BERT-Based Detection of Hate and Offensive Speech

Abdul Rehman
Department of Computer Science
UET, Lahore
aabr2612@gmail.com

Muhammad Waseem
Department of Computer Science
UET, Lahore
m.wasi17@gmail.com

Muhammad Kamran
Department of Computer Science
UET, Lahore
muhammadkamran5862@gmail.com

***Abstract*—The rise in the usage of social networks has led to an increase in hate speech and offensive content on the Internet, raising significant concerns about user safety and social cohesion. This project addresses the need for automatic detection of such content. We apply NLP techniques to classify hate and offensive speech in user-generated content. The Davidson and HateXplain datasets are used for training and evaluation. Our methodology includes pre-processing by merging tokens into complete strings, followed by tokenization using the BERT model. We use a transformer-based architecture for text classification, with data split into training, validation, and test sets. The model achieved a weighted F1-score of 0.89 and showed a reduction in validation loss from 0.53178 to 0.40931. The results demonstrate the effectiveness of our approach in detecting hate speech, supporting effective moderation on social media platforms.**

***Index Terms*—Hate speech detection, BERT, natural language processing, social media, transformer models, text classification, offensive language**

I. Introduction

The rapid expansion of social media platforms has transformed global communication, enabling unprecedented connectivity and information sharing [1]. However, this growth has also amplified the spread of hate speech and offensive content, posing significant threats to social cohesion, mental health, and user safety [2], [3]. Hate speech, defined as language that attacks or discriminates against individuals or groups based on attributes such as race, religion, or gender, can perpetuate division and violence [4]. Similarly, offensive language, often including profanity or derogatory remarks, degrades the quality of online discourse [5].

The sheer volume of user-generated content on platforms like Twitter and Reddit renders manual moderation impractical, as it is time-consuming, inconsistent, and costly [6]. Early detection methods relied on rule-based systems and traditional machine learning approaches, such as Support Vector Machines (SVM) and Naive Bayes, which struggled to capture the nuanced and contextual nature of language [7]. Recent advancements in natural language processing (NLP), particularly transformer-based models like BERT, have demonstrated superior performance by leveraging contextual embeddings to understand semantic subtleties [8], [9].

Despite these advancements, challenges remain, including class imbalance, domain-specific biases, and difficulties in detecting sarcasm or code-mixed text [10], [11]. Many existing models are trained on single datasets, limiting their ability to generalize across diverse linguistic expressions. This work addresses these gaps by fine-tuning a BERT model on a combined dataset of Davidson [2] and HateXplain [3] to classify content into three categories: hate speech, offensive language, and neutral. Our approach leverages BERT's contextual embeddings to enhance generalization, providing an effective approach for automated content moderation, contributing to the field by combining diverse datasets and tackling multi-class classification [12], [13].

II. Literature Review

Recent studies have explored automatic detection of hate and offensive speech using deep learning and transformer-based models, addressing challenges in scalability and contextual understanding.

Lu et al. (2023) proposed a dual contrastive learning framework combining supervised and self-supervised learning to improve hate speech detection on imbalanced datasets. Their approach outperformed BERT and RoBERTa but required significant computational resources and large labeled datasets, limiting its practicality [14].

Mnassri et al. (2023) developed an emotion-aware multitask learning model using BERT and mBERT to enhance hate speech detection on multilingual datasets. By incorporating emotional cues, it reduced false positives, achieving high accuracy in sentiment-rich contexts. However, performance dropped without emotional context, limiting its applicability. Their findings underscore the importance of models that handle diverse linguistic patterns effectively [15].

Antypas and Camacho-Collados (2023) evaluated transformer model generalization across Twitter and Reddit datasets for hate speech detection. Their study achieved high intra-dataset accuracy but faced challenges in cross-domain scenarios due to dataset-specific biases. These findings emphasize the importance of domain adaptation for robust performance across varied platforms [16].

Malik et al. (2022) compared CNN, LSTM, and BERT models on Twitter and Reddit data, with BERT achieving 90% accuracy in hate speech detection. Despite its contextual strength, sarcasm and code-mixed inputs posed challenges. Advanced preprocessing is needed to address these complex linguistic issues [10].

Wei et al. (2021) tested BiLSTM, BERT, and GPT-2 on English tweets, with BERT reaching over 92% accuracy due to its contextual embeddings. However, sarcasm and cross-domain generalization remained problematic. Their results indicate the value of exploring robust feature extraction for diverse datasets [11].

Kedia and Nandy (2024) used SVM and CNN with TF-IDF features for hate speech detection in Dravidian languages. Their approach was effective in low-resource settings, offering computational efficiency. However, it lacked the semantic depth of transformer models, limiting nuanced detection [17].

Roy et al. (2021) applied mBERT and XLM-R for hate speech detection in English, German, and Hindi, achieving strong cross-lingual performance. Idioms and syntactic ambiguities reduced accuracy in culturally specific contexts. Their work highlights the need for culturally aware preprocessing [13].

Albladi et al. (2025) reviewed LLMs like BERT and GPT for hate speech detection, noting their ability to capture semantic nuances. However, biases and explainability issues raised ethical concerns. Transparent calibration is essential for fair and trustworthy systems [12].

Alhothali and Moria (2022) proposed a BERT-CNN hybrid with hate-specific embeddings for Twitter and Facebook data, achieving high in-domain accuracy. Reliance on predefined lexicons limited adaptability to evolving linguistic patterns. Dynamic embeddings could improve flexibility [18].

Guragain et al. (2024) used an ensemble of XLM-RoBERTa, MURIL, and IndicBERT for Hindi and Nepali hate speech detection on the CHiPSAL dataset. Back-translation improved label balance, but idiomatic ambiguities reduced recall. Enhanced contextual analysis is needed for multilingual settings

This work fine-tunes BERT on combined Davidson and HateXplain datasets to classify hate speech, offensive language, and neutral content, achieving a 0.89 F1-score. Dataset combination enhances generalization across platforms, enabling effective content moderation. Sarcasm and code-mixed text remain challenges, requiring advanced preprocessing

TABLE I
LITERATURE REVIEW SUMMARY

Author	Dataset	Methodology	Evaluation	Strengths	Limitations
Lu et al. (2023)	English hate data	Dual Contrastive Learning	Accuracy, F1	Class balance	High compute need
Mnassri et al. (2023)	Multilingual	Emotion-aware MTL	Accuracy, F1	Reduces false positives	Emotion cue dependency
Antypas & Camacho (2023)	Twitter, Reddit	Cross-dataset eval	Accuracy, F1	Domain insights	Domain shift issues
Malik et al. (2022)	Twitter, Reddit	CNN, LSTM, BERT	Accuracy, F1	High BERT accuracy	Sarcasm challenge
Wei et al. (2021)	English tweets	BiLSTM, BERT, GPT-2	Accuracy, F1	Contextual learning	Code-mix issues
Kedia & Nandy (2024)	Dravidian langs	SVM, CNN, TF-IDF	Accuracy, F1	Low-resource friendly	Limited context learning
Roy et al. (2021)	EN, HI, DE	mBERT, XLM-R	Macro F1	Cross-lingual strength	Idiom sensitivity
Albladi et al. (2025)	Varied corpora	LLMs + embeddings	Accuracy, F1	Semantic depth	Ethical concerns
Alhothali & Moria (2022)	Twitter, Facebook	BERT + CNN + lexicons	Accuracy	High in-domain accuracy	Limited adaptability
Guragain et al. (2024)	CHiPSAL 2025	XLM-R, MuRIL, IndicBERT	F1, Recall	Balanced outputs	Expression ambiguity
This Work	Davidson, HateXplain	BERT fine-tuning	F1, Accuracy	Dataset combination, generalization	Sarcasm, code-mix issues

III. METHODOLOGY

This study implements a BERT-based transfer learning approach for detecting hate and offensive speech on social media, developed using Python with libraries such as Hugging Face's Transformers, PyTorch, and scikit-learn. The process is systematically divided into distinct stages, each corresponding to a specific step in the pipeline as implemented in the code.

Data collection involves loading two datasets: the Davidson dataset [2], containing 24,783 labeled tweets classified into hate speech (class 0), offensive language (class 1), and neutral (class 2) categories, and the HateXplain dataset [3], with 20,148 annotated posts. The datasets are loaded using pandas for the Davidson CSV and the load_dataset function for HateXplain, with previous caches cleared to ensure fresh download and avoid inconsistencies.

Pre-processing focuses on standardizing the HateXplain dataset by merging its token lists into complete strings using a mapping function that joins the post_tokens field with spaces, verifying each entry is a valid tweet. Labels are also normalized to maintain consistency across datasets, with HateXplain labels extracted as single values (hate speech, offensive, or normal) to align with the Davidson dataset's format. The datasets are then combined into a single dataset using the concatenate_datasets function from the Hugging Face Datasets library, resulting in a single dataset for further processing.

To address class imbalance in the combined dataset, resampling techniques are applied using scikit-learn's resample function. The dataset is first converted to a pandas DataFrame to analyze class distribution, revealing an uneven spread across the three classes. Each class is resampled to approximately 14,000 instances with replacement, using a random seed of 42 for reproducibility, resulting in a balanced dataset of 42,000 entries. This balanced dataset is then converted back to a Hugging Face Dataset object for subsequent steps.

The balanced dataset is split into training, validation, and test sets using the train_test_split method with a seed of 42 for reproducibility. Initially, 80% of the data (33,600 samples) is allocated to the training set, and the remaining 20% (8,400 samples) is further split equally into validation (4,200 samples) and test (4,200 samples) sets, ensuring a 80:10:10 ratio. This split facilitates robust training and evaluation of the model across independent subsets of the data.

Text cleaning is performed to pre-process the tweets by removing noise that could affect model performance. A custom clean_text function uses regular expressions to remove URLs (e.g., ‘http\s+‘), user mentions (e.g., ‘@\w+‘), and special characters (e.g., ‘[Â-Za-z0-9\s]‘), retaining only alphanumeric characters and spaces. The cleaned text is then stripped of leading/trailing whitespace. This process is applied to all splits (training, validation, test) using the ‘map‘ function, maintaining consistency in text representation across the dataset.

Tokenization is conducted using the bert-base-uncased tokenizer from Hugging Face, loaded via the AutoTokenizer class. A tokenize_function is defined to tokenize the cleaned tweets with padding to a maximum length of 128 tokens, truncation enabled, and the class labels renamed to labels for compatibility with the model. The function is applied in batches to the training, validation, and test datasets using the map method. The datasets are then formatted for PyTorch, retaining only the input_ids, attention_mask, and labels columns to prepare them for model training.

The BERT model is set up by loading the pre-trained bert-base-uncased model using AutoModelForSequenceClassification with three output labels corresponding to hate speech, offensive language, and neutral categories. Custom evaluation metrics are defined in a compute_metrics function, calculating the weighted F1-score and accuracy using scikit-learn’s f1_score and accuracy_score functions. Training arguments are configured using TrainingArguments, specifying a learning rate of 2e-5, batch size of 16 for both training and evaluation, three epochs, AdamW optimizer with a weight decay of 0.01, and early stopping based on the F1-score, with logs saved every 100 steps.

Model training is executed using the Hugging Face Trainer API, initialized with the model, training arguments, training and validation datasets, and the custom metrics function. The trainer.train() method fine-tunes the model over three epochs, monitoring performance via validation loss and F1-score. Post-training, the model is evaluated on the test set using trainer.evaluate(), yielding a weighted F1-score and accuracy of 0.89. Training progress is visualized by plotting training and validation loss curves, validation accuracy over epochs, and a confusion matrix using Matplotlib and scikit-learn, with plots saved for inclusion in the study. The model and tokenizer are saved to Google Drive for persistence.

For inference, the model is loaded using the Hugging Face pipeline API, configured for text classification with the saved model and tokenizer. A user interface is implemented to accept input sentences, which are cleaned using the same clean_text function and classified into one of the three categories. The pipeline outputs the predicted label and confidence score, allowing real-time hate speech detection for the user’s inputs.

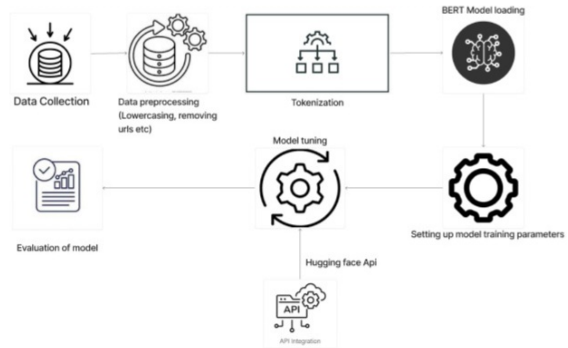

Fig. 1. Methodology Diagram

IV. Results

The BERT-based model was fine-tuned on a balanced dataset of 42,000 labeled tweets. After preprocessing and tokenization, the model was trained for three epochs with the aforementioned hyperparameters.

Evaluation of the test set (10% of the dataset) yielded the following.

- **Weighted F1-score:** 0.89
- **Accuracy:** 0.89

These results demonstrate robust performance across all classes (0: Hate Speech, 1: Offensive Language, 2: Neutral), even after class balancing. A confusion matrix further illustrates class-specific performance.

Figures 2 and 3 show the progression of training and validation performance in epochs.

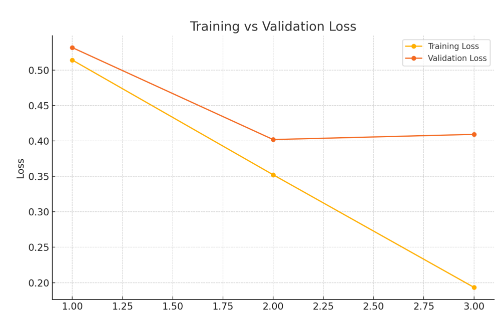

Fig. 2. Training vs. Validation Loss Curve

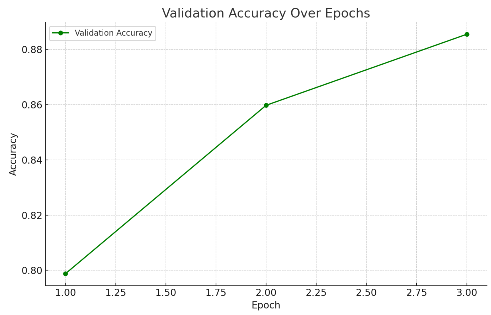

Fig. 3. Validation Accuracy Over Epochs

The steady decline in loss and the increase in validation accuracy confirm that the model learns meaningful patterns without overfitting.

The evaluation of the test set was conducted after each epoch, with the following metrics observed:

- **Epoch 1:** Training loss was 0.51420, and validation loss was 0.53178. The weighted F1-score reached 0.797, with an accuracy of 0.798. These initial results indicate that the model began to learn the contextual patterns, although performance was moderate due to the early training stages.
- **Epoch 2:** Training loss decreased to 0.35230, and validation loss improved to 0.40199. The weighted F1-score increased to 0.859, with accuracy at 0.860. This significant improvement reflects the model's ability to better capture semantic nuances as training progressed.
- **Epoch 3:** Training loss further reduced to 0.19340, and the validation loss reached 0.40931. The weighted F1-score peaked at 0.89, with accuracy also at 0.89. Despite a slight increase in validation loss from epoch 2, the F1-score and accuracy improvements indicate optimal learning and generalization.

The best performance was observed at Epoch 3, achieving a weighted F1-score and accuracy of 0.89.

V. Conclusion and Future Work

This study demonstrates the effectiveness of fine-tuning the BERT model for the detection of hate speech and offensive language. Using transformer-based embeddings and training on a balanced dataset, our model achieved a weighted F1-score and accuracy of 0.89.

These findings confirm the potential of transformer-based models for automated content moderation, providing an effective tool to improve user safety and platform integrity. The high F1-score underscores BERT's ability to handle nuanced linguistic patterns in social media data.

To advance this domain, future research will focus on exploring hybrid models combining BERT with CNN or LSTM to better capture sarcasm and code-mixed text. This will improve detection in complex linguistic scenarios. Furthermore, using multilingual datasets and domain-specific pre-training can enhance cross-platform generalizations of text data to avoid biases of models and support deployment across diverse social media [2], [3].

References

[1] A. Schmidt and M. Wiegand, "A survey on hate speech detection using natural language processing," in *Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media*, 2017, pp. 1–10. [Online]. Available: https://aclanthology.org/W17-1101/

[2] T. Davidson, D. Warmsley, M. Macy, and I. Weber, "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, 2017, pp. 512–515. [Online]. Available: https://ojs.aaai.org/index.php/ICWSM/article/view/14955

[3] B. Mathew, P. Saha, H. Tharad, S. Rajput, S. Singhania, M. B. Zafar, and others, "HateXplain: A benchmark dataset for explainable hate speech detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 17, 2020, pp. 14867–14875. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/17817

[4] P. Fortuna and S. Nunes, "A survey on automatic detection of hate speech in text," *ACM Computing Surveys*, vol. 51, no. 4, pp. 1–30, 2018. [Online]. Available: https://dl.acm.org/doi/10.1145/3232676

[5] B. Vidgen, A. Harris, D. Nguyen, R. Tromble, S. Hale, and H. Margetts, "Challenges and frontiers in abusive content detection," in *Proceedings of the Third Workshop on Abusive Language Online*, 2019, pp. 80–93. [Online]. Available: https://aclanthology.org/W19-3509/

[6] T. Gillespie, "Content moderation, AI, and the question of scale," *Big Data & Society*, vol. 7, no. 2, 2020. [Online]. Available: https://journals.sagepub.com/doi/10.1177/2053951720943234

[7] I. Kwok and Y. Wang, "Locate the hate: Detecting tweets against blacks," in *Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence*, 2013, pp. 1621–1622. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/8499

[8] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018. [Online]. Available: https://arxiv.org/abs/1810.04805

[9] A. Vaswani, N. Shazeer, N. Parmar, J. Uszoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, vol. 30, 2017. [Online]. Available: https://papers.nips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

[10] M. Malik and others, "Comparative analysis of deep learning models for hate speech detection," *Journal of Computational Social Science*, vol. 5, pp. 123–135, 2022. [Online]. Available: https://arxiv.org/abs/2202.09517

[11] P. Wei and others, "A comparative study of deep learning models for hate speech detection," *IEEE Transactions on Computational Social Systems*, vol. 8, no. 3, pp. 726–735, 2021. [Online]. Available: https://arxiv.org/abs/2108.03305

[12] S. Albladi and others, "Hate speech detection using large language models: A comprehensive review," *IEEE Access*, vol. 43, no. 2, pp. 1–25, 2025. [Online]. Available: https://ieeexplore.ieee.org/document/10848067

[13] K. Roy and others, "Multilingual transformers for hate speech detection," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 2021, pp. 3456–3465. [Online]. Available: https://arxiv.org/abs/2101.03207

[14] J. Lu and others, "Dual contrastive learning for hate speech detection," *arXiv preprint arXiv:2307.05578*, 2023. [Online]. Available: https://arxiv.org/abs/2307.05578

[15] K. Mnassri, M. S. Akhtar, and T. Chakraborty, "Emotion-aware multi-task learning for conversations," *arXiv preprint arXiv:2302.08777*, 2023. [Online]. Available: https://arxiv.org/abs/2302.08777

[16] D. Antypas and J. Camacho-Collados, "On the robustness of hate speech detection," *arXiv preprint arXiv:2307.01680*, 2023. [Online]. Available: https://arxiv.org/abs/2307.01680

[17] K. Kedia and A. Nandy, "Offensive language detection in Dravidian languages," in *Proceedings of the Fourth Workshop on DravidianLangTech*, 2024. [Online]. Available: https://aclanthology.org/2024.dravidianlangtech-1.14/

[18] A. Alhothali and K. Moria, "Detecting white supremacist hate speech using domain specific word embedding with deep learning and BERT," *IEEE Access*, vol. 9, pp. 106363–106374, 2022. [Online]. Available: https://ieeexplore.ieee.org/document/9850543

[19] B. Guragain and others, "Hate speech detection in Devanagari-scripted languages," in *Proceedings of the CHiPSAL@COLING 2025*, 2024. [Online]. Available: https://aclanthology.org/2025.chipsal-1.37/