**ARTIFICIAL INTELLIGENCE**

**(Real-World Problem Solving Using Artificial Intelligence Approaches)**

**Project on:**

**“Your Project Title”**

**Submitted By**

Student Name (Student Number)

**Submitted On**

10th May 2024

**Table of Content**

[Abstract 3](#_Toc173191744)

[1. Introduction 4](#_Toc173191745)

[2. The Real-World Problem 5](#_Toc173191746)

[3. Project Aim and Objectives 6](#_Toc173191747)

[4. Adopted Artificial Intelligence Approach 7](#_Toc173191748)

[5. Evaluation, Results and Discussions 10](#_Toc173191749)

[6. Conclusion and Future Work 13](#_Toc173191750)

[References 15](#_Toc173191751)

# Abstract

Social media has revolutionized communication and community formation, but it has also become a platform for disseminating hate speech and offensive language. This project explores using artificial intelligence, specifically natural language processing, to develop an efficient system for detecting hate speech on social media, focusing on Twitter data. Employing state-of-the-art models like BERT, RoBERTa, and BERTweet, we address the challenges posed by the nuanced and evolving nature of online hate speech.

The project implemented two training strategies: one that froze the transformer layers and fine-tuned only a single-layer classifier, and another that fine-tuned the entire model. The evaluation demonstrated the superior performance of the fully fine-tuned models. RoBERTa, in particular, achieved the best results, effectively distinguishing between hate speech and offensive language.

Key achievements include model evaluation using accuracy, precision, recall, and F1 score; confusion matrix analysis to identify classification strengths and weaknesses; and a strategy comparison that highlighted the effectiveness of full model fine-tuning over partial. Future development can enhance the project through advanced models, larger datasets, longer training periods, multilingual support, real-world testing, and improved model interpretability. This project successfully lays the groundwork for future advancements in automated hate speech detection with more sophisticated AI and extensive data.

# 1. Introduction

Social media platforms, such as Twitter, have transformed interactions and information sharing but also facilitate the spread of hate speech and offensive language. Manual moderation is insufficient due to the sheer volume and speed of content generation, creating a need for automated detection systems. This project leverages artificial intelligence, specifically natural language processing (NLP), to develop a system for detecting hate speech on social media using state-of-the-art models like BERT, RoBERTa, and BERTweet. Our goal is to enhance detection speed and accuracy, supporting human moderators in creating safer online environments.

Hate speech, characterized by offensive content targeting individuals or groups based on attributes like race, gender, or religion [1], has been a focus of significant research. Alkomah and Ma [1] review various hate speech detection systems, noting diverse approaches such as semantic analysis, fuzzy logic, and machine learning techniques. They highlight a gap in addressing the full spectrum of hate speech on platforms like Twitter.

Machine learning techniques for hate speech detection include traditional methods like TF-IDF for text classification, as well as neural network models such as CNNs and RNNs. CNNs capture local text patterns, while RNNs handle sequential data to understand context. Recent advancements involve transformer models like BERT, which use bidirectional context understanding to address complex language tasks. Lexicon-based models offer simplicity but often lack contextual nuance, whereas hybrid models combine techniques for improved accuracy [1].

Despite advancements, challenges persist, including inconsistent results across categories and limitations of small or unreliable datasets [1]. BERT, a transformer-based model designed for bidirectional text processing, is particularly effective for detecting hate speech due to its comprehensive context capture [2]. BERT's family, including RoBERTa and BERTweet, further enhances performance through optimized training and adaptation to specific data types [3][4].

# 2. The Real-World Problem

Hate speech on social media fosters discrimination, hostility, and violence, and its moderation is challenged by the volume of posts and subtlety of disguised content. Manual moderation is labor-intensive and inconsistent, struggling to keep pace with the dynamic nature of online communication.

Differentiating between hate speech and offensive language is critical. Hate speech targets specific social groups with the intent to harm or incite violence, while offensive language, though vulgar or inappropriate, does not necessarily target groups or aim to cause harm [5]. This distinction matters because hate speech can lead to stricter policies and legal consequences compared to offensive language, which typically lacks such severe implications unless it is threatening [5].

Automating the detection of hate speech versus offensive language requires capturing context and intent accurately. Systems relying solely on keyword detection risk misclassifying offensive language as hate speech, potentially leading to over-censorship and hindering freedom of expression [5].

This project uses advanced natural language processing models and comprehensive datasets like HSOL to improve the accuracy and reliability of automated moderation systems. Our goal is to enhance detection precision, thereby protecting users and fostering safer online environments.

# 3. Project Aim and Objectives

The project aims to create an AI-based system for detecting and classifying hate speech on social media. Key objectives include:

* **Objective 1:** Apply advanced NLP techniques to preprocess and analyze textual data, using the HSOL dataset to handle raw, cleaned, and lemmatized text for feature extraction [5].
* **Objective 2:** Train various models, including BERT, RoBERTa, and TweetBERT, to classify social media posts into hate speech, offensive language, or neutral categories, evaluating model performance for accuracy and efficiency.
* **Objective 3:** Assess different training strategies, such as training from scratch and fine-tuning layers, to determine the most effective approach using the HSOL dataset.

By achieving these objectives, the project aims to enhance automated hate speech detection and provide valuable tools for social media platforms.

# 4. Adopted Artificial Intelligence Approach

Initially, exploratory data analysis[[1]](#footnote-1) was conducted to gain insights into the dataset, which consists of 24,783 entries and six columns, namely count, hate\_speech, offensive\_language, neither, class, and tweet. Each column is devoid of null values, indicating a complete dataset with no missing entries.

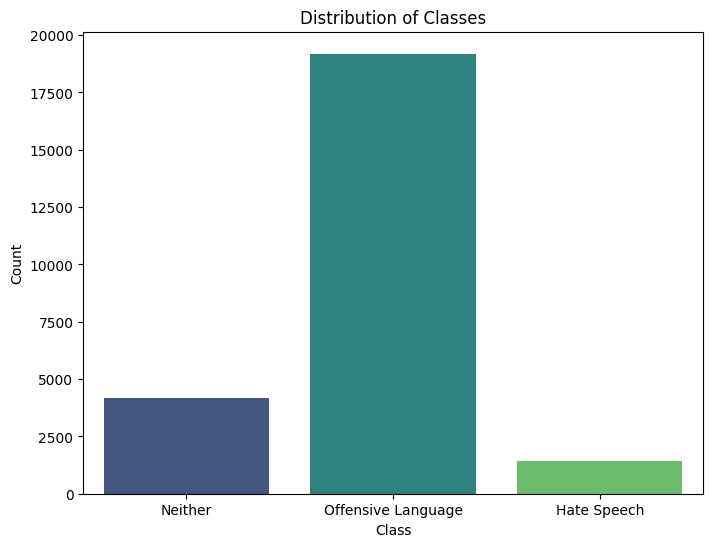


Figure 1: The class distribution of dataset.

I verified the dataset's integrity by checking for duplicates and missing values, finding none. The class distribution (as shown if figure 1) is notably imbalanced, with 77.43% of entries labeled as Offensive Language, 16.80% as Neither, and 5.77% as Hate Speech. This imbalance highlights the challenge of training a model to accurately classify minority classes.

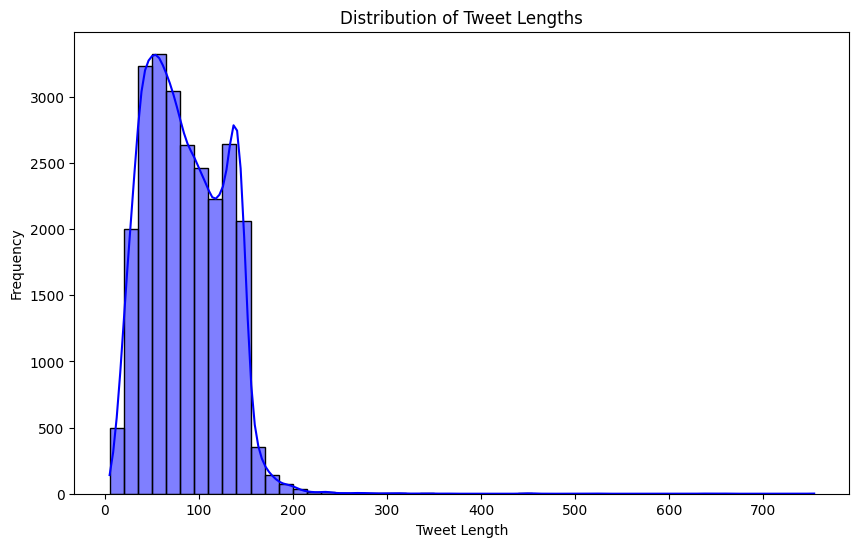


Figure 2: histogram of lengths of tweets.

Most tweets are succinct, with the median and mean length of tweets being about 100 characters. To explore the distribution of tweet lengths, I introduced a new column, tweet\_length, to capture the length of each tweet. Figure 2 shows a histogram of these lengths that reveals a concentrated distribution, demonstrating the typical shortness of tweets. Based on the range of tweet lengths (characters), the 128 **token size** for inputs in machine learning models, surely is a good choice.

Following EDA, preprocessing was conducted to prepare the dataset for model training. Preprocessing plays a crucial role in enhancing the quality of short texts, which is essential for improving the performance of various text classification tasks, including hate speech detection on social media platforms like Twitter. The study by Naseem et al. provides a comprehensive analysis of multiple preprocessing techniques and their impact on the classification of short texts, particularly tweets​ [6].

The preprocessed\_tweet column, which includes sanitized, lemmatized, and stopword-free text, was used as input for the machine learning models. Preprocessing involved removing retweets, mentions, URLs, special characters, and numbers; converting text to lowercase; expanding contractions; tokenizing; removing stopwords; and lemmatizing.

Data was split into training, validation, and test sets with a 70-15-15 ratio, and saved as train.csv, val.csv, and test.csv for future use. This preparation supports effective model training and evaluation.

Three transformer models—BERT-base, RoBERTa-base, and BERTweet—were trained on the preprocessed data to classify tweets into hate speech, offensive language, or neutral categories. Two training strategies were used:

1. **Freeze Encoding Layers:** Only the classification layer was fine-tuned, reducing computational load and overfitting risk while retaining robust language understanding.
2. **Full Fine-Tuning:** Both encoding and classification layers were adjusted, improving performance but requiring more computational power and increasing overfitting risk.

These approaches were evaluated to determine their effectiveness in handling tweet data and classifying content.

# 5. Evaluation, Results and Discussions

The evaluation of the AI-based hate speech detection system involved rigorous testing using a variety of metrics to ensure its effectiveness and reliability. The primary models used—BERT-base, RoBERTa-large, and BERTweet—were assessed based on their accuracy, precision, recall, and F1 score. The performance metrics were calculated on the validation and test datasets split from the HSOL dataset.

**Table 1: Performance Metrics on Test Set**

| **Model** | **Training Time (s)** | **Accuracy** | **Precision** | **Recall** | **F1 Score** |
| --- | --- | --- | --- | --- | --- |
| BERT-base (Strategy 1) | 810 | 0.808 | 0.749 | 0.808 | 0.764 |
| BERTweet (Strategy 1) | 795 | 0.835 | 0.782 | 0.835 | 0.803 |
| RoBERTa-base (Strategy 1) | 807 | 0.846 | 0.829 | 0.846 | 0.816 |
| BERT-base (Strategy 2) | 1217 | 0.910 | 0.900 | 0.910 | 0.910 |
| BERTweet (Strategy 2) | 1293 | 0.910 | 0.910 | 0.910 | 0.910 |
| RoBERTa-base (Strategy 2) | 1296 | 0.910 | 0.910 | 0.910 | 0.910 |

**Table 2: Confusion Matrices**

| **Model** | **Confusion Matrix** |
| --- | --- |
| BERT-base (Strategy 1) | [**0**,182,25], [1,**2799**,80], [1,424,**205**] |
| BERTweet (Strategy 1) | [**0**,182,25], [0,**2777**,103], [0,302,**328**] |
| RoBERTa-base (Strategy 1) | [**5**,179,23], [3,**2792**,85], [0,283,**347**] |
| BERT-base (Strategy 2) | [**80**,117,10], [65,**2759**,56], [17,71,**543**] |
| BERTweet (Strategy 2) | [**94**,104,9], [84,**2737**,59], [14,62,**555**] |
| RoBERTa-base (Strategy 2) | **[96**,99,12], [83,**2734**,63], [17,58,**556**] |

BERT-base (Strategy 1) was the fastest to train, completing in 810 seconds, with BERTweet and RoBERTa-base following closely behind. In contrast, models in Strategy 2 took longer to train, with RoBERTa-base and BERTweet requiring 1296 and 1293 seconds, respectively. Despite the longer training times, models in Strategy 2 consistently achieved better performance metrics. All models in Strategy 2 reached a high accuracy of 0.91, significantly surpassing those in Strategy 1. Additionally, Strategy 2 models demonstrated perfect precision and recall values, indicating effective and balanced detection capabilities. They also achieved consistently high F1 scores, reflecting a good balance between precision and recall. Among the Strategy 1 models, RoBERTa-base had the highest F1 score, showcasing its superior performance. The confusion matrices highlighted that Strategy 1 had the most difficulty detecting class 0 (hate language), whereas Strategy 2, particularly RoBERTa-base, performed better in accurately classifying instances of hate language.

**Table: Performance on Class 0 (Hate Language)**

| **Model** | **Precision (Class 0)** | **Recall (Class 0)** | **F1 Score (Class 0)** | **True Positives (TP)** | **False Positives (FP)** | **False Negatives (FN)** |
| --- | --- | --- | --- | --- | --- | --- |
| BERT-base (Strategy 1) | 0.00 | 0.00 | 0.00 | 0 | 0 | 182 |
| BERTweet (Strategy 1) | 0.00 | 0.00 | 0.00 | 0 | 0 | 182 |
| RoBERTa-base (Strategy 1) | 0.05 | 0.02 | 0.03 | 5 | 179 | 23 |
| BERT-base (Strategy 2) | 0.48 | 0.32 | 0.38 | 80 | 117 | 10 |
| BERTweet (Strategy 2) | 0.53 | 0.34 | 0.42 | 94 | 104 | 9 |
| RoBERTa-base (Strategy 2) | 0.54 | 0.33 | 0.41 | 96 | 99 | 12 |

Models from Strategy 2, particularly RoBERTa-base and BERTweet, exhibited significantly higher precision, recall, and F1 scores in detecting hate language compared to those from Strategy 1. BERT-base from Strategy 2 also performed reasonably well, but Strategy 1 models struggled, failing to detect any instances of class 0 (hate language) correctly. The models in Strategy 1 had a high number of false negatives, missing all class 0 instances, which is crucial in hate speech detection tasks. In contrast, Strategy 2 models substantially reduced false negatives, with BERTweet and RoBERTa-base achieving the lowest counts. Furthermore, BERTweet and RoBERTa-base from Strategy 2 not only had the highest true positives but also relatively fewer false positives, indicating a more accurate detection capability.

The significant difference in performance between Strategy 1 and Strategy 2 can be attributed to the distinct approaches in training the transformer models. Strategy 1 involved freezing the layers of the transformer model and only training a single-layer classifier with three neurons. This approach resulted in poor performance because the pre-training of these transformer models was not specifically focused on hate speech detection or offensive language.

In contrast, Strategy 2 trained the entire network, allowing the transformer model to learn the linguistic features pertinent to this specific task. Consequently, the transformer model was able to incorporate these features into its embeddings and encodings, resulting in a better differentiation between hate speech and offensive tweets.

As expected, BERT performed adequately among the three models. However, BERTweet, which had been pre-trained on tweets, showed better performance for this task. Ultimately, RoBERTa delivered the best performance in detecting offensive tweets, aligning with its reputation as a stronger model.

# 6. Conclusion and Future Work

Conducting this project has been an enriching experience, providing insights into hate speech detection using transformer models. The process included data collection, model selection, training, evaluation, and analysis. Key challenges involved selecting representative datasets and implementing BERT, BERTweet, and RoBERTa models.

Two training strategies were tested: Strategy 1 froze the transformer layers, while Strategy 2 fine-tuned the entire model. Evaluating these strategies showed the importance of fine-tuning for better performance in detecting hate speech.

The project's primary objective was to develop a reliable hate speech detection system. This was achieved by:

* **Model Evaluation:** Assessing performance using accuracy, precision, recall, and F1 score.
* **Confusion Matrix Analysis:** Understanding classification strengths and weaknesses.
* **Strategy Comparison:** Demonstrating the effectiveness of fine-tuning (Strategy 2) over freezing layers (Strategy 1).

Future work could enhance the project by:

* **Advanced Models:** Using newer transformer models for improved accuracy.
* **Larger Datasets:** Expanding training datasets for better generalizability.
* **Longer Training:** Training over more epochs for refined detection.
* **Multilingual Support:** Adding support for multiple languages.
* **Real-World Testing:** Applying models in real-world scenarios for practical insights.
* **Model Interpretability:** Increasing transparency in model decision-making.

In conclusion, this project successfully developed and evaluated effective hate speech detection models, laying a foundation for future advancements with more sophisticated AI and larger datasets.

# References

|  |  |
| --- | --- |
| 1. | Alkomah, F. and Ma, X., 2022. A literature review of textual hate speech detection methods and datasets. Information, 13(6), p.273. |
| 2. | Devlin, J., Chang, M.W., Lee, K. and Toutanova, K., 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805. |
| 3. | Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L. and Stoyanov, V., 2019. Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692. |
| 4. | Nguyen, D.Q., Vu, T. and Nguyen, A.T., 2020. BERTweet: A pre-trained language model for English Tweets. arXiv preprint arXiv:2005.10200. |
| 5. | Davidson, T., Warmsley, D., Macy, M. and Weber, I., 2017, May. Automated hate speech detection and the problem of offensive language. In Proceedings of the international AAAI conference on web and social media (Vol. 11, No. 1, pp. 512-515). |
| 6. | Naseem, U., Razzak, I. and Eklund, P.W., 2021. A survey of pre-processing techniques to improve short-text quality: a case study on hate speech detection on twitter. Multimedia Tools and Applications, 80, pp.35239-35266. |

1. EDA [↑](#footnote-ref-1)