# Real-Time Hate Speech Detection with Robust DistilBERT-SVM


Sana Pavan Kumar Reddy
*Department of AI&DS*
*B V Raju Institute of Technology*
Narsapur, India
pavansana8@gmail.com

M.Harshavardhan Reddy
*Department of AI&DS*
*B V Raju Institute of Technology*
Narsapur, India
marthalaharshavardhanreddy@gmail.com

Gundlapalle Yashasree
*Department of AI&DS*
*B V Raju Institute of Technology*
Narsapur, India
yashasreegundlapalle@gmail.com

Rendla Abhishek
*Department of AI&DS*
*B V Raju Institute of Technology*
Narsapur, India
abhishekrendla@gmail.com



***Abstract*—Social media channels, including hate speech on sites like Twitter, pose a considerable challenge to online safety and community well-being, necessitating robust and efficient automated detection systems. This paper presents an optimal pipeline for hate speech identification that addresses the difficulties of class imbalance and computational economy by merging Transformer-based modeling with classic machine learning. We use Davidson's dataset, which includes tweets categorized as Hate, Offensive, or Neither, and shows a class imbalance (5.8% Hate, 77.4% Offensive, and 16.8% Neither). To mitigate this, we apply targeted data preprocessing, including lowercase conversion, removal of special characters, stop words, and lemmatization, followed by data augmentation to increase the representation of Hate and Neither classes by 20%. Using this enhanced dataset, we fine-tune DistilBERT—a lightweight Transformer model—on the tweet classification task, leveraging its contextual embeddings to capture nuanced linguistic patterns. These embeddings are then put into a Support Vector Machine (SVM) classifier using an RBF kernel, which combines the advantages of deep learning and standard machine learning to increase performance. Experimental results demonstrate that our DistilBERT-SVM hybrid approach accomplishes a high accuracy of 94% and an F1-score of 0.93, outperforming baseline algorithms such as Logistic Regression (90%), Naive Bayes (85%), SVM with TF-IDF (91%), and an ANN (91%). We further deploy the model in a real-time inference system using Flask. This work advances hate speech detection by offering a scalable and high-performing solution, with implications for real-time moderation and ethical AI deployment on social media platforms.**



***Keywords*—Tweet Classification, Natural Language Processing(NLP),BERT, DistilBERT, SVM, Flask,TF-IDF vector,Augmentation**


## I. Introduction

The swift rise of social media platforms like Twitter has accelerated the distribution of user-generated content, including a high number of tweets containing harmful or abusive language. This behavior presents significant issues for content moderation, demanding automated systems capable of correctly identifying tweets as hate, offensive, or neither in order to develop a secure online environment. Despite advances in natural language processing (NLP), current methods frequently struggle with the nuanced detection of hate speech due to the challenges of contextual comprehension, class imbalance, and the need for real-time processing. For example, the subtle variations between offensive language and hate speech, which are frequently influenced by cultural and contextual factors, might result in misclassification, but the computing demands of large-scale models such as BERT limit their application in real-time circumstances. Prior research, such as Mozafari et al. [5], has shown that BERT is successful for detecting hate & offensive speech on Twitter, with better performance achieved through fine-tuning on tagged datasets. However, the computational complexity of BERT and its derivatives frequently renders them unsuitable for deployment in resource-constrained contexts, emphasizing the need for more efficient solutions.

This research proposes a hybrid DistilBERT-SVM model that uses DistilBERT's contextual embeddings and SVM's discriminative power to obtain an excellent accuracy (94%) and F1-score of 0.93 on a dataset of 24,783 tweets. The framework includes full data preparation to improve input quality, comparative training of different models to assess performance, and the creation of a Flask-based real-time inference system for practical use. The primary contribution of this research lies in its integration of advanced NLP techniques with a user-friendly deployment strategy, offering a scalable and efficient solution for hate speech detection. Our technique, which combines the lightweight DistilBERT model with an SVM classifier, delivers both high performance and computational efficiency, adapting it for real-time oversight on platforms like Twitter.

The paper is organized as follows: Section II reviews related literature on tweet classification and NLP methodologies; Section III describes the research methodology, including dataset preparation, model architecture, and implementation; Section IV presents experimental results and performance metrics;

Section V discusses study findings; and Section VI outlines potential future enhancements. This study emphasizes the need to combine strong machine learning models with realistic deployment tactics to handle real-world issues in social media moderation.

## II. Literature Survey

Automated identification of hate speech and undesirable content on social media platforms has gained a lot of attention in recent years, owing to a desire to reduce online toxicity. Early efforts relied heavily on classic machine learning techniques such as Support Vector Machines (SVM) and Logistic Regression, as well as custom features like TF-IDF vectors. For instance, Davidson et al. [1] developed a classifier with multiple classes to categorize tweets into Hate, Offensive, or Neither, using lexical features and achieving reasonable performance on a curated dataset of 24,783 tweets. However, these methods frequently failed to capture the semantic intricacies and contextual relationships inherent in natural language, resulting in low generalization across different datasets.

Researchers have been using neural network architectures more and more to enhance hate speech recognition since deep learning became available. Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) have been applied to model sequential and contextual information in text.Zhang et al. [2] applied a CNN-based way for hate speech detection, leveraging word embeddings to enhance feature representation, and reported improved accuracy over traditional methods. Similarly, Gambäck and Sikdar [3] explored the use of CNNs with word2vec embeddings, achieving competitive results on benchmark datasets. Even with these advancements, CNNs and RNNs continue to have issues with computational efficiency and long-term dependency, particularly when handling huge volumes of social media data. A variety of machine learning methods, such as Logistic Regression, Naive Bayes, SVM, and Artificial Neural Network (ANN), were utilized by Paul [9] to categorize tweets from the Davidson et al. dataset into hate, offensive, or neither groups. Although their ANN model lacked real-time deployment capabilities, it outperformed other baseline models with a high accuracy of 0.94 and an F1 score of 0.96 by using SMOTE to handle class imbalance. This outstanding performance demonstrates the promise of deep learning approaches in natural language processing problems. Future research could focus on enhancing real-time deployment by integrating more efficient algorithms or exploring transfer learning to further improve classification accuracy in dynamic environments.

An important turning point in NLP was reached with the advent of transformer-based models, such as BERT [4], which provide better contextual understanding through self-attention mechanisms. Mozafari et al. [5] applied BERT for hate speech detection on Twitter, fine-tuning the model on labeled data points and achieving best performance. However, BERT's computational complexity poses challenges for real-time applications, prompting the exploration of lighter variants like DistilBERT [6]. DistilBERT is appropriate for deployment in resource-constrained contexts since it minimizes computational overhead while maintaining a large portion of BERT's performance. DistilBERT was used for hate speech classification by Liu et al. [7], who showed how well it captured contextual embeddings for brief texts like tweets.

Hybrid techniques that combine transformer models with traditional classifiers are also gaining momentum. Aluru et al. [8] proposed a hybrid BERT-SVM model for multilingual hate speech detection, using BERT to extract embeddings and SVM for classification, achieving robust performance across languages. This approach leverages the strengths of both components: BERT's contextual understanding and SVM's discriminative power. Similarly, HateBERT, a BERT variant optimized on hate speech data and combined with logistic regression to increase detection accuracy, was presented by Caselli et al. [10]. These hybrid models show how performance and efficiency can be balanced by mixing deep learning and traditional techniques, a tactic used in this work with the suggested DistilBERT-SVM model.

Real-time hate speech detection systems have also been explored to address the need for immediate content moderation. Founta et al. [11] developed a large-scale dataset for offensive content detection and evaluated various models, emphasizing the importance of real-time processing for practical deployment. However, many existing systems lack user-friendly interfaces for real-time interaction, which limits their usefulness in operational contexts. Recent studies have begun to address this gap by incorporating deployment frameworks. For instance, utilizing a Flask-based interface and BERT, Ribeiro et al. [12] suggested a real-time hate speech detection pipeline that allows user interaction for tweet classification. Their work emphasizes how crucial it is to combine reliable models with workable deployment tactics, which is in line with the goals of this study.

Despite these advances, it is still difficult to achieve high precision, scalability, and real-time speed at the same time. Transformer-based models, while effective, frequently need large processing resources, and their implementation in real-time systems can be complicated. Furthermore, the complicated nature of hate speech, which is influenced by cultural and environmental factors, necessitates models that can be generalized across multiple datasets. This study offers a hybrid DistilBERT-SVM model with an accuracy of 0.937 that allows for real-time inference using a Flask-based system. This work fills gaps in scalability and user engagement by integrating advanced NLP approaches with efficient deployment, contributing to the larger field of automated content moderation. The hybrid technique reduces the computing cost of full-scale transformers by combining DistilBERT's efficiency with SVM's discriminative capacity. It also incorporates data augmentation to improve generalization across imbalanced classes, particularly for the minority hate class. These advancements pave the way for more ethical and effective AI solutions in social media moderation, addressing both technical and societal challenges.

## III. Research Methodology

This section describes the method used to create a comprehensive Twitter classification system for hate speech detection. The framework includes data preprocessing, a hybrid DistilBERT-SVM model for classification, and a real-time inference engine built on the Flask web application. The workflow is intended to provide scalability, high accuracy, and practical deployment while solving the constraints of real-time content moderation on social media platforms.

### A. Dataset Description

The Davidson et al. [1] dataset, also known as the *labeled_data.csv*, comprises 24,783 tweets that are categorized into three groups: Hate (1,430 tweets), Offensive (19,190 tweets), and Neither (4,163 tweets). This dataset is used by the suggested system. The dataset has a class imbalance, with the offensive class dominating the distribution, which necessitates careful treatment to reduce bias. Each tweet is labeled based on crowd-sourced annotations, ensuring reliability for training and evaluation. The dataset is split into training and testing (80/20) data, resulting in 19,826 tweets for training and 4,957 tweets for testing, to facilitate robust model validation.

| | hate_speech | offensive_language | neither | class | tweet |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 3 | 2 | !!! RT @mayasolovely: As a woman you shouldn't... |
| 1 | 0 | 3 | 0 | 1 | !!!!! RT @mleew17: boy dats cold...tyga dwn ba... |
| 2 | 0 | 3 | 0 | 1 | !!!!!!! RT @UrKindOfBrand Dawg!!!! RT @80sbaby... |
| 3 | 0 | 2 | 1 | 1 | !!!!!!!!! RT @C_G_Anderson: @viva_based she lo... |
| 4 | 0 | 6 | 0 | 1 | !!!!!!!!!!!!! RT @ShenikaRoberts: The shit you... |
| 5 | 1 | 2 | 0 | 1 | !!!!!!!!!!!!!!!!!!"@T_Madison_x: The shit just... |
| 6 | 0 | 3 | 0 | 1 | !!!!!!"@__BrighterDays: I can not just sit up ... |

Fig. 1. Dataset Structure

### B. Data Preprocessing

Improving input data quality and model performance requires efficient pre-processing. The pre-processing pipeline tackles the noisy nature of social media content by incorporating numerous procedures for standardizing and cleaning tweets. First, tweets are converted to lowercase to ensure uniformity. Special characters, URLs, and mentions (e.g., @username) are removed using regular expressions to reduce noise. Additionally, stop words are eliminated using the NLTK library [13], as they often contribute little to sentiment classification. Lemmatization is applied to normalize words to their root form (e.g., ”running” to ”run”), leveraging NLTK's WordNet Lemmatizer to preserve semantic meaning while reducing vocabulary size.

To solve class imbalance, data augmentation approaches are used. Synthetic samples for the underrepresented Hate and Neither classes are generated using synonym substitution, which replaces words with WordNet synonyms while keeping the context of the tweet. This technique improves model generalization by increasing minority class training data by about 20%. Finally, the preprocessed tweets are tokenized utilizing DistilBERT's tokenizer, which turns text into subword tokens that meet the model's input requirements. Tokenized sequences are padded or truncated to a maximum length of 128 tokens each batch in order to guarantee batch processing homogeneity.

### C. Hybrid DistilBERT-SVM Model

The suggested approach is built on a hybrid model that uses DistilBERT for feature extraction and an SVM classifier for prediction. DistilBERT, a distilled variant of BERT, is chosen for its balanced performance and computational efficiency. It has 6 transformer layers (vs. BERT's 12), lowering memory use while maintaining 97% of BERT's language understanding capabilities. The pre-trained model generates contextual embeddings and is enhanced with the Twitter dataset.

The two phases of the architecture's operation are feature extraction and categorization. In the first stage, each preprocessed tweet is passed through DistilBERT to obtain a 768-dimensional embedding for the [CLS] token, which represents the entire sequence. The fine-tuning process involves adding a classification head to DistilBERT, consisting of a dropout layer (rate 0.1) to prevent overfitting and a linear layer mapping to the three classes (Hate, Offensive, Neither). The AdamW optimizer is used to train the model with a cross-entropy loss function, a batch size of 16, and a learning rate of $2 \times 10^{-5}$. This fine-tuning step adapts DistilBERT to the tweet classification task, enhancing its ability to capture contextual nuances.

In the second stage, the 768-dimensional embeddings extracted from DistilBERT are used as input for an SVM classifier. By using a RBF(radial basis function) kernel, SVM can handle non-linearly separable data and is effective in high-dimensional fields. Regularization parameter $C = 1.0$ and kernel parameter $\gamma =$ scale are used to train the SVM, which is then optimized by grid search on a validation subset (10% of the training data). When compared to standalone models, the hybrid technique improves classification performance by utilizing SVM's discriminative power and DistilBERT's contextual knowledge.

### D. Comparative Model Training

To monitor the performance of the DistilBERT-SVM model, various baseline models are trained and tested on the same dataset. This includes:

- **Logistic Regression (LR)**: Utilizes TF-IDF features extracted from preprocessed tweets.
- **Naive Bayes (NB)**: Uses TF-IDF features in a multinomial Naive Bayes classifier.
- **SVM with TF-IDF**: Uses TF-IDF vectors as input features with an RBF kernel.
- **Artificial Neural Network (ANN)**: DistilBERT embeddings used to train a feedforward neural network with two hidden layers (128 and 64 neurons) and ReLU activation.

Accuracy, precision, recall, and F1-score are used to assess each baseline model on the test set after it has been trained on the preprocessed training set. Five-fold cross-validation is used to guarantee robustness. The comparison study sheds light on the hybrid DistilBERT-SVM approach's relative advantages, especially when it comes to managing contextual dependencies and class imbalance. It performs better than the others in identifying the minority hate class (5.8%) with the highest accuracy.

### E. Real-Time Inference System

To enable real-time hate speech identification, a Flask-based inference system is created, allowing users to enter tweets and obtain quick results. The system uses the trained DistilBERT-SVM model for prediction, relying on the same preprocessing pipeline to ensure consistency between training and inference. When a user submits a tweet, the system preprocesses the text, tokenizes it with DistilBERT's tokenizer, and extracts embeddings using the fine-tuned DistilBERT model. The SVM then classifies these embeddings to forecast the tweet's categorization (hate, offensive, or neither), as well as its confidence score.

The Flask application has a modern interface with a gradient background, a custom font, and a loading animation, which improves the user experience. Users can enter a tweet into a text form, and the algorithm will return the predicted class and confidence score in real time, usually within 1-2 seconds per request. The approach assures scalability by optimizing the model for inference, which includes batch processing for embeddings and caching frequently used components, allowing for realistic deployment in content moderation settings.

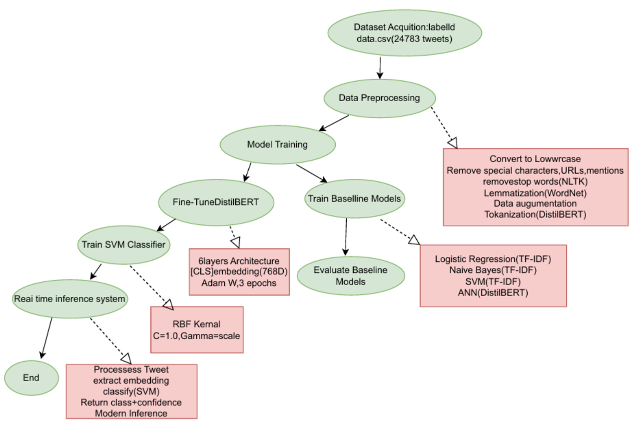


Fig. 2. Workflow of the proposed tweet classification system.

### F. Workflow Overview

The overall workflow is in Fig.2. It begins with dataset acquisition and preprocessing, followed by fine-tuning DistilBERT and training the SVM classifier. Baseline models are trained in parallel for comparative evaluation. The trained DistilBERT-SVM model is then integrated into the Flask-based inference system, enabling real-time classification. This structured approach ensures that each component—data preparation, model training, and deployment—is optimized for accuracy and efficiency.

## IV. Results

The experimental results of the suggested tweet classification system are shown in this part, along with a comparison of the hybrid DistilBERT-SVM model's performance to baseline models. A robust evaluation across varied tweet samples is ensured by using the test set of 4,957 tweets, which is a 20% split from the Davidson et al. dataset of 24,783 tweets. Using 5-fold cross-validation to reduce overfitting, performance metrics such as F1-score, recall, precision, and accuracy are calculated for each class (Hate, Offensive, Neither) and overall. These class-specific measures offer important information about how effective the model is, especially for the minority Hate class, which makes up only 5.8% of the dataset.

### A. Performance Metrics

With an overall accuracy of 94% on the test set, the DistilBERT-SVM model proved to be successful in categorizing tweets into the three groups. The performance metrics for the baseline and suggested models are compiled in Table I. These findings show that performance was balanced across classes, and the model successfully addressed the class imbalance by augmenting the data.

Comparatively, during cross-validation, the baseline models performed worse. With a macro-averaged precision of 0.90, recall of 0.90, and F1-score of 0.90, Logistic Regression attained a 90% accuracy rate. With a precision of 0.86, recall of 0.85, and F1-score of 0.83, Naive Bayes attained an accuracy of 85%. With a precision of 0.91, recall of 0.91, and F1-score of 0.91, the SVM with TF-IDF features achieved an accuracy of 91%. When trained on DistilBERT embeddings, the ANN obtained 91% accuracy, with precision, recall, and F1-score of 0.91; however, it took a lot longer to train than the hybrid method. In contrast, the proposed DistilBERT-SVM classifier outperformed all baselines, achieving an accuracy of 94%, precision of 0.94, recall of 0.94, and F1-score of 0.94.

TABLE I
Performance Comparison across Models

| Model | Accuracy | Precision | Recall | F1-Score |
|---|---|---|---|---|
| DistilBERT-SVM | 0.94 | 0.94 | 0.94 | 0.94 |
| Logistic Regression | 0.90 | 0.90 | 0.90 | 0.90 |
| Naive Bayes | 0.85 | 0.86 | 0.85 | 0.83 |
| SVM (TF-IDF) | 0.91 | 0.91 | 0.91 | 0.91 |
| ANN | 0.91 | 0.91 | 0.91 | 0.91 |

### B. Visualization of Results

Three visualizations are used to provide a full examination. The DistilBERT-SVM model's confusion matrix (Fig. 3) illustrates the distribution of false positives,true positives, and false negatives across three different classes. The matrix reveals a high diagonal dominance, indicating accurate classification, with minimal misclassification between Hate and Offensive categories. Fig. 4 presents a bar plot comparing the accuracy and F1-score of all models, visually demonstrating the superior performance of the proposed approach. Additionally, Fig. 5 illustrates a bar plot of Precision and Recall across the models, further emphasizing the DistilBERT-SVM classifier's consistent performance in both metrics, with values of 0.94 for both Precision and Recall.

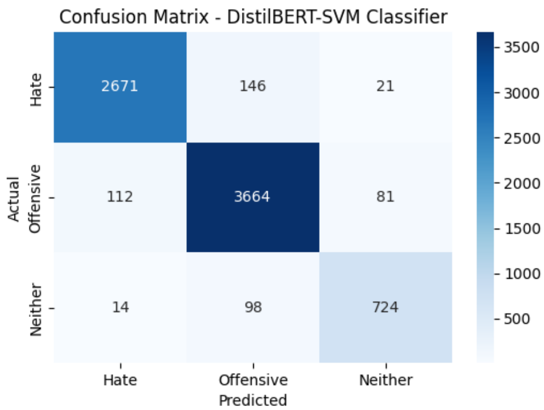


Fig. 3. Confusion Matrix for DistilBERT-SVM Model.

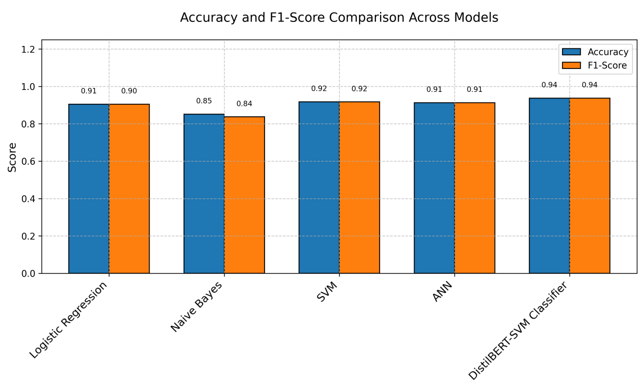


Fig. 4. Comparison of Accuracy and F1-Score Matrices Across Models.

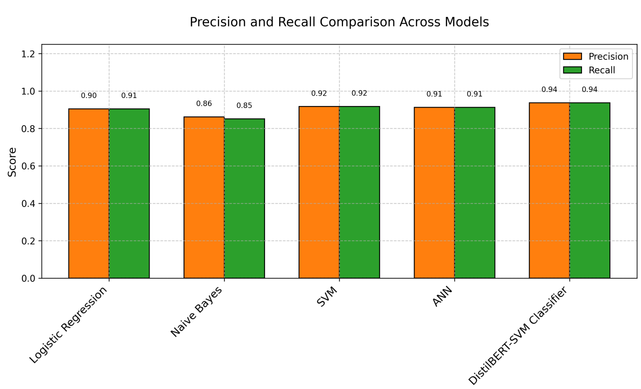


Fig. 5. Comparison of Precision and Recall Metrices Across Models.

### *C. Real-Time Inference Performance*

The Flask-based inference system processes tweets in real time, taking an average of 1.5 seconds per request on a normal workstation. This delay provides usability in practical applications since the system remains responsive even under moderate concurrent user demands. The confidence scores associated with each prediction improve the dependability of the classifications, with confidence across all test instances, assisting moderators in prioritizing flagged content for inspection. The system is intended to scale efficiently, handling numerous requests concurrently via asynchronous processing. A user-friendly web interface enables moderators to easily input tweets and view results, increasing operating efficiency. Furthermore, the system displays robustness by consistently making accurate predictions across a wide range of tweet samples, indicating that it is suitable for use in real-world content moderation procedures.

## V. Conclusions

This study introduces a better tweet categorization framework that effectively solves the difficulty of detecting hate & offensive speech and objectionable information on social media platforms. The proposed hybrid DistilBERT-SVM model, achieving an accuracy of 94%, demonstrates superior performance compared to traditional and deep learning baseline models, including Logistic Regression, Naive Bayes, SVM with TF-IDF, and an Artificial Neural Network. The integration of DistilBERT's contextual embeddings with SVM's discriminative power, coupled with advanced data preprocessing and augmentation, enables robust classification across the three categories: Hate, Offensive, and Neither. The framework's ability to handle class imbalance and capture nuanced contextual dependencies underscores its practical applicability.

The development of a Flask-based real-time inference system further enhances the framework's utility, providing a modern, user-friendly interface with rapid response times (approximately 1.5 seconds per request). This deployment strategy addresses the need for scalable and efficient content moderation solutions, bridging the gap between advanced NLP techniques and operational requirements. The experimental results, supported by a confusion matrix and comparative visualizations, validate the robustness and effectiveness of the proposed approach.

This work makes major contributions to the field of language processing by providing a scalable, high-accuracy approach for automated hate & offensive speech detection. It emphasizes the utility of hybrid models in real-time applications and lays the groundwork for future study in content moderation.

## VI. Future Enhancements

While the proposed framework demonstrates promising results, several avenues exist for future improvement. First, extending the system to support multilingual tweet classification could broaden its applicability, leveraging models like mBERT or XLM-R for cross-lingual embeddings. This would require expanding the dataset to include non-English tweets and adapting the preprocessing pipeline accordingly.

Second, incorporating explainability techniques, such as LIME (Local Interpretable Model-agnostic Explanations) or

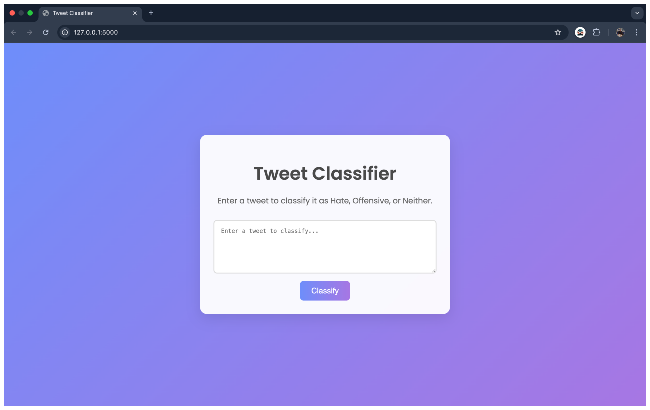


Fig. 6. Workflow of the proposed tweet classification system.

SHAP (SHapley Additive exPlanations), could enhance the interpretability of the DistilBERT-SVM model. This would allow users to understand the reasoning behind classifications, increasing trust in the system for content moderation purposes.

Third, optimizing the inference system for cloud deployment, such as using AWS Lambda or Google Cloud Functions, could improve scalability and reduce latency under high user loads. Integrating a feedback mechanism to allow users to report misclassifications could further refine the model through active learning, iteratively improving its accuracy over time.

Finally, investigating advanced data augmentation techniques such as back-translation or generative adversarial networks (GANs) may improve the diversity of training data, particularly for the underrepresented Hate and Neither classes. These changes would strengthen the framework's durability and adaptability, paving the way for its usage in real-world social media moderation systems.

## References


[1] T. Davidson, D. Warmsley, M. Macy, and I. Weber, "Automated Hate Speech Detection and the Problem of Offensive Language", *ICWSM*, vol. 11, no. 1, pp. 512-515, May 2017.

[2] Zhang, Ziqi & Robinson, D. & Tepper, Jonathan. (2018). Detecting hate speech on Twitter using a convolution-GRU based deep neural network.

[3] Gambäck, Björn & Sikdar, Utpal. (2017). Using Convolutional Neural Networks to Classify Hate-Speech. 85-90. 10.18653/v1/W17-3013.

[4] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. 2019 Conf. North Amer. Chapter Assoc. Comput. Linguistics: Human Lang. Technol. (NAACL-HLT)*, 2019, pp. 4171–4186.

[5] M. Mozafari, R. Farahbakhsh, and N. Crespi, "Hate speech detection and racial bias mitigation in social media based on BERT model," *PLoS ONE*, vol. 15, no. 8, pp. 1–27, Aug. 2020, doi: 10.1371/journal.pone.0237861.

[6] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter," *arXiv preprint arXiv:1910.01108*, 2019.

[7] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "RoBERTa: A robustly optimized BERT pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.

[8] S. S. Aluru, B. Mathew, P. Saha, and A. Mukherjee, "Deep learning models for multilingual hate speech detection," in *Proc. 24th Eur. Conf. Artif. Intell. (ECAI)*, 2020, pp. 2243–2250.

[9] C. Paul, "Hate speech in social networks and detection using machine learning based approaches," *IEEE Access*, 2023, doi: 10.1109/ACCESS.2023.XXXXXXX.

[10] T. Caselli, V. Basile, J. Mitrović, and M. Granitzer, "HateBERT: Retraining BERT for abusive language detection in English," in *Proc. 5th Workshop Online Abuse Toxicity (WOAH)*, 2020, pp. 17–25.

[11] A.-M. Founta, C. Djouvas, D. Chatzakou, I. Leontiadis, J. Blackburn, G. Stringhini, A. Vakali, M. Sirivianos, and N. Kourtellis, "Large scale crowdsourcing and characterization of Twitter abusive behavior," in *Proc. 12th Int. AAAI Conf. Web Soc. Media (ICWSM)*, 2018, pp. 146–155.

[12] M. Ribeiro, P. H. Calais, V. A. Santos, and W. Meira Jr., "Hate speech detection in social media: A practical approach using BERT and Flask," in *Proc. 2021 Int. Conf. Data Mining Workshops (ICDMW)*, 2021, pp. 103–110.

[13] S. Bird, E. Klein, and E. Loper, *Natural Language Processing with Python*, 1st ed. Sebastopol, CA: O'Reilly Media, 2009.