM+RandomEmbedding+GBDT, CNN+Random Embedding, and Char ngram+Logistic Regression. They used 'adam' for CNN and LSTM and 'RMSProp' for FastText as our optimizer. They performed training in batches of size 128 for CNN LSTM and for FastText the batch size was 64. Robison et al. (2018) [23] used SVM for detecting hate speech. They spitted it into 75:25 to use 75% for parameter tuning using 5-fold cross-validation experiments and test the optimized model on the 25% held-out data. For the WZL dataset, they achieved micro F1 74%. They evaluated seven public datasets and reported the outcomes. Malmasi et al (2018) [24] used 10-fold cross-validation on 14,509 data and achieved 77.5% accuracy. They used different feature analysis techniques. Researchers employed RBF kernel SVM Meta classifier and they got impressive results which are 79.8%.Researchers detect hate speech using not only NLP or DL models but also many other ML models as well. In

Recent work with ML algorithms used for comparison, researchers try to improve their performance on specific tasks.Alfina et al. (2017) [35] utilized Naive Bayes, Support Vector Machine, Bayesian Logistic Regression, and Random Forest. Barakat et al. (2013) [26] applied an audio-based Dynamic Time Wrapping (DTW) method to detect offensive words in video blogs. They suggested an audio content-based methodology that employs speaker-independent keyword spotting. They employed the DTW algorithm to compare keyword templates with video audio selections to detect offensive terms. This keyword detection method, which uses voice data to identify individual words spoken, is highly accurate. For improved sentiment classification, they used Nave Bayesian classifier output probability text filtering. The suggested system can obtain an F1 score of 0.66, which is encouraging given that the sentiment classifier can get an F1 score of 0.78 if the input text is devoid of errors. Wu CS et al. (2020) [27] utilized SVM, RF, RNN, and NB models for video datasets. They collected 300 data for YouTube and achieved the best accuracy for RF at 81% for three labels.

3. Methodology

In this section, we describe the entire methodology of this paper. We divided the total process into nine blocks: Data Collection, Conversion of the videos to textual data, then splitting the dataset, applying the models, and finally result analysis and comparison between models.

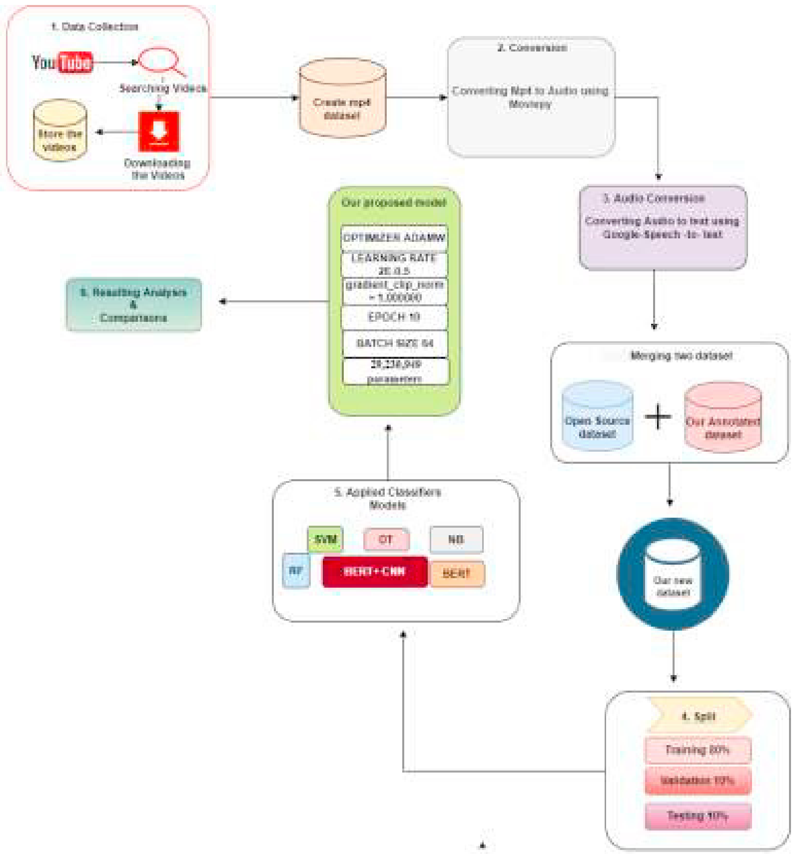

Figure1: The Process diagram of this paper workflow where each step is represented with a block. Block (1): Data collection, Block (2): Video conversion, Block (3): Converting Audio to text Block (4): Split the dataset, Block (5): Applied models, and finally the outcome and comparison section.

Data Collection Process

In this section, the data collection process is described. Several works have been done on YouTube video comments but our focus is not on the comment section of the videos. The focal point of this research is the videos where people used offensive language.

We developed the dataset manually by collecting the videos from the internet and social forums. We employed YouTube as our dominant data source because the data is easily assembled from YouTube. Not every video is required for our dataset, as we aim to find hate speech. So, we searched for videos that contain discrimination, racism, or sexual content. At first, we set the developer key and YouTube API version. Later on, we enabled the YouTube Data API V3 and then utilized the search option. After that, we applied the list method to recover results matching the selected query term. Add each result to the appropriate list, and then depict the lists of video search results. The API delivers the search results by keyword attribute and returns with videos channel name or description and title that retain given keywords. We utilize these keywords for searching the videos “Hijab”, “Black”, “Racism”, “China”, “Rohingya”, and so forth. We specified the path to store the file and downloaded the video file in mp4 format using pytube [34]: a python library used for downloading YouTube videos. Because YouTube API does not support downloading any videos. Thus we create a video dataset.

Video Conversion

The main objective of this study is to detect hate speech from videos; we need to extract text from them. To do so, we must first convert the videos to audio files, and then the audio files to text. So we used Moviepy, a Python library. It reads and exports video and audio files. The FFmpeg API [28] is a multimedia framework for converting media across various formats; already available in Moviepy.

Audio Conversion

We utilize the Google Cloud Speech-to-Text API. The API requires the audio files should be in FLAC or LINEAR format and have only a mono channel. So, we converted the mp4 to FLAC format and converted all stereo channels to the mono channel. Then upload the file to the storage bucket of Google. After that, we can successfully identify speech in the audio file by using the Google Speech-To-Text API. Next, we converted the audio file into text. Eventually, we keep the result in the transcript file and store it in a file. Finally, we created an annotated dataset for our paper.

Dataset Description

For this research, we analyzed a new dataset. We gathered the videos on YouTube and then merged them with an open-source dataset (Davidson et al.). There are 25502 rows and three columns in our new dataset. We divided the data into three categories: hate speech, offensive speech, and neither. 8.7% of the total is hate speech, 73.8% of the total is Offensive language and neither means positive text or neither hate nor offensive language is 17.4% of the total dataset. The description of the dataset is summarized in Table 1.

Table 1: Dataset Description

Name	**Description**
Number of Rows	25502
Number of Columns	3
Class of Dataset	3
Name of labels	Offensive, Hate Speech, Neither
Dimension	25502 by 3
Number of Videos	50

Data Pre-processing

We collected noisy data from videos with numbers, hashtags, URLs, and whitespace [2]. So we removed all the numbers, URLs, and Hashtags. For example “#racist” to “racist” and “mlew1w227” to “mleww”. We converted all our text to lowercase. We removed all punctuation marks, unknown Unicode, and unnecessary delimiting characters.

However, we kept all the stop words because our model BERT has deep bidirectional representations meaning the model learns information from left to right and from right to left; trains the order of words in a text. e.g.,“@If y1‘ou’re in a hurry, get the hell out, okay?” to “if you’re in a hurry, get the hell out, okay?”

Splitting Dataset

Splitting the dataset is the next and last stage before training. In this section, we describe the total number of data used for training, testing, and validation for hate speech detection. We split the whole dataset into the Train, Test, and Validation dataset: 80% for training, 10% of the total for validation, and for testing 10% respectively. The description of the whole splitting dataset is summed up in table 2.

Table 2: The split dataset's summary (Training, Validation, and Testing).

Category	Labels	Data Type	Number of dataset
Hate Speech	0	Train Validation Test	1161 329 343
Offensive	1	Train Validation Test	15000 1730 2000
Neither	2	Train Validation Test	4159 343 419

Modeling Dataset

In this section, we represent the proposed model architecture. How it works on our dataset and Figure 3 depicts the architecture of our proposed model.

Our proposed Model (BERT+CNN)

In this paper, we have utilized BERT for detecting hate speech from videos. BERT Devlin et al. (2018) [34], were previously trained with NSP (next sentence prediction) and MLM (masked language modeling). We obligation to examine the contextual data retrieved from BERT's previously trained layers and fine-tune it by employing expounded datasets since the BERT model is previously trained on generic corpora. We regenerate weights utilizing a labeled dataset that is already a trained model by fine-tuning. BERT acknowledges a series of tokens with a supreme length of 512 as an input and an output 768-dimensional vector representation of the sequence [2]. To accomplish the hate speech classification function, we employ the BERTbase model to classify individual text in our datasets as Offensive, Hate speech, or neither. To do so, our focal point on fine-tuning the BERTbase parameters that have already been trained. Fine-tuning requires training a classifier with multiple layers of 768 dimensions on top of the BERTbase transformer to diminish task-specific parameters.

We aim to have the best accuracy for our dataset. BERT is a powerful model but we target to achieve good performance from the BERT. So, we applied the Novel Fine-Tuning Strategies and proposed a model that gives the best performance on our dataset.

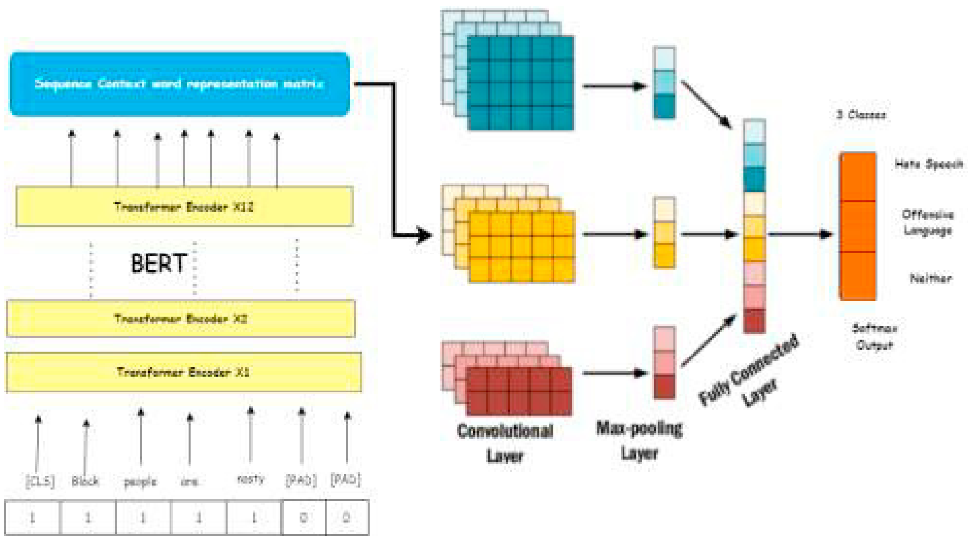

Figure 2: The architecture diagram of our proposed model for detecting hate speech.

The Strategies of Fine Tuning

To improve hate speech identification from videos, we present a transfer learning strategy based on the pre-trained language model BERT learned on English Wikipedia and Book Corpus. To that aim, we offer novel fine-tuning methodologies to BERT with CNN for the classification tasks. For the text categorization task, we must choose the most effective layer. Fine-tuning is an approach where transfer learning is employed or implemented. Fine-tuning tweaks the previously trained model for one given task to accomplish a second similar task. A neural network's layers can assemble multiple levels of syntactic and denotative data. While the higher layers contain task-specific data, The BERT model's bottom layers may include more generic data [2], and we can apply them to fine-tune strategies with diverse rates of learning. Four alternative fine-tuning techniques are utilized in our classification task. All of these rely on pre-trained BERTbase transformer encoders. More information on the architectures of these transformer encoders is provided. As the fine-tuning step, the model is configured with the previously trained parameters and subsequently fine-tuned utilizing the labeled dataset.

Evaluation Methodology

For implementing our proposed model, we have utilized the Google Colab with Tesla t4 GPU as the implementation environment. We trained our model with ten epochs and a batch size of 32. As it includes invalid punctuation characters, hashtags, and lowercasing words, so we tokenized each text with the BERT tokenizer. The dropout probability is set to 0.01 for all layers. We employed the Adam Weight Decay optimizer with a learning rate of 5e-5. The total parameters of our model are 29,230,949, whereas the trainable parameters were 292, 309 and 48. The gradient clip norm for our model is 1.000000, and the random seed is 42. We initialized the weights for fine-tuning. Weight for class 0: 5.82, weight for class 1:0.43, weight for class 2: 1.91. For emanating overfitting, we are assuming 80% of each dataset as training data 10% as validation data to estimate the out-of-sample implementation of the model during training, and 10% as test data to estimate the out-of-sample execution after training. The evaluation outcome is listed based on the model Accuracy [32], Recall [25], F1 score [29], and Precision [25].

Result Analysis

In our experiments, to implement our dataset, we have used some classifier models: NB, SVM, RF, DT, RNN, and BERT. We compared the performance of each model. We used the Google Colaboratory tool as the implementation environment, a free research tool with 12 GB RAM. We used the machine learning classifier model MNB, SVM, RF, and vectorization with count vector, tf-idf, n-gram, and hash vector. Count vector + tf-idf is a technique employed to uncover the meaning of any sentences that consist of words. The required outcomes are shown in Table 3. Also, BERT and our proposed model's outcomes are shown in the table.

Table 3. Performance comparison of Classifier models and proposed model on our preprocessed dataset.

Model	Precision	Recall	F1 Score	Accuracy
Naive Bayes with Count Vectors +TF-IDF	0.78	0.78	0.71	0.79
Naive Bayes with N-Gram Vectors	0.77	0.78	0.70	0.77
Naive Bayes with Hash Vectors	0.85	0.86	0.85	0.85
Linear SVM with Count Vectors +TF-IDF	0.88	0.89	0.88	0.89
Linear SVM with N-Gram Vectors	0.88	0.88	0.84	0.85
Linear SVM with Hash Vectors	0.85	0.86	0.85	0.86
Random Forest with Count Vectors +TF-IDF	0.87	0.88	0.86	0.88
Random Forest with N-Gram Vectors	0.83	0.85	0.82	0.85
Random Forest with Hash Vectors	0.85	0.87	0.85	0.87
Decision Tree with Count Vectors +TF-IDF	0.87	0.88	0.88	0.88
Decision Tree with N-Gram Vectors	0.88	0.89	0.88	0.88
Decision Tree with Hash Vectors	0.87	0.88	0.87	0.88
BERT	0.81	0.79	0.79	0.81
Our proposed Model	0.91	0.91	0.92	0.92

4. Experimental Analysis

This section presents the dataset we have used in our experiments and a discussion of the obtained results. We experiment with our proposed model with different learning rates and random seeds.

Different Learning Rates

The learning rate is a hyper-parameter that defines how much we alter our network's weights in proportion to the loss gradient. The learning rate is a parameter that determines how quickly the model adapts to the situation. Given the minor changes to the weights of each update, lesser learning rates necessitate more training epochs, whereas massive learning rates need fewer training epochs. Devlin et al. (2018) [29] in this paper, authors recommend fine-tuning for 4 epochs over the following hyperparameter options: 8, 16, 32, 64, 128 batch size and learning rates: 3e-4, 1e-4, 5e-5, 3e-5. So we examined this learning rates performance over our dataset where the batch size remains the same at 32 and the Optimizer is ADAMW. We achieve the best accuracy and less error for learning rate 5e-5. The learning rate impacts how fast our model can connect to local minima. It also takes less time to execute 285 seconds per epoch and 435ms/per step.

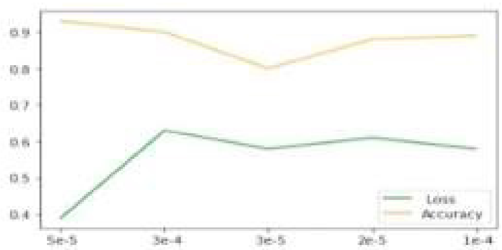

Figure 3: Performance comparison based on proposed Model accuracy and loss matrix by applying different learning rates for our dataset.

Different Random seeds

The most obvious repercussions of COVID-19 include an economic slowdown, a crisis in global governance, trade policy, and a surge in isolationism. There have been limitations on international travel, cultural exchanges, and interpersonal interactions. However, this is only the start [30]. People are spending more time watching videos on social networking sites because they can't go outside. Social networking websites are used by the majority of young people, including students. Students are the nation's future leaders and assets. However, substance misuse is causing a rise in student dropout rates today. Drug addiction is primarily brought on by depression [31]. Depression is a serious medical illness that can arise for a variety of reasons [33]. Hate speech is a contributor to depression and suicidal thoughts. So, detecting hate speech from social media platforms is crucial. For experimental purposes, we examined our proposed model with different random seeds. We evaluate fine-tuning the BERTbase model with CNN for classification tasks where the model detects hate speech or vulgar language. The random seed used to initialize the weights and choose the training data sequence can significantly impact how well a fine-tuned pre-trained language model like BERT performs. In figure (a) we just fine-tuned BERT with random seed 42, batch size 32, and drop out 0.01. On the contrary, in figure (b) we evaluate our same model with the different random seeds which is 2018 and we freeze all hypermeters. The difference between this two is very clearly depicted in the confusion metrics.

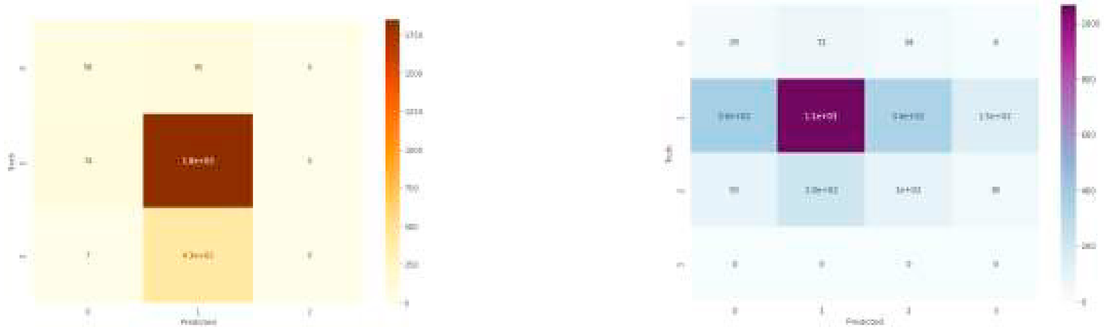

Figure 4. Confusion matrix of our proposed model with different random seeds and hypermeters (a) BERT+CNN (random seed=42), (b) BERT+CNN (random seed=2018) + Freeze all the hypermeters.

Table 4: Our proposed model with different random seeds. All performances are listed based on model loss, accuracy, and execution time.

Epoch	Random Seed	Hyper parameter	GPU(s)	Loss	Accuracy
10	42	Unfreeze	1504s	0.48	0.92
10	2018	Freeze all the layers	1524	1.21	0.76

Discussion and Comparison to Related Work

In this paper, we created our own dataset and proposed a model that gives the best performance on our dataset. In this section, we compare all the models used in this with our proposed model. Then compare the dataset to existing datasets and compare the proposed model with existing models.

Comparison with other models

We evaluate the model performance based on their Evaluation Metrics- Accuracy, F1 score, Recall, and precision (Table-4). Finally, this score considers both false positives and false negatives. Figure 5, shows that the highest F1 score is for our proposed model (BERT-CNN). The second highest is the RF model's F1 score. After that, SVM, NB, and RNN models F1 scores, respectively

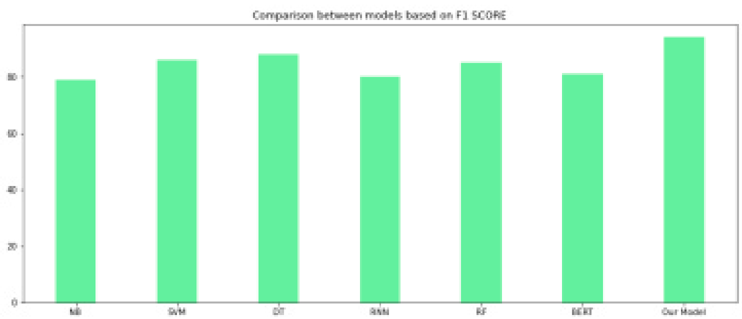

Figure-5: Comparison between models performance (F1 score).

Comparison with existing Datasets

In this section, we compare our dataset with some familiar datasets that are frequently used in many studies which were mentioned in the literature review section. This dataset was used for detecting hate speech.

Table 5: Comparison between existing dataset to our dataset.

Dataset	Total number of dataset	Label/Class	Source
Waseem And Hovey et al.	16000	Racism, Sexism, Neither	Twitter
Davidson et al.	24000	Hate speech, Offensive Language and Neither	Twitter
Ethos	1431	Hate speech Non Hate speech	YouTube comments
Our dataset	25502	Hate speech, Offensive Language and Neither	YouTube videos

Comparison with existing Models

In this section, we compare our proposed model with models that frequently used many studies which were mentioned in the literature review section.

Table 6: Comparison between existing models to our model.

Model	Optimizer	Epoch	Batch size	Drop-out	Learning rates	Random Seeds
Existing model	Adam	3	32	0.1	2e-5	-
Our proposed model	AdamW	10	64	0.01	2e-0.5	42

5. Conclusion

In this paper, we have provided an empirical evaluation of hate speech detection using our proposed model from the videos. Nowadays people are sharing videos every day. So, detecting hate speech from videos is very crucial for our society to prevent cyberbullying. Our paper would be the first step in detecting hate speech from the videos. Not only that, we will implement the advanced techniques and conduct the relevant test with low-resolution videos as well from the live stream. It will help us to identify the real person related to these kinds of hate speeches. Finally, our research will be a significant contribution to individuals who are interested in videos and detecting hate speech on social media sites.

References

[1] I. Grattagliano, A. Rossi, E. Marconi and F. Lapi “Determinants of hcv related complications in italian primary care patients,” Liver Intl., 2021.

[2] M. Mozafari, R. Farahbakhsh, and N. Crespi, “Hate speech detection and racial bias mitigation in social media based on bert model,” PloS one, vol. 15, no. 8, p. e0237861, 2020.

[3] X. Cheng, C. Dale, and J. Liu, “Statistics and social network of youtube videos,” in 2008 16th Intl. Workshop on Quality of Service. IEEE, 2008.

[4] P. Burnap and M. L. Williams, “Cyber hate speech on twitter: An application of machine classification and statistical modeling for policy and decision making,” Policy & internet, vol. 7, no. 2, pp. 223–242, 2015.

[5] A. Obadimu, E. Mead, M. N. Hussain, and N. Agarwal, “Identifying toxicity within youtube video comment,” in International Conference on Social Computing, Behavioral-Cultural Modeling and Prediction and Behavior Representation in Modeling and Simulation. Springer, 2019, pp. 214–223.

[6] G. S. Kalra, R. S. Kathuria, and A. Kumar, “Youtube video classification based on title and description text,” in 2019 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS). IEEE, 2019, pp. 74–79.

[7] R. B. Jackson, C. Le Quer´e,´ R. Andrew, J. G. Canadell, J. I. Korsbakken, Z. Liu, G. P. Peters, and B. Zheng, “Global energy growth is outpacing decarbonization,” Environmental Research Letters, vol. 13, no. 12, p. 120401, 2018.

[8] C. Nobata, J. Tetreault, A. Thomas, Y. Mehdad, and Y. Chang, “Abusive language detection in online user content,” in Proceedings of the 25th international conference on world wide web, 2016, pp. 145–153.

[9] K. Nugroho, E. Noersasongko, A. Z. Fanani, R. S. Basuki et al., “Im-proving random forest method to detect hatespeech and offensive word,” in 2019 International Conference on Information and Communications Technology (ICOIACT). IEEE, 2019.

[10] S. MacAvaney, H.-R. Yao, E. Yang, K. Russell, N. Goharian, and O. Frieder, “Hate speech detection: Challenges and solutions,” PloS one, vol. 14, no. 8, p. e0221152, 2019.

[11] N. Chetty and S. Alathur, "Hate speech review in the context of online social networks," Aggression and violent behavior, vol. 40, pp. 108– 118, 2018.

[12] S. Abro, Z. S. Shaikh, S. Khan, G. Mujtaba, and Z. H. Khand, "Au-tomatic hate speech detection using machine learning: A comparative study," Machine Learning, vol. 10, no. 6, 2020.

[13] B. Kennedy, M. Atari, A. M. Davani, L. Yeh, A. Omrani, Y. Kim,K. Coombs, S. Havaldar, G. Portillo-Wightman, E. Gonzalez et al., "The gab hate corpus: A collection of 27k posts annotated for hate speech," 2018.

[14] U. Naseem, I. Razzak, and I. A. Hameed, "Deep context-aware embed-ding for abusive and hate speech detection on twitter." Aust. J. Intell. Inf. Process. Syst., vol. 15, no. 3, pp. 69–76, 2019.

[15] A. M. Founta, D. Chatzakou, N. Kourtellis, J. Blackburn, A. Vakali, andI. Leontiadis, "A unified deep learning architecture for abuse detection," in Proceedings of the 10th ACM conference on web science, 2019, pp. 105–114.

[16] P. Badjatiya, S. Gupta, M. Gupta, and V. Varma, "Deep learning for hate speech detection in tweets," in Proceedings of the 26th international conference on World Wide Web companion, 2017, pp. 759–760.

[17] A. Schmidt and M. Wiegand, "A survey on hate speech detection using natural language processing," in Proceedings of the fifth international workshop on natural language processing for social media, 2017, pp. 1–10.

[18] M. Sajjad, F. Zulifqar, M. U. G. Khan, and M. Azeem, "Hate speech detection using fusion approach," in 2019 International Conference on Applied and Engineering Mathematics (ICAEM). IEEE, 2019, pp. 251–255.

[19] K. Florio, V. Basile, M. Polignano, P. Basile, and V. Patti, "Time of your hate: The challenge of time in hate speech detection on social media," Applied Sciences, vol. 10, no. 12, p. 4180, 2020.

[20] C. Abderrouaf and M. Oussalah, "On online hate speech detection. effects of negated data construction," in 2019 IEEE International Conference on Big Data (Big Data). IEEE, 2019, pp. 5595–5602.

[21] R. Qasim, W. H. Bangyal, M. A. Alqarni, and A. Ali Almazroi, "A fine-tuned bert-based transfer learning approach for text classification," Journal of Healthcare Engineering, vol. 2022, 2022.

[22] S. Biere, S. Bhulai, and M. B. Analytics, "Hate speech detection using natural language processing techniques," Master Business AnalyticsDe-partment of Mathematics Faculty of Science, 2018.

[23] D. Robinson, Z. Zhang, and J. Tepper, "Hate speech detection on twitter: Feature engineering vs feature selection," in European Semantic Web Conference. Springer, 2018, pp. 46–49.

[24] S. Malmasi and M. Zampieri, "Challenges in discriminating profanity from hate speech," Journal of Experimental & Theoretical Artificial Intelligence, vol. 30, no. 2, pp. 187–202, 2018.

[25] C. Liang, B. Shanmugam, S. Azam, A. Karim, A. Islam, M. Zamani, et al., Intrusion detection system for the internet of things based on blockchain and multi-agent systems, Electronics. 9 (2020) 1120. doi:10.3390/electronics9071120.

[26] M. Barakat, C. Ritz, and D. Stirling, "Temporal sentiment detection for user generated video product reviews," in 2013 13th International Symposium on Communications and Information Technologies (ISCIT). IEEE, 2013, pp. 580–584.

[27] C. S. Wu and U. Bhandary, "Detection of hate speech in videos using machine learning," in 2020 International Conference on Computational Science and Computational Intelligence (CSCI). IEEE, 2020, pp. 585– 590.

[28] K. Srilakshmi, S. S. Ch, S. Jagannath, and A. N. PS, "Video summarisa-tion using shot boundary detection and tf-idf vectorization," in 2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS). IEEE, 2021, pp. 977–983.

[29] F.M. Shamrat, S. Azam, A. Karim, R. Islam, Z. Tasnim, P. Ghosh, et al., Lungnet22: A fine-tuned model for multiclass classification and prediction of lung disease using X-ray images, Journal of Personalized Medicine. 12 (2022) 680. doi:10.3390/jpm12050680.

[30] Smartphone-Based Expert System for COVID-19 Risk Prediction at Early Stage. Bioengineering. 2022 Jul;9(7):281.

[31] Hassan MM, Peya ZJ, Zaman S, Angon JH, Keya AI, Dulla AU. A Machine Learning Approach to Identify the Correlation and Association among the Students' Drug Addict Behavior. In2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT) 2020 Jul 1 (pp. 1-5). IEEE.

[32] A. I. Chowdhury, M. M. Shahriar, A. Islam, E. Ahmed, A. Karim, M. R. Islam, An automated system in ATM booth using face encoding and emotion recognition process, 2020 2nd International Conference on Image Processing and Machine Vision. (2020). doi:10.1145/3421558.3421567.

[33] M. M. Hassan, M. A. R. Khan, K. K. Islam, M. M. Hassan and M. M. F. Rabbi, "Depression Detection system with Statistical Analysis and Data Mining Approaches," 2021 International Conference on Science & Contemporary Technologies (ICSCT), 2021, pp. 1-6, doi: 10.1109/ICSCT53883.2021.964255.

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

[35] I. Alfina, R. Mulia, M. I. Fanany, and Y. Ekanata, "Hate speech detection in the indonesian language: A dataset and preliminary study," in 2017 International Conference on Advanced Computer Science and Information Systems (ICACSIS). IEEE, 2017.