Journal of Computer Science

Original Research Paper

Scalable and Advanced Framework for Hate Speech Detection on Social Media Using BERT and GPT-2 through Encoder and Decoder Architectures

[1]Usman, [2]Nabeela Hasan and [1]Syed Mohammad Khurshid Quadri

[1]Department of Computer Science, Jamia Millia Islamia, New Delhi, India
[2]Department of Artificial Intelligence and Machine Learning, New Delhi Institute of Management, New Delhi, India

Article history
Received: 24-07-2024
Revised: 24-09-2024
Accepted: 17-10-2024

Corresponding Author:
Usman
Department of Computer Science, Jamia Millia Islamia, Delhi, India
Email: usman.mca.du.2014@gmail.com

Abstract: Hate speech is a major problem on social media platforms. Every day, numerous instances of hateful behavior based on race, ethnicity, religion, or gender are witnessed on social media. Most of the leading social media platforms like Instagram, Facebook, Twitter Reddit, etc., have strong community guidelines that condemn and restrict the exchange of hateful language/content in any form. Despite the guidelines, some of these instances go unnoticed due to the suitability of the language and expression. This encourages the need for strong automated hate speech detection techniques that can flag such content and ensure a safer environment for users belonging to all domains of life. There is a concept of transformers model it is based on two encoders and decoder blocks. The model which has only an encoder block is called Bidirectional Encoder Representations from Transformers (BERT) and the model which contains only a decoder block is called Generative Pre-trained Transformer (GPT). In this study, we propose a method that uses a pre-trained BERT model for hate speech detection on Twitter data. The dataset contains tweets belonging to three different classes i.e., hate speech (0), offensive language (1), and neither of these (2). We evaluated our proposed model on this dataset: Without data augmentation and with data augmentation using Generative Pre-trained Transformer-2 (GPT-2). It shows that data augmentation with GPT-2 enhances the performance of the BERT model by achieving 81% accuracy in comparison to un-augmented data. Despite strong community guidelines, subtle forms of hate speech on social media often go undetected, highlighting the need for robust detection methods. The suggested method uses a pre-trained BERT algorithm to categorize tweets as hate speech, inflammatory language, or neutral content. Data augmentation with GPT-2 considerably improves the BERT model's performance, obtaining an 81% accuracy rate.

Keywords: BERT, GPT-2, Hate Speech, Machine Learning, Social Media

Introduction

The last decade has witnessed tremendous advancement in social media communication, with the introduction and meteoric rise of platforms such as Facebook, X (previously Twitter), Instagram, and Reddit, among others. With the availability of instant messaging, microblogging, video sharing, and other communication tools, individuals are increasingly turning to these platforms for communication. Traditional ways of communication, such as letter writing and telephone conversations, are now seldom used in official settings (Hasan and Chaudhary, 2024; Hasan and Alam, 2023). The growing reliance on communication and social contact has given rise to previously unseen and reluctantly voiced activities. Hate speech in the context of digital communication is a severe issue that these social media platforms face daily.

Hate is a serious issue in the social media domain or internet world. There are several available methods proposed for automatic hate speech detection. However, there are still certain limitations in the detection of hate speech on social media platforms. Most of these limitations stem from linguistic intricacies. When it

comes to the definition of "hate speech", there is no universal definition of hate speech. Both in academic and socio-political-judicial perspectives, hate speech has a range of connotations. United Nations, the biggest flagbearer of human rights in the world, defines hate speech as, "any kind of communication in speech, writing or behavior, that attacks or uses pejorative or discriminatory language with reference to a person or a group on the basis of who they are, in other words, based on their religion, ethnicity, nationality, race, color, descent, gender or other identity factor (Nations, 2023)." On the other hand, the Encyclopedia of American Constitution defines hate speech as "…speech that attacks a person or group on the basis of attributes such as race, religion, ethnic origin, national origin, sex, disability, sexual orientation, or gender identity (Nockleby, 2000)." Besides, different social media platforms, social network sites like Facebook, X (previously Twitter), Reddit YouTube, etc., have platform-specific definitions and community guidelines for hate speech (MacAvaney *et al*., 2019; Jahan and Oussalah, 2023).

Hate speech recognition has grown into a popular subject in recent years, particularly with the rise of social networking platforms that enable destructive words to spread quickly. Despite adopting a variety of machine learning and deep learning techniques to address this issue, existing systems face several problems and can perform badly in reality.

There is a big headache: Contextual awareness. A lot of hate speech hides meaning between layers of sarcasm, irony, or hidden meaning when the offensive content is disguised and not directly expressed (Fortuna and Nunes, 2019). Adding to this, human language is messy and inherently ambiguous and noisy, with undocumented words or changing slang (Zampieri *et al*., 2019) this makes it even harder for a system to accurately detect.

Bias in training data: The Case of OCR & more, however, bias in the data on which machine learning models are trained is often unwittingly reflected by the model, meaning that it will detect some groups or communities as a higher risk than others and vice versa. This leads to a problem of either over or under-flagging benign content while missing hate speech from certain marginalized communities (Davidson *et al*., 2017). Additionally, the hate speech detection models cannot be applied to other languages or cultures effectively due to their rigidity. And many of the commercially available solutions are developed on a single language or region data, which makes it hard for these systems to perform within a multilingual context, or for cross-cultural audiences.

Furthermore, fine-grained classification is often challenging for current methods. This plays directly into the censorialism of our age since these definitions of hate speech are so vague that it can sometimes be next to impossible for any automated or even semi-automated system to separate them from merely offensive language, or bland content which importantly is being expressed in similar language as hate speech if it were coming up against. These limitations point to the continuing imperative of enhancing accuracy and fairness on hate speech detection also in terms of:

a Getting better at understanding language and social dynamics and
b Tracking the ever-evolving nature of technologies that fuel this conflict in society

Background

Hate Speech Detection

Hate speech detection on social media is an important undertaking that aims to reduce the negative effect of inflammatory information online. Given the wide and diversified nature of social media sites, this identification requires powerful Natural Language Processing (NLP) and machine learning approaches. Algorithms are taught to spot hate speech patterns, focusing on information that encourages discrimination, violence, or prejudice based on characteristics like race, ethnicity, religion, or gender.

GPT2

GPT-2 offers a significant leap in NLP by including a complex language model. GPT-2, developed by OpenAI, is based on the Transformer architecture and has a large neural network with 1.5 billion parameters. The model does pre-training on a wide range of materials, capturing sophisticated linguistic patterns and structures.

Notable is GPT-2's ability to generate cohesive and contextually appropriate text. It uses an autoregressive technique to anticipate the next word in a series based on the prior context. The model's ability to perform a wide range of linguistic tasks, from text completion to narrative synthesis, demonstrates its complex knowledge of context and semantics.

GPT-2's flexibility originates from its capacity to do zero-shot and few-shot learning, demonstrating its adaptability to tasks that lack task-specific training data. However, its sheer vastness necessitates extensive computer resources for training and inference. GPT-2 exemplifies the changing environment of large-scale language models, encouraging more research into language comprehension and creation.

BERT

BERT (Kenton *et al*., 2019) changed natural language processing by introducing a significant shift in contextual comprehension. BERT, developed by Google AI, takes a bidirectional approach to understanding words, considering their contextual subtleties inside a phrase as well as their isolation. Its basis is based on the Transformer architecture, which emphasizes self-attention mechanisms.

During pre-training, BERT learns by guessing masked words in phrases, which fosters a comprehensive understanding of linguistic nuances. The model's bidirectional context analysis enables it to grasp complicated connections, considerably improving its performance across a variety of NLP tasks. BERT's integration represents words in a three-dimensional space, capturing semantic links and contextual significance.

BERT's influence extends to a wide range of applications, including sentiment analysis and question answering. Its versatility, shown by its fine-tuning for individual tasks, strengthens its position as a cornerstone in modern natural language comprehension, enabling advances in machine learning and linguistic analysis.

Explainable Artificial Intelligence

Interpretable machine learning is the capacity to comprehend and explain a machine learning model's decisions and predictions in a transparent and human-accessible manner. In many complicated models, particularly deep neural networks, the decision-making process may be like a "black box," making it difficult to understand how and why a certain prediction is generated.

Local Interpretable Model-Agnostic Explanations (LIME)

LIME (Ribeiro *et al*., 2016) is an approach for understanding the decision-making process of complicated machine learning models. LIME works on a simple principle: It gives interpretable explanations for specific predictions, making the model's behavior more apparent. Regardless of the model's complexity, LIME approximates its behavior around a given instance by perturbing the input data and monitoring the subsequent changes in predictions. These affected data are utilized to train a locally interpretable surrogate model, which is often a simpler model such as linear regression or decision trees. This surrogate model approximates the complicated model's behavior in the area of the provided instance, revealing how input characteristics influence the prediction.

LIME's model-agnostic nature assures its application across a wide range of machine-learning techniques, making it a flexible tool for interpretation. LIME fulfills the requirement for transparency in machine learning by providing localized and intelligible explanations for individual predictions, which fosters confidence and facilitates the wider adoption of complicated models in real-world applications.

SHAP

Shapley Additive ExPlanations (SHAP) (Lundberg and Lee, 2017) is a strong technique in the field of interpretable machine learning that provides information about the contribution of each feature to a model's predictions. SHAP values are derived from cooperative game theory and assign a unique value to each feature, signifying its effect on the model's output. SHAP values consider all conceivable feature combinations, calculating each feature's average contribution over several situations. By capturing the interaction effects of features, SHAP provides a thorough knowledge of the elements that influence a model's choice.

SHAP's major strength is its ability to equitably divide credit among contributing features. This fairness is rooted in the Shapley values principle, which ensures that each feature is given its fair amount of credit depending on its impact. SHAP values increase interpretability by identifying which attributes cause predictions to rise or fall, which helps with model debugging, validation, and improvement. SHAP's vast application to many model types and tasks highlights its importance in increasing openness and confidence in machine learning systems. Overall, SHAP is an invaluable tool for practitioners and scholars seeking to understand the complexities of complex models and make educated judgments on model behavior.

Related Work

Hate speech detection on social media is an important undertaking that aims to reduce the negative effect of inflammatory information online. Given the wide and diversified nature of social media sites, this identification requires powerful NLP and machine-learning approaches. Algorithms are taught to spot hate speech patterns, focusing on information that encourages discrimination, violence, or prejudice based on characteristics like race, ethnicity, religion, or gender. These algorithms often leverage linguistic details, contextual information, and user interactions to detect abusive words. However, the ever-changing nature of language and growing internet trends make it difficult to keep up with new hate speech patterns. Human-in-the-loop techniques and continual model changes are critical for adjusting to the rapidly changing terrain of online communication.

Hate speech detection efforts not only protect users from dangerous information but also help to create a safer and more inclusive online environment. Striking a balance between free expression and damaging speech is an ongoing problem that requires multidisciplinary cooperation and continual advances in machine learning to develop effective and ethical hate speech detection systems.

In this section, we discuss some of the relevant studies that focus on hate speech detection on social media using various machine learning and deep learning approaches. While performing the literature survey from sources like SCOPUS, Web of Science, and Google Scholar, we came across papers based on hate speech detection in varying applications of machine learning and deep learning techniques along with various combinations of data processing, balancing, data augmentation feature engineering, etc. These papers can be categorized into

groups based on the techniques applied. Following are the key concepts associated with hate speech detection.

Text classification using supervised learning: Using labeled datasets for text classification with the help of supervised machine learning algorithms.

Deep learning for text classification: Application of deep learning models such as Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN) such as Long Short-Term Memory Networks (LSTMs) and Bi-LSTMs, etc., for text classification.

Transfer Learning with Pre-trained Language Models: Apply transfer-learning techniques using models pre-trained on vast amounts of data, such as GPT or BERT and fine-tune them on a smaller hate speech dataset for better performance.

Ensemble methods: Combine multiple machine learning models to improve hate speech detection. The ensemble's decision can be based on voting or averaging the predictions of individual models.

NLP with feature engineering: Extract and engineer features from text data, such as sentiment, part-of-speech tags, and syntactic structures, to help machine learning models identify hate speech.

Data augmentation for imbalanced datasets: Augment training data for hate speech detection by generating synthetic examples, which can help address class imbalance in datasets.

Machine learning emerged as an effective tool for automatic hate speech detection in recent years. More and more ML algorithms are being used in combination with various feature extraction and data processing methods to enhance the performance and credibility of the models. A range of studies have explored the use of machine learning for hate speech detection.

Text Classification with Supervised Learning

Several early studies have focused on text classification using traditional supervised learning algorithms. Aulia and Budi (2019) applied a Support Vector Machine (SVM) classifier to detect hate speech in long Indonesian texts, achieving an 85% F1 score. Similarly, Jemima *et al.* (2022) reported a 6% improvement in F1 using deep neural networks for hate speech detection on social media. These studies demonstrate the utility of traditional supervised methods, but their limited ability to capture linguistic subtleties such as sarcasm or hidden meanings often leads to suboptimal performance.

Ensemble Methods and Data Augmentation

In the context of video content, Wu and Bhandary (2020) found that a Random Forest Classifier model performed best in classifying videos as normal or hateful based on spoken content. Rupesh *et al*. (2022) proposed a hate speech detection model for social media, using machine-learning algorithms such as logistic regression and random forest, with the capability to send alert messages to users and take strict actions against hate speech. These studies collectively demonstrate the potential of machine learning in detecting hate speech across different types of content and platforms. Besides supervised learning, the unsupervised learning approach is also applied to text classification. Saini *et al*. (2020) performed topic modeling using the Latent Dirichlet allocation (LDA) for abusive text and applied unsupervised learning using the Self Organizing Maps (SOM). In comparison to k-means clustering, SOM along with LDA enhanced the performance of the model.

Deep Learning Approaches

Deep learning algorithms also prove effective for automatic hate speech detection. Dubey *et al.* (2020); and Bisht *et al.* (2020) both achieved high accuracy in classifying toxic comments and hate speech on Twitter using LSTM, with precision and recall scores above 90%. Baruah *et al*. (2019) also found success with a BiLSTM model, particularly in the English language context. Faris *et al*. (2020) extended this study to the Arabic language, using a hybrid CNN-LSTM model to achieve strong results in classifying hate speech on Twitter. Pitsilis *et al*. (2018); and Paetzold *et al.* (2019) both achieved competitive results using RNNs, with (Pitsilis *et al*., 2018) focusing on user-related information and word frequency vectors and Paetzold using minimalistic compositional RNNs. Bennessir *et al.* (2022) combined RNNs with transformers for Arabic hate speech detection, (Kumar, 2022) and compared RNNs with other models, finding that RNNs performed well in both unweighted and weighted cases for foul language categorization.

Transfer Learning with Pre-Trained Language Models

Since its development at Google, there has been a manifold increase in applications of BERT (Kenton *et al*., 2019) in NLP. BERT has shown cutting-edge performance on a variety of NLP benchmarks and tasks. Its capacity to recognize context and record subtle word associations makes it a useful model for a variety of NLP-related problems. Besides, its bidirectional architecture can understand the context far better than conventional algorithms. Its ability to transform words as embeddings in high-dimensional vector space makes it more impactful in understanding the semantic relationship between the words. We have come across a number of studies applying BERT as a language model for text classification in hate speech detection studies. BERT-based hate speech detection with various enhancements is a comprehensive approach incorporating different methods and techniques to improve hate speech detection. Mozafari *et al*. (2020) introduced a novel transfer learning approach using BERT for detecting hateful content on social media. Their

fine-tuned model outperformed other conventional methods in terms of performance metrics like precision, recall, and F1 score. Caselli *et al*. (2021) re-trained the BERT model with the RAL-E Reddit database and compared its performance with the general-purpose BERT model. Re-trained BERT outperformed the general model in detecting the abusive content in the dataset extracted from Reddit comments. Koufakou *et al*. (2020) incorporated lexical features with the baseline BERT model to observe the change in performance. They found that BERT models with lexical features significantly improved the performance in comparison to the baseline BERT model. Besides, there is a range of studies that focus on improving the performance of BERT by incorporating various methods for predicting hate speech. These studies vary from English language to other foreign (Junqueira *et al*., 2023; Makram *et al*., 2022; Bayrak *et al*., 2023) and local languages (Bilal *et al*., 2023; Ghosh *et al*., 2023).

Materials and Methods

In this section, we discuss the detailed methodology of the proposed framework for hate speech detection on Twitter data using BERT. Figure (1) shows the graphical representation of the steps followed.

Data Collection

The training and testing of the prediction model was performed using the data available from (Nobata *et al*., 2016). The dataset is publicly available and used extensively across platforms for hate speech detection model development using AI. It contains a dataset of ~25000 tweets having examples of hateful, offensive, and neutral language. Out of three classes, 5.77% of the tweets belong to hate speech, 77.43% to offensive language class and the rest 16.80% belong to neither of the two classes. Figure (2) shows the visual representation of the class distribution.

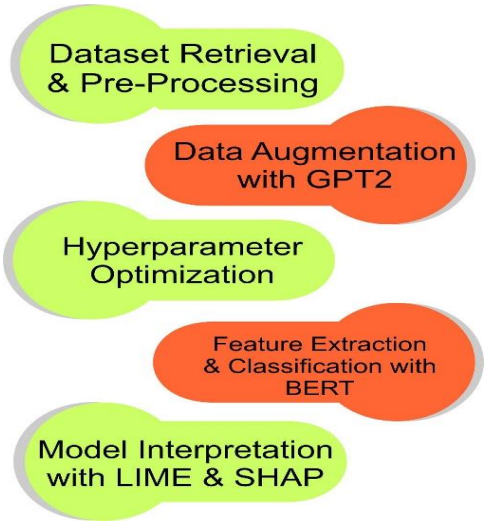

Fig. 1: Graphical representation of the proposed workflow

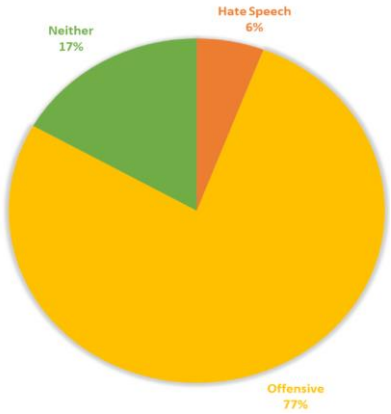

Fig. 2: Class-wise data distribution

Pre-Processing

Data pre-processing is a crucial step in machine learning applications. The presence of redundancies, null values, and irrelevant information or noise might affect the quality of model training and this might lead to prediction biases. Therefore, it is very important to pre-process the dataset before using it for model training and testing. We carefully pre-processed the datasets prior to model development to reduce all sorts of redundancies. Pre-processing of this type of data is very crucial because it contains a lot of informal expressions that are difficult to process by NLP models. Sometimes the sample can have spelling mistakes, symbols and emojis, numerals and abbreviations, etc., that can be hard to process. Therefore, handling all such irregularities in the input text is very important.

Data Augmentation Using GPT2

GPT-2 generates synthetic hate speech samples, which are rigorously validated through human review and statistical analysis to ensure they enhance dataset diversity without introducing noise. This augmented dataset is then used to fine-tune BERT, improving the model's detection accuracy and robustness. The approach is validated through cross-validation and performance metrics to ensure effective and reliable hate speech detection across varied datasets. Unbalanced data in machine learning is a major issue. Especially in NLP-based studies, it is crucial to standard and balance input to ensure the credibility of the model. The dataset that we used suffered from data imbalance in terms of the length of the tweets. For example, the word length in tweets ranges from 2 ~ 50. To tackle this issue, a GPT2-based model was used for scaling the tweets. As discussed earlier, GPT2 is a pretrained model for text generation. It can successfully generate text based on the inputs and is able to guess the preceding words in each sentence. In this way, the dataset generated using the GPT2 was further utilized for model training and testing.

Feature Extraction

In this framework, BERT extracts rich, contextual embeddings from text by encoding semantic and syntactic information into high-dimensional vectors. These embeddings serve as crucial features for hate speech classification. GPT-2 enhances feature extraction by generating synthetic text samples that are contextually like real hate speech, thereby expanding the feature space. This augmented data is integrated into the training process, providing additional nuances that improve the model's ability to detect hate speech. A more thorough discussion should cover the specifics of how these embeddings are utilized, how synthetic features are validated, and their impact on the overall model performance. Feature extraction refers to the process of transforming raw (text) data into numerical features that can be processed while preserving the information in the original data set. It is used to delete the non-dominant features and accordingly reduce the training time and mitigate the complexity of the developed classification models. We have used BERT, Glove, and Word2Vec for word embedding and feature extraction. BERT is a model that knows how to represent text. Once you feed it a sequence, it repeatedly scans left and right until it outputs a vector representation for every word.

Hyperparameter Optimization

Hyperparameter tuning or optimization is a standard practice in machine learning. This method systematically tests a range of values for each hyperparameter, ultimately selecting the combination that yields the best performance on the validation set. The hyperparameters tuned in this study included learning rate, batch size, and the number of epochs. The results showed that fine-tuning these parameters significantly impacted the model's accuracy and convergence speed.

While this optimization process improved model performance, it is worth considering the computational expense associated with hyperparameter tuning. Large models like BERT require substantial resources for training and running multiple trials to optimize parameters can be time-consuming and costly. Techniques such as Bayesian optimization or random search could be explored as alternatives to GridSearchCV, potentially reducing the computational burden while still achieving competitive results.

It is the process of testing different sets of parameters associated with the model being applied. We also performed hyperparameter optimization for the BERT model using different sets of parameters such as the number of epochs, batch size, optimizers, and methods for calculating loss. A test run was performed with a grid of parameters using GridSearchCV implemented from the scikit-learn library. The performance of the BERT model was tested on input data with these sets of parameters. Eventually, performance was evaluated based on prediction accuracy. GridSearchCV is capable of deducing the best parameters for a given model that ensure higher prediction accuracy.

Model Deployment and Classification

To train and evaluate the BERT model, we used the TensorFlow hub and imported the group of BERT models. The model was optimized throughout the training phase using the 'adamw' optimizer, which is a common tool for fine-tuning transformer-based models. The use of the 'small_bert' variation, notably with a four-layer design, 512 hidden units, and eight attention heads, reduced processing requirements while preserving the core of BERT's contextual awareness.

The model was trained with different combinations of epochs, iterating over the dataset to improve its understanding of language subtleties. This lengthy training period enabled the model to acquire complicated contextual linkages, which were necessary for good predictions in future challenges. The selection of 'bert_en_uncased' indicates an uncased English model suited for a wide variety of NLP applications.

Following training, the model was rigorously tested to determine its generalization performance on previously encountered data. Metrics including accuracy, precision, and recall were used to assess the model's ability to understand and anticipate patterns in the data. The use of best practices in model fine-tuning, optimizer selection, and prolonged training durations to maximize the power of BERT for nuanced language comprehension tasks.

Model Interpretation

After the successful deployment of the BERT model for classification of the input text, we applied LIME and SHAP for model interpretation. Model interpretation methods are gaining popularity among the community. It is crucial to understand the meaning of predictions for real-world applications. It also helps in improving the performance of the model. Both LIME and SHAP are being extensively used for machine learning model interpretation. Details of the model interpretation are provided in subsequent sections.

Results and Discussion

In this section, we discuss the performance evaluation of the BERT model and its interpretation. As discussed earlier, two approaches for text classification were applied. The first one uses the input data without augmentation and the second approach incorporates data augmentation using GPT-2 along with a pre-trained BERT model. The objective is the observe the changes in the performance of the BERT model with and without data augmentation.

BERT without Data Augmentation

After successful data cleaning and preprocessing, the dataset was subjected to split into train, test, and validation sets. The BERT model's performance for hate speech detection reached a commendable 81% accuracy on the test data without any data augmentation. This result is consistent with expectations, as BERT is known for its ability to handle a wide variety of natural language processing tasks with high efficiency due to its transformer-based architecture. However, the introduction of GPT-2 for data augmentation pushed this accuracy to 86%, highlighting the benefits of expanding the dataset with synthetic examples. The increased accuracy can be attributed to the model's enhanced ability to generalize to previously unseen examples, which underscores the significance of having a diverse and robust dataset in tasks like hate speech detection.

This improvement aligns with findings in previous research, where data augmentation techniques have been shown to improve model performance by introducing variability and covering edge cases that would otherwise be missed in smaller datasets. The 5% boost in accuracy suggests that the augmented data likely filled gaps in the original dataset, helping the BERT model learn nuanced language patterns associated with hate speech more effectively. This is a standard practice in machine learning to reduce biases in prediction. The inputs for the models were prepared using TensorFlow functionalities prior to model training and testing. Eventually, models we trained with training datasets and predictions were made using the test set left out for validation purposes. The model achieved an overall accuracy of 0.81 for the test samples.

BERT with Data Augmentation Using GPT-2

The discussion of data augmentation highlights its importance in improving the generalization capability of the BERT model. GPT-2, a generative model, was employed to generate synthetic examples that mimic the linguistic structure and patterns present in the original hate speech dataset. These synthetic examples helped mitigate the potential problem of overfitting by providing more varied training data.

However, it is essential to acknowledge that data augmentation with synthetic text is not without its challenges. One major concern is the potential introduction of biases from the language model used for augmentation. GPT-2, for instance, is trained on large corpora of internet text and there is a risk that it could inadvertently generate biased or inappropriate examples, particularly when tasked with generating hate speech-related data. Ensuring that the synthetic data does not reinforce harmful biases is crucial. In this study, close scrutiny of the generated text was conducted to ensure that the augmented dataset remained representative and unbiased, though further research could delve into methods for systematically addressing this issue.

Data augmentation and leveling is an effective method to enhance the performance of the model. Data augmentation to make the models generalize better for hate speech detection. Therefore, we applied GPT-2 for augmentation of the input tweets up to the length of 50. After fine-tuning the model and training it with the resampled data, the performance of the BERT model enhanced significantly. The model achieves an accuracy of 0.86 with the augmented dataset which is higher than that without augmentation. This proves the effectiveness of the GPT-2 in enhancing the readability of the input data for automated text classification. Figure (3) shows the values of loss and validation loss for the BERT model. It is observed that the model achieves convergence between 0-5 epochs. A similar trend is observed for accuracy.

Interpretation of BERT with LIME

One of the key contributions of this study was the use of interpretability tools like LIME and SHAP to analyze model predictions. These tools provided insights into how the model arrived at its predictions, making it easier to identify cases where the model might be relying on spurious correlations or misinterpreting input features. For instance, LIME allowed for the identification of specific words or phrases that contributed the most to the model's classification decisions, while SHAP provided a global understanding of feature importance across the entire dataset.

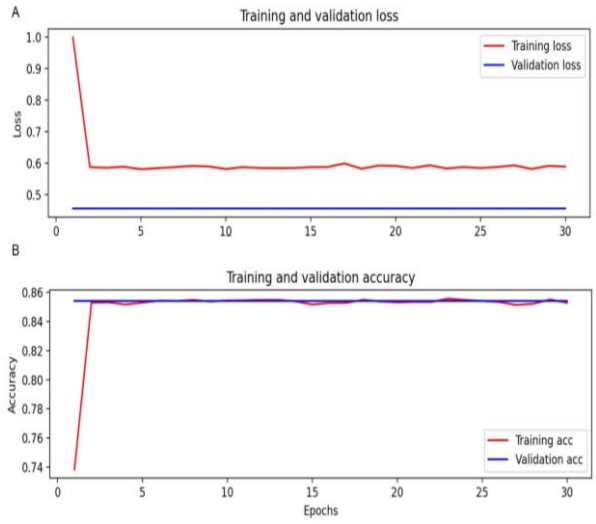

Fig. 3: Line plot showing the trade-off between (A) loss and validation loss, (B) accuracy and validation accuracy for the BERT model

The inclusion of these interpretability tools is crucial in a task like hate speech detection, where the consequences of incorrect predictions can be severe. False positives, where benign content is flagged as hate speech, can lead to censorship and unnecessary restrictions on free speech. On the other hand, false negatives, where actual hate speech is not detected, can allow harmful content to spread unchecked. By using LIME and SHAP, the model's predictions become more transparent, and potential areas for improvement can be identified more easily.

However, it should be noted that interpretability methods like LIME and SHAP also have limitations. LIME, for example, provides explanations by perturbing the input data and observing changes in model predictions. While this can offer valuable insights, it may not always capture the full complexity of the model's decision-making process. Similarly, SHAP values can sometimes be difficult to interpret in models as large and complex as BERT, particularly when dealing with high-dimensional input data like text.

LIME provides local explanations for individual predictions of a model. For a BERT model, LIME generates perturbed instances of the input text and observes how the model's predictions change. LIME assigns weights to the features (words or tokens in the text) based on how much they contribute to the model's predictions in the local neighborhood of a specific instance. This helps identify which words or phrases are most influential in determining the outcome of the BERT model for a given prediction. It employs visualizations through highlighting important words in the instances, to present the interpretation in an accessible manner. This makes it easier to understand the contribution of different words to the BERT model's outcome for a specific prediction. Figure (4) shows an example of how LIME interprets the output of a BERT model by assigning prediction probabilities for all the classes using the feature importance values of the words (token) in the sample text.

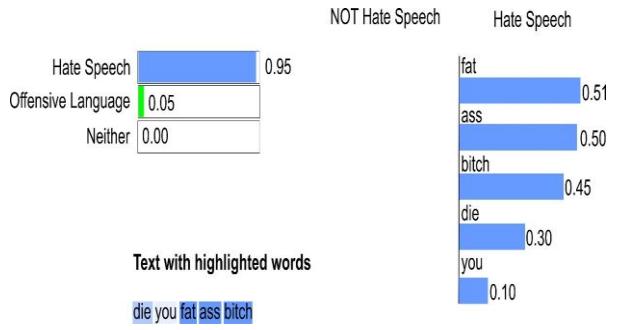

Fig. 4: BERT model interpretation using LIME

Interpretation of BERT with SHAP

SHAP is helpful in model interpretability for BERT. SHAP values provide a way to understand the contribution of each feature to the model's output. In the context of interpreting a BERT model's outcome, SHAP values can be used to attribute the prediction made by the model to individual input features. SHAP values quantify the impact of each feature on the model's prediction. For a BERT model, these features might represent different words or tokens in the input text. SHAP helps identify which words or tokens had a more significant influence on the final prediction. Figure (5) shows the SHAP outputs for sample texts used for interpreting the BERT model for all three classes.

Limitations and Future Directions

While the results of this study demonstrate the potential of BERT and GPT-2 for hate speech detection, several limitations should be acknowledged. First, the dataset used for training and testing, although augmented, still may not be fully representative of the diversity of hate speech encountered in real-world applications. Hate speech can vary significantly across different platforms, languages, and cultural contexts. Future work could explore the generalization of the model across multiple platforms and in multilingual environments.

Moreover, while data augmentation improved model performance, there is a risk that the synthetic data generated by GPT-2 may introduce noise or unintended biases. This is an area that requires careful consideration and future research could focus on developing more sophisticated data augmentation techniques that are tailored specifically for hate speech detection.

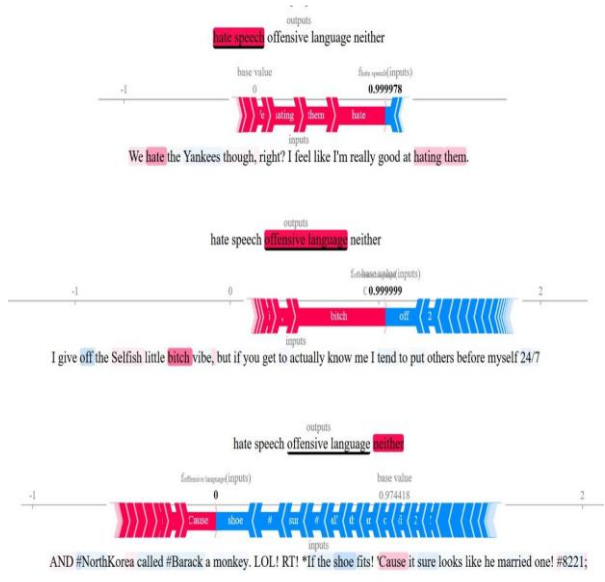

Fig. 5: BERT model interpretation using SHAP

Finally, it is important to consider the ethical implications of deploying hate speech detection models in real-world applications. These models must strike a balance between accurately detecting harmful content and avoiding over-censorship. Developing models that are not only accurate but also fair and transparent will be crucial as hate speech detection technologies continue to evolve.

To build on the findings of this study, future research could explore several avenues. One potential direction is the use of multilingual models to detect hate speech in different languages and across different cultural contexts. Hate speech is a global problem and models that can operate effectively in multiple languages would have a broader impact. Additionally, integrating context-aware models that consider the social dynamics of conversations, such as the role of user interactions or the spread of hate speech through networks, could improve the model's ability to detect hate speech in complex scenarios.

Another area for future exploration is the development of more advanced data augmentation techniques. While GPT-2 provided useful synthetic examples in this study, other generative models or approaches, such as Generative Adversarial Networks (GANs), could be explored for creating more realistic and diverse training data.

Lastly, further work on model interpretability is essential. While LIME and SHAP provided valuable insights into the model's decision-making process, more robust interpretability methods that are better suited to large, complex models like BERT are needed. This will ensure that models remain transparent and that their predictions can be trusted in real-world applications.

Conclusion

This research examined the efficacy of a BERT model for hate speech identification, highlighting its ability to identify objectionable information inside online conversations. The first evaluation suggested adequate baseline performance, however the model showed significant improvement after data augmentation using GPT2. Following augmentation, prediction accuracy increased to 81%, demonstrating the significant influence of augmented data on improving classification. The integration between BERT and GPT2 demonstrated a sophisticated comprehension of hate speech subtleties, exhibiting advanced language models' adaptation to the complexity of online communication. However, ethical concerns about biases in enhanced data need continual investigation. This research confirms the effectiveness of hybrid models in detecting hate speech, underlining the ongoing need for innovation to handle the changing terrain of language patterns in online conversation. The increased accuracy post-augmentation indicates that the model was successfully fortified against the obstacles given by dynamic language usage, leading to advances in responsible and effective natural language processing for minimizing the impact of objectionable material.

Acknowledgment

I thank my PhD supervisor for their guidance and support.

Funding Information

The authors have not received any financial support or funding to report.

Author's Contributions

Usman: Conceptualization, methodology, software, investigation, implementation writing the original draft.

Nabeela Hasan: Extensive editing and formatting of the manuscript, figure arrangement, and upgradation.

Syed Mohammad Khurshid Quadri: Supervision, reviewed the manuscript.

Ethics

This article is original and contains unpublished material. The corresponding author confirms that all of the other authors have read and approved the manuscript and that no ethical issues are involved.

References

Aulia, N., & Budi, I. (2019). Hate Speech Detection on Indonesian Long Text Documents Using Machine Learning Approach. *Proceedings of the 2019 5th International Conference on Computing and Artificial Intelligence*, 164–169. https://doi.org/10.1145/3330482.3330491

Baruah, A., Barbhuiya, F., & Dey, K. (2019). ABARUAH at SemEval-2019 Task 5 : Bi-directional LSTM for Hate Speech Detection. *Proceedings of the 13th International Workshop on Semantic Evaluation*, 371–376. https://doi.org/10.18653/v1/s19-2065

Bayrak, Ş., Karaca, A., Toson, F., Kocabey, A., & Arslanoğlu, F. B. (2023). Detection of Hate Speech in Turkish Social Media Posts with BERT-Base Model. *2023 31st Signal Processing and Communications Applications Conference (SIU)*, 1–4. https://doi.org/10.1109/siu59756.2023.10224040

Bennessir, M. A., Rhouma, M., Haddad, H., & Fourati, C. (2022). iCompass at Arabic Hate Speech 2022: Detect Hate Speech Using QRNN and Transformers. *Proceedinsg of the 5th Workshop on Open-Source Arabic Corpora and Processing Tools with Shared Tasks on Qur'an QA and Fine-Grained Hate Speech Detection*, 176–180.

Bilal, M., Khan, A., Jan, S., Musa, S., & Ali, S. (2023). Roman Urdu Hate Speech Detection Using Transformer-Based Model for Cyber Security Applications. *Sensors*, *23*(8), 3909. https://doi.org/10.3390/s23083909

Bisht, A., Singh, A., Bhadauria, H. S., Jitendra, V., & Kriti. (2020). Detection of Hate Speech and Offensive Language in Twitter Data Using LSTM Model. *Recent Trends in Image and Signal Processing in Computer Vision*, 243–264. https://doi.org/10.1007/978-981-15-2740-1_17

Caselli, T., Basile, V., Mitrović, J., & Granitzer, M. (2021). HateBERT: Retraining BERT for Abusive Language Detection in English. *Proceedings of the 5th Workshop on Online Abuse and Harms (WOAH 2021)*, 17–25. https://doi.org/10.18653/v1/2021.woah-1.3

Davidson, T., Warmsley, D., Macy, M., & Weber, I. (2017). Automated Hate Speech Detection and the Problem of Offensive Language. *Proceedings of the International AAAI Conference on Web and Social Media*, *11*(1), 512–515. https://doi.org/10.1609/icwsm.v11i1.14955

Dubey, K., Nair, R., Khan, Mohd. U., & Shaikh, Prof. S. (2020). Toxic Comment Detection using LSTM. *2020 Third International Conference on Advances in Electronics, Computers and Communications (ICAECC)*, 1–8. https://doi.org/10.1109/icaecc50550.2020.9339521

Faris, H., Aljarah, I., Habib, M., & Castillo, P. (2020). Hate Speech Detection using Word Embedding and Deep Learning in the Arabic Language Context. *Proceedings of the 9th International Conference on Pattern Recognition Applications and Methods - ICPRAM*, 453–460. https://doi.org/10.5220/0008954004530460

Fortuna, P., & Nunes, S. (2019). A Survey on Automatic Detection of Hate Speech in Text. *ACM Computing Surveys*, *51*(4), 1–30. https://doi.org/10.1145/3232676

Ghosh, K., Sonowal, D., Basumatary, A., Gogoi, B., & Senapati, A. (2023). Transformer-Based Hate Speech Detection in Assamese. *2023 IEEE Guwahati Subsection Conference (GCON)*, 1–5. https://doi.org/10.1109/gcon58516.2023.10183497

Hasan, N., & Alam, M. (2023). Role of machine learning approach for industrial internet of things (IIoT) in cloud environment-a systematic review. *International Journal of Advanced Technology and Engineering Exploration*, *10*(108), 1391–1416. https://doi.org/10.19101/ijatee.2023.10101133

Hasan, N., & Chaudhary, K. (2024). ρi-BLoM: a privacy preserving framework for the industrial IoT based on blockchain and machine learning. *International Journal of System Assurance Engineering and Management*. https://doi.org/10.1007/s13198-024-02330-x

Jahan, M. S., & Oussalah, M. (2023). A systematic review of hate speech automatic detection using natural language processing. *Neurocomputing*, *546*, 126232. https://doi.org/10.1016/j.neucom.2023.126232

Jemima, P. P., Majumder, B. R., Ghosh, B. K., & Hoda, F. (2022). Hate Speech Detection using Machine Learning. *2022 7th International Conference on Communication and Electronics Systems (ICCES)*, 1274–1277. https://doi.org/10.1109/icces54183.2022.9835776

Junqueira, J. da R., Da Silva, F., Costa, W., Carvalho, R., Bender, A., Correa, U., & Freitas, L. (2023). BERTimbau in Action: An Investigation of its Abilities in Sentiment Analysis, Aspect Extraction, Hate Speech Detection and Irony Detection. *The International FLAIRS Conference Proceedings*, *36*(1). https://doi.org/10.32473/flairs.36.133186

Kenton, J. D., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *Proceedings of NaacL-HLT*, 4171–4186.

Koufakou, A., Pamungkas, E. W., Basile, V., & Patti, V. (2020). HurtBERT: Incorporating Lexical Features with BERT for the Detection of Abusive Language. *Proceedings of the Fourth Workshop on Online Abuse and Harms*, 34–43. https://doi.org/10.18653/v1/2020.alw-1.5

Kumar, A. (2022). A Study: Hate Speech and Offensive Language Detection in Textual Data by Using RNN, CNN, LSTM and BERT Model. *2022 6th International Conference on Intelligent Computing and Control Systems (ICICCS)*, 1–6. https://doi.org/10.1109/iciccs53718.2022.9788347

Lundberg, S. M., & Lee, S.-I. (2017). A Unified Approach to Interpreting Model Predictions. *Advances in Neural Information Processing Systems*. Advances in Neural Information Processing Systems.

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. https://doi.org/10.1371/journal.pone.0221152

Makram, K., Nessim, K. G., Abd-Almalak, Malak Emad, Roshdy, S. Z., Salem, S. H., Thabet, F. F., & Mohamed, E. H. (2022). CHILLAX} - at {A}rabic Hate Speech 2022: A Hybrid Machine Learning and Transformers based Model to Detect {A}rabic Offensive and Hate Speech. *Proceedinsg of the 5th Workshop on Open-Source Arabic Corpora and Processing Tools with Shared Tasks on Qur`an QA and Fine-Grained Hate Speech Detection*, 194–199.

Mozafari, M., Farahbakhsh, R., & Crespi, N. (2020). A BERT-Based Transfer Learning Approach for Hate Speech Detection in Online Social Media. *Complex Networks and Their Applications VIII*, 928–940. https://doi.org/10.1007/978-3-030-36687-2_77

Nations, U. (2023). *Understanding Hate Speech*. https://www.un.org/en/hate-speech/understanding-hate-speech/what-is-hate-speech

Nobata, C., Tetreault, J., Thomas, A., Mehdad, Y., & Chang, Y. (2016). Abusive Language Detection in Online User Content. *Proceedings of the 25th International Conference on World Wide Web*, 145–153. https://doi.org/10.1145/2872427.2883062

Nockleby, J. T. (2000). Hate speech. *Encyclopedia of the American Constitution*, *3*(2), 1277–1279.

Paetzold, G. H., Zampieri, M., & Malmasi, S. (2019). UTFPR at SemEval-2019 Task 5: Hate Speech Identification with Recurrent Neural Networks. *Proceedings of the 13th International Workshop on Semantic Evaluation*, 519–528. https://doi.org/10.18653/v1/s19-2093

Pitsilis, G. K., Ramampiaro, H., & Langseth, H. (2018). Effective hate-speech detection in Twitter data using recurrent neural networks. *Applied Intelligence*, *48*(12), 4730–4742. https://doi.org/10.1007/s10489-018-1242-y

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why Should I Trust You?": Explaining the Predictions of Any Classifier. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1135–1144. https://doi.org/10.1145/2939672.2939778

Rupesh, C., Ritik, G., Pranav, G., Mangesh, G., & Pawa, A. (2022). Hate Speech Detection on Social Media Using Machine Learning Algorithms. *Journal of Cognitive Human-Computer Interaction*, *2*(2), 56–59.

Saini, Y., Bachchas, V., Kumar, Y., & Kumar, S. (2020). Abusive Text Examination Using Latent Dirichlet allocation, Self Organizing Maps and K Means Clustering. *2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS)*, 1233–1238. https://doi.org/10.1109/iciccs48265.2020.9121090

Wu, C. S., & Bhandary, U. (2020). Detection of Hate Speech in Videos Using Machine Learning. *2020 International Conference on Computational Science and Computational Intelligence (CSCI)*, 585–590. https://doi.org/10.1109/csci51800.2020.00104

Zampieri, M., Malmasi, S., Nakov, P., Rosenthal, S., Farra, N., & Kumar, R. (2019). Predicting the Type and Target of Offensive Posts in Social Media. *Proceedings of the 2019 Conference of the North*, 1415–1420. https://doi.org/10.18653/v1/n19-1144