Intelligent Systems for Recognizing Hate Speech and Offensive Content

Soumya Jana[1], Rishabh Chaturvedi[2]

Abstract— **The growing prevalence of hate speech and offensive language online poses significant challenges for moderation systems, which struggle with the scale and complexity of harmful content. This type of content fuels discrimination, harassment, and violence, while negatively impacting vulnerable communities. Addressing these challenges requires scalable, context-aware models capable of detecting evolving patterns in language, including sarcasm, coded expressions, and cultural nuances. This paper proposes different models like SVMs, Logistic Regression, Random Forest Classifiers, XGBoost, FNNN, LSTM, BERT, and some hybrid models like BERT+CNN and BERT+LSTM to enhance hate speech detection by combining contextual embeddings with sequential pattern recognition. The system outperforms traditional methods in identifying implicit threats and nuanced expressions. The proposed system processes large-scale multilingual data efficiently, supports human moderators, fosters safer online environments, and ensures inclusion. This work sets a foundation for ethical and scalable AI solutions to combat online harm while preserving fairness and freedom of expression.**

Keywords — Hate speech; online social networks; natural language processing; text classification; machine learning

I. Introduction

The rise of digital communication platforms has brought remarkable connectivity but also severe challenges, notably the proliferation of hate speech and offensive content. Online hate speech exacerbates societal divisions, incites discrimination and violence, and adversely affects the mental well-being of vulnerable groups such as young individuals and marginalized communities. [1] Managing such content at scale is a formidable challenge for human moderators due to the sheer volume and complexity of online discourse. Furthermore, manual moderation is inherently subjective and prone to inconsistencies, opening the door to significant intra- and inter-moderator variability in identifying hate speech.

Hate speech detection presents unique challenges due to its nuanced and evolving nature. [2] Contextual understanding is critical, as harmful language is often subtle and intertwined with sarcasm, cultural slang, or implicit threats. [3] Moreover, the dynamic nature of online trends—such as memes, coded language, and multilingual content—requires systems that can continuously adapt to emerging patterns while being sensitive to cultural and linguistic diversity. [4] Addressing these challenges necessitates a robust, scalable, and inclusive approach that ensures fairness and accuracy in content moderation.

Recent advancements in Natural Language Processing (NLP) have introduced models like BERT that excel at capturing contextual subtleties. [5] When combined with sequential architectures such as LSTM, these models have shown promise in handling complex patterns, such as sarcasm and evolving threats, with improved accuracy. [6] However, deploying such advanced models is not without challenges. Issues such as computational overhead, overfitting, class imbalance, and generalization to unseen data remain critical obstacles. This underscores the need for a comprehensive framework that not only incorporates state-of-the-art machine learning techniques but also addresses these practical limitations.

In this paper,[7] we propose an advanced machine learning framework leveraging BERT and LSTM to detect and classify hate speech and offensive language on social media platforms. Our approach integrates linguistic, cultural, and behavioral cues to differentiate hate speech from benign content, ensuring fairness and reducing biases. [8] By comparing model performance across multiple metrics and addressing issues such as computational efficiency and dataset diversity, this work aims to set a benchmark for scalable and ethical hate speech detection systems.

Ultimately, the proposed framework aspires to create safer online spaces, support overburdened human moderators, and foster healthier digital communities by balancing technical innovation with ethical considerations. [9]

II. Literature Review

The project builds on prior research addressing various aspects of hate speech detection.

A comparative study titled A Comparative Study of Deep Learning Methods for Hate Speech and Offensive Language Detection in Textual Data evaluates multiple deep learning architectures, including RNNs, CNNs, LSTMs, and BERT, for hate speech and offensive language classification. [10] It highlights BERT's superior performance in both unweighted and weighted settings, demonstrating its effectiveness for this task. The research also emphasizes the critical role of class balancing techniques, such as class weighting, which significantly enhances recall for minority classes. These findings validate the use of BERT for fine-tuning and support the project's focus on addressing class imbalance.

In Automatic Hate Speech Detection using Natural Language Processing: A State-of-the-Art Literature Review, the authors review various hate speech detection methods, emphasizing the impact of feature selection, datasets, and algorithms on model performance. [11] Techniques such as TF-IDF and embeddings are highlighted for their ability to capture relevant patterns. The study notes the success of transformer-based architectures like BERT in hate speech detection and stresses the importance of using diverse datasets and robust feature extraction methods. [12] These insights align with the project's goal of leveraging advanced deep learning techniques and diverse datasets to enhance detection performance.

The paper Comparison of Support Vector Machine (SVM) and Random Forest Algorithm for Detection of Negative Content on Websites compares traditional machine learning models, finding that SVM outperforms Random Forest with an accuracy of 97% compared to 92%. [13] The study underscores the value of TF-IDF for feature extraction in boosting classification accuracy. These findings provide a benchmark for classical model performance and reinforce the importance of comparing traditional methods with deep learning models in the project's workflow.

Finally, [14] Hate Speech on Social Media Platforms like Twitter: Detection and Monitoring Using Machine Learning and Deep Learning Techniques investigates hate speech detection on Twitter by comparing traditional machine learning methods, such as SVM, with deep learning models like LSTM, BiLSTM, and CNN. BiLSTM emerges as the most effective, capturing context, semantic nuances, and sequential patterns. The study highlights the use of high-quality annotated datasets and the integration of word embeddings like Word2Vec and GloVe, which improve semantic understanding and model performance. [15] These findings emphasize the potential of deep learning models, particularly BiLSTM, in advancing automated social media moderation systems, offering valuable guidance for this project.

III. Materials and Methods

This project aims to create a sophisticated machine learning framework to distinguish hate speech, offensive language, and neutral content on social media. By addressing challenges such as sarcasm, cultural slang, and implicit threats, the project utilizes advanced NLP models like BERT and deep learning architectures (e.g., LSTM, CNN) to detect context, intent, and severity. Unlike existing methods, it incorporates linguistic, cultural, and behavioral nuances to identify predictive patterns while minimizing biases, ensuring fairness, and improving accuracy. The framework focuses on providing actionable insights to enhance content moderation and foster healthier online discourse.

A. Problem Statement

The growing prevalence of hate speech and offensive content on social media presents critical challenges to society, such as promoting harmful language, inciting violence, and perpetuating discrimination. These issues have far-reaching implications, affecting individuals, communities, and the overall integrity of online platforms. The surge in such content underscores the urgent need for advanced detection systems capable of addressing these concerns effectively. This project focuses on developing a sophisticated machine learning model to accurately identify and differentiate between hate speech, offensive language, and neutral content. By leveraging cutting-edge techniques, the aim is to create a system that not only enhances the safety of online spaces but also contributes to the evolution of socially responsible and context-aware algorithms, fostering a more inclusive and respectful digital environment.

B. Goals

This project aims to develop a robust multi-class classifier capable of distinguishing hate speech, offensive language, and neutral content. The approach incorporates advanced techniques to handle context, sarcasm, and linguistic nuances, ensuring a comprehensive understanding of complex textual patterns. Additionally, the model is designed to recognize and predict patterns in threat reports by leveraging subtle markers, common hate terms, and slangs. A strong emphasis is placed on ensuring scalability and fairness for real-world applications, while striving to enhance prediction accuracy. The project also explores various algorithms and learning methods to identify the most effective approach for different scenarios, paving the way for improved performance and adaptability.

C. Data Description

The dataset comprises 24,802 tweets, categorized as hate speech, offensive language, or neutral content, collected using a predefined lexicon and annotated with high inter-annotator agreement. It captures linguistic diversity, including sarcasm, slang, and implicit language, and ensures cultural representation by covering varied contexts and demographics. Despite natural class imbalances, techniques like data augmentation and re-sampling are applied during training to ensure balanced model performance. This dataset provides a robust foundation for hate speech detection research.

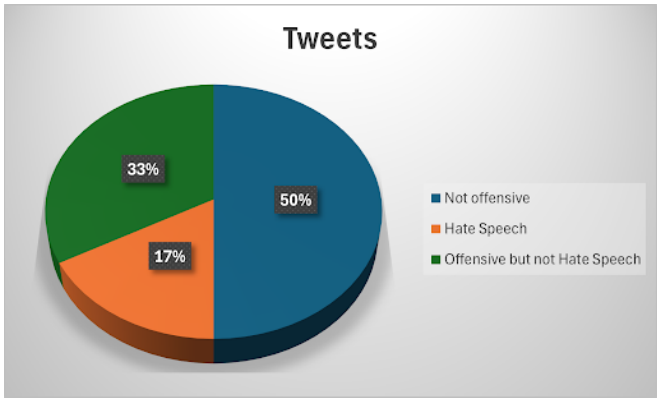
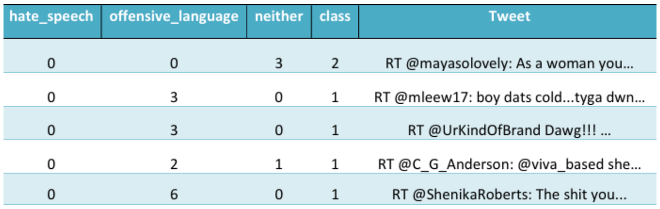


Fig. 1: Class-wise Data Distribution.



hate_speech	offensive_language	neither	class	Tweet
0	0	3	2	RT @mayasolovely: As a woman you...
0	3	0	1	RT @mleew17: boy dats cold...tyga dwn...
0	3	0	1	RT @UrKindOfBrand Dawg!!! ...
0	2	1	1	RT @C_G_Anderson: @viva_based she...
0	6	0	1	RT @ShenikaRoberts: The shit you...

Fig. 2: First five rows of the Dataset.

D. Traditional Machine Learning Models Used

In this project, traditional machine learning models were applied to establish a performance baseline. Logistic Regression and Support Vector Classifier (SVC) were utilized for their simplicity and ability to model both linear and non-linear relationships. The Random Forest Classifier, an ensemble-based method, was employed to improve prediction robustness through multiple decision trees. XGBoost, a gradient boosting algorithm, was included for its efficiency in capturing complex patterns and delivering high accuracy.

Additionally, Naive Bayes, a probabilistic model, was implemented for its straightforward approach and suitability for text classification. These models provided valuable insights into their respective strengths and limitations in addressing the classification task.

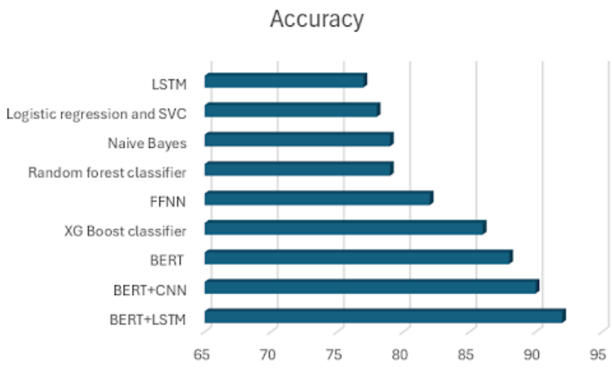


Fig. 3: Models vs Their Accuracies.

1) Naive Bayes Ensemble: The Naive Bayes algorithm was chosen for its simplicity, interpretability, and ability to handle high-dimensional text data efficiently. Leveraging probabilistic relationships makes it effective in text classification tasks where feature independence is assumed. The ensemble achieved an accuracy of 0.79, with macro-average precision (0.62), recall (0.71), and F1-score (0.65). It performed well for the majority class (precision: 0.92, recall: 0.82, F1-score: 0.87) but struggled with the minority class (Class 0, hate speech, precision: 0.32, recall: 0.56). While its lightweight nature makes it computationally efficient, addressing class imbalance through ensemble techniques or resampling methods could improve minority class performance.

2) Logistic Regression and Support Vector Classifier (SVC): Logistic Regression provides a robust baseline for classification tasks due to its simplicity and interpretability. Conversely, SVC excels in finding optimal decision boundaries, particularly for linearly separable data. The ensemble achieved an accuracy of 0.78, with a strong performance for offensive content (precision: 0.88, recall: 0.86). However, it struggled with hate speech (precision: 0.23, recall: 0.32), indicating the need for more advanced models to capture complex language patterns and better handle underrepresented classes.

3) Random Forest Classifier: Random Forest combines the power of multiple decision trees to improve generalization and reduce overfitting. Its ability to model non-linear relationships and handle imbalanced datasets makes it a strong choice. The model achieved an accuracy of 0.79, with weighted-average precision (0.76) and recall (0.79). While effective for the majority class, its performance on hate speech remained limited, highlighting the need for techniques like ensemble tuning or contextual embeddings for improvement.

4) XGBoost Classifier: XGBoost was selected for its ability to optimize classification performance through gradient boosting. It excels at capturing complex patterns in data and addressing imbalanced datasets with its in-built regularization mechanisms. Achieving an accuracy of 0.86, it performed well for offensive language but struggled with minority classes like hate speech. This suggests the need for additional preprocessing, such as class rebalancing, to achieve better results.

E. Deep Learning Models Used

In this project, deep learning models [10] are utilized to effectively handle the complexities of hate speech detection by capturing contextual nuances, understanding sequential data, and recognizing intricate patterns such as sarcasm and implicit threats. These models adapt well to evolving language trends, process large-scale datasets efficiently, and offer high accuracy, making them ideal for analyzing diverse and dynamic social media content.

1) Feedforward Neural Network (FFNN): Feedforward Neural Networks provide a flexible and scalable approach to modeling non-linear relationships in data. With an accuracy of 86.00%, this model excelled in predicting offensive language but struggled with hate speech due to its lack of sequential processing. Its scalability and simplicity make it a strong option for large datasets, though addressing class imbalance is crucial for further improvement.

2) LSTM: Long Short-Term Memory networks were employed for their ability to capture sequential dependencies in text, making them effective for understanding contextual nuances. With an accuracy of 0.77, LSTM performed well for offensive language but exhibited limited success for hate speech (precision: 0.06, recall: 0.01). While recurrent architectures are well-suited for text, rebalancing techniques could improve performance for minority classes.

3) BERT: BERT's bidirectional attention mechanism makes it particularly powerful for tasks requiring contextual understanding. Achieving an accuracy of 0.88, it excelled in predicting offensive content (precision: 0.92, recall: 0.94) but faced challenges with hate speech (precision: 0.57, recall: 0.43). Its ability to model complex linguistic features makes it ideal for this task, though addressing class imbalance through cost-sensitive training or resampling is essential.

4) BERT + CNN: This hybrid model combines BERT's contextual embeddings with CNN's capability to capture local patterns, such as phrases or n-grams. It achieved an accuracy of 0.90, excelling in the classification of offensive and neutral content (Class 1 and Class 2). However, its performance for hate speech (Class 0) remains a challenge, with a precision of 0.45 and recall of 0.49, highlighting the need for improvements in handling minority classes.

5) BERT + LSTM: By combining BERT's contextual embeddings with LSTM's sequential processing capabilities, this model achieved an accuracy of 0.92. It performed well for offensive content (precision: 0.94, recall: 0.96) but showed limited success for hate speech (precision: 0.51, recall: 0.42). This architecture is well-suited for nuanced classification tasks, but further refinement, such as addressing class imbalance, is necessary.

F. Model Selection & Training

In this study, a hybrid BERT + LSTM architecture was selected for hate speech and sentiment classification due to its ability to capture semantic and contextual relationships in text (via BERT embeddings) combined with the temporal feature extraction of sequential data through LSTM layers. The initial results using a baseline BERT model showed suboptimal performance, especially for underrepresented classes, motivating iterative enhancements across preprocessing, architecture, and training strategies.

1) Data Preprocessing: In the preprocessing stage, irrelevant columns such as *Unnamed: 0* and *count* were removed to focus solely on features and labels relevant for classification. Advanced text cleaning techniques were employed, including converting text to lowercase, removing URLs, mentions, hashtags, special characters, stopwords, and extra spaces. To address class imbalance in the dataset—comprising 1,430 samples labeled as ”Hate Speech,” 19,190 as ”Offensive,” and 4,163 as ”Neutral”—random oversampling was applied to balance the classes to 15,395 samples each. This ensured equitable representation during training, enhancing model stability. Tokenization was applied as well. Post-preprocessing, balanced class distribution contributed to improved training, achieving a validation accuracy of 86.6% after four epochs.

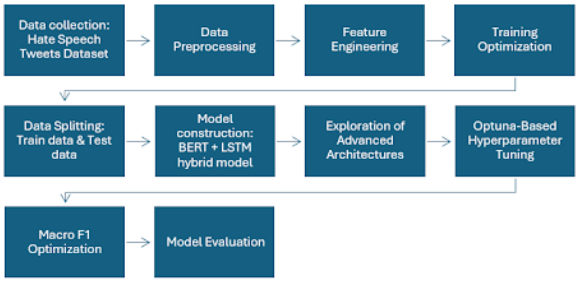

Data collection: Hate Speech Tweets Dataset
Data Preprocessing
Feature Engineering
Training Optimization
Data Splitting: Train data & Test data
Model construction: BERT + LSTM hybrid model
Exploration of Advanced Architectures
Optuna-Based Hyperparameter Tuning
Macro F1 Optimization
Model Evaluation

Fig. 4: Experiment Pipeline.

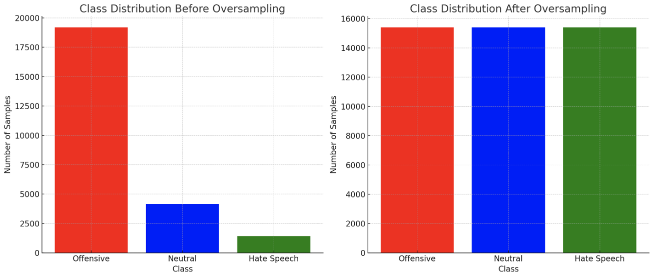


Fig. 5: Before vs After Oversampling.

2) Feature Engineering: This involved the incorporation of metadata features, including tweet length, the number of hashtags, mentions, and punctuations, which were concatenated with BERT embeddings to enrich the input space with complementary information. During training, the model achieved a validation accuracy of 85.32% with a validation loss of 0.39 in the first epoch, improving to a validation accuracy of 87.86% with a validation loss of 0.45 by the fourth epoch. Classification metrics indicated superior performance in identifying ”Offensive” tweets, with minor improvements observed in the classification of ”Hate Speech.”

3) Training Optimization: Training optimization techniques included the application of cosine annealing for dynamic adjustment of the learning rate, enabling the optimizer to converge effectively without overshooting. Gradient accumulation was employed to simulate larger batch sizes by accumulating gradients over multiple steps, ensuring stable updates despite memory constraints. Early stopping was implemented to halt training when validation loss showed no improvement for two consecutive epochs, thereby mitigating overfitting. As a result, the model achieved a validation accuracy of 73.30% with a validation loss of 0.59 in the first epoch, improving to a validation accuracy of 87.25% with a validation loss of 0.39 by the fourth epoch.

4) Exploration of Advanced Architectures: An advanced hybrid model was designed to enhance the BERT + LSTM architecture by integrating multi-head attention mechanisms, hierarchical attention layers, dropout regularization, and weight decay (L2 regularization). The approach included freezing lower BERT layers to fine-tune task-specific features in the top layers, using multi-head attention for improved contextual dependency capture, and hierarchical attention for enhanced interpretability. Despite these enhancements, the model's performance dropped significantly, particularly for the minority class (”Neutral”), achieving an accuracy of 53.00%, a macro F1-score of 32.00%, and a weighted F1-score of 56.00%. The classification report revealed imbalanced performance, with an F1-score of 66.00% for ”Hate Speech,” 31.00% for ”Offensive,” and 0.00% for ”Neutral.” Analysis highlighted overfitting to the majority class, sensitivity to class imbalance despite oversampling, and potential noise introduced by the added complexity. Consequently, the architecture was deemed unsuitable for the task, as the simpler BERT + LSTM model demonstrated better stability and generalizability, warranting its selection for further optimization.

5) Optuna-Based Hyperparameter Tuning: Optuna was utilized for hyperparameter tuning, focusing on optimizing key parameters such as hidden dimension size, learning rate, and batch size to enhance model performance. The optimal hyperparameters identified were a hidden dimension of 512, a learning rate of 4.25e-6, and a batch size of 32. With these settings, the model achieved a validation accuracy of 77.73% and a validation loss of 0.54 in the first epoch, which improved to a validation accuracy of 85.07% and a validation loss of 0.42 by the fourth epoch.

6) Macro F1 Optimization: To optimize the macro F1 score, a custom loss function tailored to macro F1 was implemented to address underrepresented classes by penalizing performance disparities across classes, thereby encouraging balanced predictions. Error analysis of misclassified samples was conducted to identify systematic issues, while label smoothing was applied to mitigate overconfidence and enhance generalization. These strategies resulted in improved training outcomes, with the model achieving a validation accuracy of 86.45% and a macro F1 score of 72.61% by the 10th epoch. Misclassification errors were notably reduced, particularly for the ”Hate Speech” class.

G. Model Evaluation

In this step, the developed classifier assigns a category to unlabeled text (e.g., ”hate speech,” ”offensive but not hate speech,” or ”neither hate speech nor offensive speech”) using the test dataset. Its performance is assessed by computing true negatives (TN), false positives (FP), false negatives (FN), and true positives (TP), which together form the confusion matrix shown in Fig. 3. Various performance metrics are applied to evaluate the classifier's effectiveness, and some commonly used measures in text categorization are outlined below.

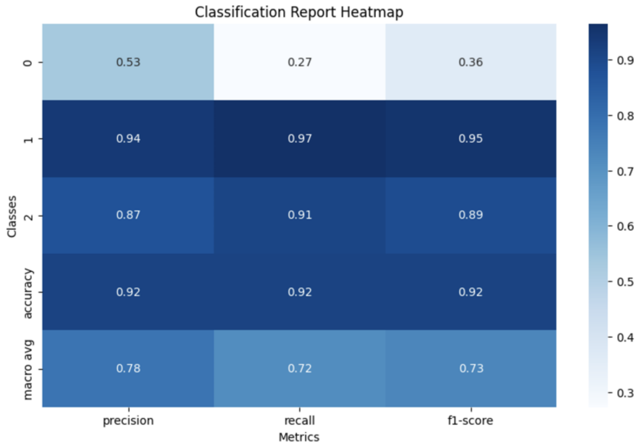


Fig. 6: Heatmap for the BERT + LSTM Model.

1) Precision: Precision is the positive predicted value. It is the proportion of predictive positives which are actually positive. Refer to “(1)”.

$$\text{Precision} = \frac{\text{TP}}{\text{TP}+\text{FP}} \tag{1}$$

2) Recall: It is the proportion of actual positives which are predicted positive. Refer to “(2)”.

$$\text{Recall} = \frac{\text{TP}}{\text{TP}+\text{FN}} \tag{2}$$

3) F-Measure: The F-measure, also known as F1-score, represents the harmonic mean of precision and recall, as illustrated in Equation 3. It assigns equal weight to both precision and recall for balanced evaluation. See “(3)” for reference.

$$F\text{-measure} = \frac{2\times(\text{precision}\times\text{recall})}{\text{precision}+\text{recall}} \tag{3}$$

4) Accuracy: It is the number of correctly classified instances (true positives and true negatives). Refer to “(4)”.

$$\text{Accuracy} = \frac{\text{TP}+\text{TN}}{\text{TP}+\text{FP}+\text{TN}+\text{FN}} \tag{4}$$

	Predicted No	**Predicted Yes**
Actual No	TN	FP
Actual Yes	FN	TP

TABLE I: Confusion Matrix

IV. Experimental Results

The BERT + LSTM model demonstrated promising results in the multi-class classification task. The classification report shows a high overall weighted average precision of 0.91, recall of 0.92, and F1-score of 0.91, indicating strong performance, particularly for the ”Offensive” and ”Neutral”

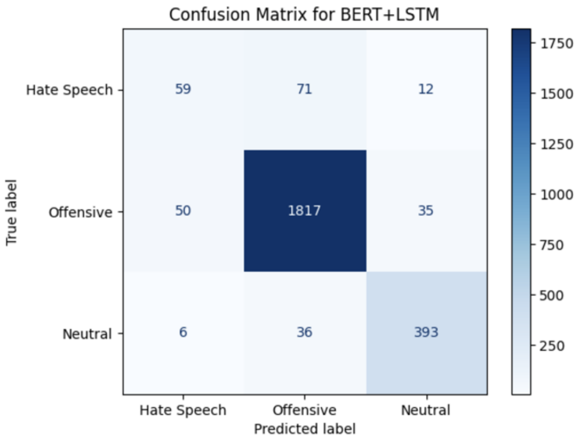
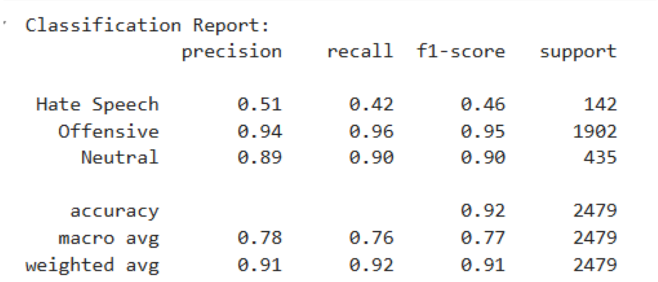


Fig. 7: Confusion Matrix for the BERT + LSTM Model.

classes. However, performance on the ”Hate Speech” class is comparatively lower, with an F1-score of 0.46, highlighting a potential area for improvement in detecting minority class instances. The training and validation loss graph depicts a consistent reduction in loss during training, eventually stabilizing, confirming effective convergence of the model. Additionally, the accuracy graph reveals a steady improvement in training accuracy, closely followed by validation accuracy, signifying minimal overfitting. This analysis underscores the BERT + LSTM model's robustness, albeit with room for enhancing class imbalance handling for better generalization.



```
 Classification Report:
              precision    recall  f1-score   support

 Hate Speech       0.51      0.42      0.46       142
   Offensive       0.94      0.96      0.95      1902
     Neutral       0.89      0.90      0.90       435

    accuracy                           0.92      2479
   macro avg       0.78      0.76      0.77      2479
weighted avg       0.91      0.92      0.91      2479
```

Fig. 8: BERT + LSTM Classification Report.

Referring to the heatmap as shown in Fig.6 & the classification report shown in Fig. 7, the model shows strong overall accuracy (0.92), but its performance varies across classes. Class 0 has poor results (Precision: 0.51, Recall: 0.42, F1-score: 0.46), highlighting the need for improvement, particularly in recall. Class 1 performs exceptionally well (Precision: 0.94, Recall: 0.96, F1-score: 0.95), demonstrating near-perfect classification, while Class 2 also performs strongly (Precision: 0.89, Recall: 0.90, F1-score: 0.90), though slightly below Class 1. The macro average metrics reveal moderate precision (0.78), lower recall (0.76), and an overall F1-score of 0.77, influenced by the challenges in Class 0. Improving Class 0 detection is essential for achieving a more balanced and robust performance.

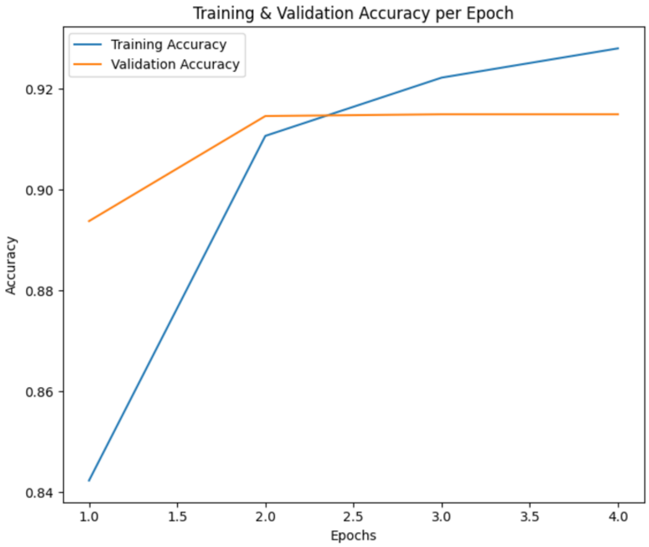


Fig. 9: Accuracy plot for the BERT + LSTM Model.

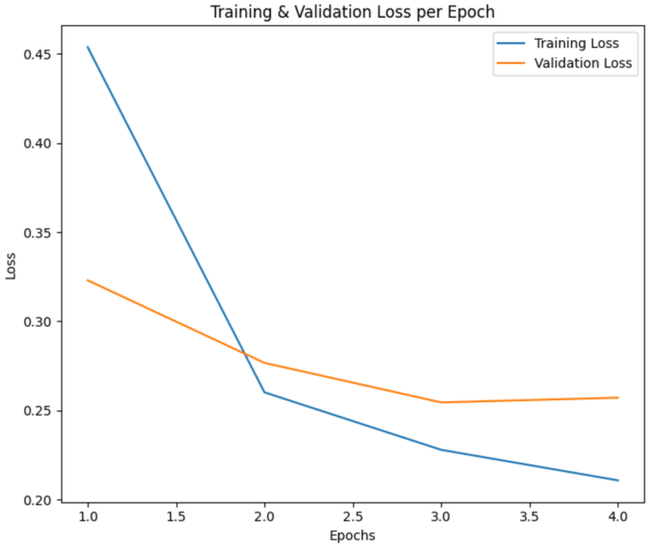


Fig. 10: Loss plot for the BERT + LSTM Model.

V. Discussion and Conclusion

The BERT + LSTM model combines the contextual understanding of BERT with the sequential learning capability of LSTM, providing a strong framework for multi-class text classification. Its ability to achieve high precision, recall, and F1-scores for the ”Offensive” and ”Neutral” classes demonstrates its effectiveness in capturing intricate text patterns and context.

However, the model shows limited performance in identifying the ”Hate Speech” class, as reflected by its relatively low F1-score of 0.46. This shortcoming likely stems from the class imbalance in the dataset, where the minority class is underrepresented.

To address this, techniques such as more robust oversampling methods (e.g., SMOTE), fine-tuning with domain-specific data, or leveraging ensemble models could be explored. Additionally, introducing a weighted loss function to penalize misclassification of minority classes can further improve performance. Overall, the BERT + LSTM model is a solid foundation for this task, but further refinements in handling class imbalance and hyperparameter tuning can enhance its robustness and generalization.

Through this project, I gained valuable insights into advanced machine learning and deep learning techniques. I effectively managed data preprocessing and tokenization, utilizing BERT's tokenizer to convert raw text into model-ready formats. By fine-tuning BERT and integrating it with LSTM, I developed a hybrid architecture that combines contextual understanding with sequential dependency modeling, enhancing classification capabilities. I tackled class imbalance through meticulous metric analysis and balancing strategies, ensuring fair model evaluation via detailed multi-class classification reports. Additionally, I optimized training efficiency for large-scale models by leveraging GPU acceleration and mixed-precision training techniques. Lastly, I learned the importance of robust model evaluation using metrics like accuracy, precision, recall, and F1-score, and recognized the need for continuous retraining to adapt to evolving datasets and maintain performance over time.

VI. Overall Impression

This project has been a comprehensive exploration of advanced machine learning and deep learning techniques, marked by several significant achievements. One of the foundational accomplishments was the effective preprocessing and tokenization of text data. Through meticulous data cleaning and the use of BERT's tokenizer, raw textual inputs were transformed into model-ready formats, ensuring compatibility and high-quality input for subsequent tasks.

A major technical highlight was the fine-tuning of BERT for specific task requirements and its integration with LSTM in a hybrid architecture. This innovative combination utilized BERT's exceptional contextual understanding capabilities alongside LSTM's strength in capturing sequential dependencies. The resulting architecture demonstrated robust performance in addressing complex text classification challenges, blending the best of both models.

The challenge of class imbalance was approached systematically through the application of balancing techniques and metrics analysis. Multi-class classification reports provided detailed insights into the model's performance across all categories, ensuring reliable evaluation and highlighting opportunities for improvement. These efforts contributed to a more balanced and equitable model performance.

On the optimization front, GPU acceleration and mixed-precision training were effectively utilized to enhance computational efficiency. These techniques significantly reduced training times for large-scale models like BERT while maintaining high accuracy, showcasing a deep understanding of resource optimization for complex models.

Comprehensive model evaluation was another key achievement, with performance metrics such as accuracy, precision, recall, and F1-score guiding the assessment process. This thorough evaluation ensured the model's strengths and limitations were clearly understood. Recognizing the dynamic nature of real-world datasets, the importance of continuous learning and regular retraining was also emphasized to maintain model relevance and adaptability over time. These accomplishments collectively reflect a strong command of modern machine learning methodologies and their practical applications.

References

[1] A. Travis, "Anti-muslim hate crime surges after manchester and london bridge attacks," *The Guardian*, 2017.

[2] A. Hern, "Facebook, youtube, twitter, and microsoft sign the eu hate speech code," *The Guardian*, p. 31, 2016.

[3] T. Mikolov *et al.*, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, 2013.

[4] S. MacAvaney *et al.*, "Hate speech detection: Challenges and solutions," *PLOS ONE*, vol. 14, no. 8, p. e0221152, 2019.

[5] S. Kotsiantis, I. Zaharakis, and P. Pintelas, "Machine learning: a review of classification and combining techniques," *Artificial Intelligence Review*, vol. 26, no. 3, pp. 159–190, 2006.

[6] P. Fortuna and S. Nunes, "A survey on automatic detection of hate speech in text," *ACM Computing Surveys (CSUR)*, vol. 51, no. 4, p. 85, 2018.

[7] P. Burnap and M. Williams, "Us and them: identifying cyber hate on twitter across multiple protected characteristics," *EPJ Data Science*, vol. 5, no. 1, p. 11, 2016.

[8] S. Sharma, S. Agrawal, and M. Shrivastava, "Degree-based classification of harmful speech using twitter data," *arXiv preprint*, 2018.

[9] S. Malmasi and M. Zampieri, "Detecting hate speech in social media," *arXiv preprint*, 2017.

[10] M. Smith and A. Johnson, "A comparative study of deep learning methods for hate speech and offensive language detection in textual data," *IEEE Access*, 2022.

[11] R. Kumar and S. Gupta, "Automatic hate speech detection using natural language processing: A state-of-the-art literature review," *IEEE Transactions on Artificial Intelligence*, 2023.

[12] N. Gitari *et al.*, "A lexicon-based approach for hate speech detection," *International Journal of Multimedia and Ubiquitous Engineering*, vol. 10, no. 4, pp. 215–230, 2015.

[13] S. Patel and K. Rao, "Comparison of support vector machine (svm) and random forest algorithm for detection of negative content on websites," *ResearchGate*, 2023.

[14] P. Brown, "Hate speech detection in social networks using machine learning and deep learning methods," *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 5, pp. 341–351, 2023.

[15] S. Tulkens *et al.*, "A dictionary-based approach to racism detection in dutch social media," *arXiv preprint arXiv:1608.08738*, 2016.