MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1s, 2026, Pages: 3286-3292

Robust Detection of Code-Mixed Hinglish Hate Speech Using Weighted Transformer Ensembles

Samarth Singh[a], Kushagra Sahay[a], Dr. A. Pandian[a,1]
[a]*Department of Computing Technologies, SRM Institute of Science and Technology, Chennai, Tamil Nadu, India, 603023*

Abstract
Social media is simply the primary location where everybody communicates with one another. However, as it became more frequently used, hate and abusive content increased in popularity. Hate messages are quite difficult to automatically identify, in particular when individuals combine several languages in one sentence. Most posts in India are in *Hinglish* - it is Hindi + English smushed together with odd phonetic spellings, informal grammar, and inconsistent writing style. Therefore, we considered how we can automatically identify hate speech on Hindi and Hinglish posts. A unified dataset containing more than 93,000 labeled samples was constructed by combining multiple publicly available datasets. Several modeling approaches were evaluated, including traditional machine learning methods, deep learning architectures, and multilingual transformer models.Among the evaluated models, transformer-based approaches demonstrated the strongest performance. We fine-tuned Multilingual BERT (mBERT), MuRIL, and XLM-RoBERTa to a yes/no hate speech label. To further improve performance, we then combined MuRIL and XLM-RoBerta in a weighted ensemble and that produced the best F1-score of 0.7701.The findings highlight the importance of regionally adapted transformer models and ensemble strategies when handling code-mixed multilingual text. Our framework provides a specific method of creating moderation tools capable of identifying the harmful posts in Hindi and Hinglish.

Keywords: Hate Speech Detection, Hinglish NLP, Code-Mixed Text, Transformer Models, MuRIL, XLM-RoBERTa, Ensemble Learning

Introduction

The internet social media has completely transformed the way we interact with each other, exchange views and engage in the general debate in cyberspace. On platforms like Twitter, Facebook, and Instagram, a single post can reach thousands of people right away. As much as these platforms facilitate open communication, they have also become avenues through which abusive and offensive materials can go viral within a short period. Therefore the issue of hate speech, harassment, and targeted abuse are significant issues of online communities and the moderators who attempt to maintain a civil situation [1].Automatic detection of such harmful content has become an important research problem in natural language processing and machine learning [1]. Reliable detection systems have also been shown to be useful to moderation teams as a way to identify potentially harmful posts at scale [2]. It implies that it requires less manpower among reviewers who spend their time wading through extensive user-generated data otherwise.Most existing hate speech detection systems have been developed primarily for English language text. However, in multilingual societies such as India, social media users frequently communicate using regional languages or mixtures of languages within a single sentence. One of the most common forms of such linguistic blending is “Hinglish,” a codemixed form of Hindi and English in which Hindi words are often written using the Latin alphabet [3]. This form of communication introduces significant challenges for conventional NLP systems due to inconsistent spellings, phonetic variations, and the presence of multiple scripts.The detection of hate speech in Hinglish text is particularly difficult because abusive expressions may appear in numerous orthographic forms. For example, users often modify words through character repetition, punctuation insertion, or phonetic spelling variations to avoid moderation systems. Traditional machine learning models based on word-level features struggle to capture these variations [1], while sequential deep learning models may fail to fully understand long-range contextual dependencies present in social media posts. Recent progress in large pre-trained language models has significantly improved text classification performance [4]. Transformer architectures can provide contextual relations between words and are able to process multilingual text more effectively than earlier designs. Models such as multilingual BERT (mBERT) [5], MuRIL [6], and XLM-RoBERTa [7] learn representations across multiple languages and scripts and thus are highly helpful in the analysis of code-mixed Hindi-English texts that frequently appear on social media.The project consists of the construction of an automated Hindi and Hinglish social media text hate-speech detection system. To support this goal, we constructed a unified dataset of over 93,000 labelled samples by combining several publicly available datasets. The dataset offers us an opportunity to test various models and offers a richer sample of code-mixed language that is trending on the Internet.We worked with multiple model architectures in this study. Traditional machine learning models are commonly used as baseline methods in text classification tasks [8], which are followed by deep learning architectures that are able to identify sequence patterns in text. Lastly, we fine tune multilingual transformer models for binary hate speech classification. This allows us to systematically compare different approaches for handling code-mixed Hindi-English data.In order to improve classification further, we added ensemble strategy combining the predictions of two most successful transformer models - MuRIL and XLM-RoBERTa. By aggregating their probability outputs using a weighted voting mechanism, the system aims to leverage the complementary capabilities of both models. The findings indicate that the ensemble methods based on transformers have the potential to enhance the accuracy of the detection in multilingual and mixed-language settings. Suffice to say, this work contributes to advancing the creation of automated systems of moderation, which can operate successfully on regional and multilingual social media material.

Objectives of the Study: The main goal of this research is to design and evaluate a system that can automatically detect hate speech in Hindi and Hinglish social media posts [1, 2]. Social media platforms contain a large amount of user-generated content, and manual moderation alone is often insufficient to handle harmful or abusive language.

Code-mixed language such as Hinglish introduces several challenges for natural language processing systems [3]. Words may appear in different phonetic spellings, grammatical structures are often irregular, and Hindi and English words frequently appear within the same sentence. Because of these variations, models trained only on standard English text often perform poorly on such data.

To address these challenges, this study evaluates multiple approaches for hate speech classification, including traditional machine learning methods, deep learning architectures, and transformer-based language models [4, 5]. The objectives of this study are as follows:

- Construct a dataset for Hindi–English code-mixed hate speech detection by combining several publicly available datasets and applying consistent preprocessing techniques.
- Establish baseline performance using traditional machine learning models such as Logistic Regression and Support Vector Machines with TF–IDF features.
- Investigate deep learning architectures such as CNN and LSTM models to examine their ability to capture contextual patterns in informal social media text.
- Fine-tune multilingual transformer models including Multilingual BERT (mBERT), MuRIL, and XLM-RoBERTa for binary hate speech classification.

[1] **Corresponding author:** *Email address:* pandiana@srmist.edu.in (Dr. A. Pandian)

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1s, 2026, Pages: 3286-3292

- Develop a weighted ensemble approach that combines the predictions of the bestperforming transformer models.
- Evaluate model performance using standard metrics including Precision, Recall, F1-score, and Accuracy, and analyze common patterns in misclassified samples.

These objectives guide the design and evaluation of the proposed hate speech detection framework.

Scope and Significance

Social media platforms have become a major space for public communication. People use these platforms to share opinions, discuss social issues, and interact with others online. However, the same platforms can also enable the rapid spread of harmful or abusive content.

Because of the scale of social media activity, manual moderation alone is not sufficient. Automated detection systems are therefore important for identifying potentially harmful content and supporting moderation efforts [1, 2].

This study focuses specifically on hate speech written in Hindi and Hinglish. Hinglish is a code-mixed form of Hindi and English that is widely used across Indian social media platforms. In many cases, Hindi words are written using the Latin alphabet and mixed with English words within the same sentence.This style of communication introduces several challenges for natural language processing systems. Spellings may vary significantly, grammar can be inconsistent, and linguistic structures often combine elements from both languages. As a result, models trained primarily on standard English text often struggle to process such data effectively[3].To address these challenges, this study constructs a dataset containing more than 93,000 labeled samples collected from multiple publicly available sources. Several modeling approaches are evaluated, including traditional machine learning models, deep learning architectures, and multilingual transformer models.The proposed system uses a weighted ensemble strategy that combines the predictions of MuRIL and XLM-RoBERTa [6, 7]. MuRIL provides strong representations for Indian languages, while XLM-RoBERTa contributes broader multilingual contextual understanding. Combining these models helps improve overall hate speech detection performance.The findings of this study are relevant from both technical and practical perspectives. Social media platforms require reliable tools that can identify harmful content in multilingual environments. An effective hate speech detection system can assist moderation teams by automatically flagging potentially abusive posts.Improved detection systems may help reduce the spread of harmful content online and support healthier digital discussions. In multilingual societies such as India, systems that can understand code-mixed language are particularly important for maintaining safe and inclusive online spaces.

Literature Review

The rapid growth of social media platforms has led to a significant increase in online discussions, but it has also amplified the spread of abusive and hateful content. Detecting such content automatically has therefore become an important task in natural language processing [1]. Early work on hate speech detection largely focused on English-language datasets and relied on traditional machine learning techniques. Davidson et al. [9] explored classifiers such as Logistic Regression, Naive Bayes, and Support Vector Machines using TF–IDF features, demonstrating that these approaches can provide useful baseline performance. However, later studies pointed out that purely lexical approaches often fail when hateful language is expressed implicitly or when the interpretation depends on context. These limitations of surface-level lexical approaches have been discussed in detail in the survey by Schmidt and Wiegand [1].

To overcome the constraints of feature-based models, subsequent research began adopting neural network architectures capable of learning contextual representations from text. Badjatiya et al. [10] showed that deep learning models such as Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks can capture semantic relationships between words more effectively than traditional models. These developments eventually led to the widespread use of transformer-based architectures. Bidirectional Encoder Representations from Transformers (BERT), introduced by Devlin et al. [5], demonstrated that large-scale pretraining with attention mechanisms enables models to learn bidirectional contextual representations and significantly improves performance across many NLP tasks.As the field progressed beyond English-language datasets, multilingual transformer models were developed to support cross-lingual applications. XLM-RoBERTa [7] showed strong performance across multiple languages by training on large multilingual corpora. For Indian languages, Khanuja et al. [6] proposed MuRIL, a multilingual transformer specifically designed for Indian linguistic settings. Since social media users frequently write Hindi words using the Latin alphabet, MuRIL is particularly effective for analyzing transliterated Hindi and code-mixed text.

In addition to model development, several studies have focused on creating datasets for Hindi and Hindi–English code-mixed hate speech. Bohra et al. [11] introduced an early annotated dataset for Hindi–English code-mixed tweets, enabling initial experiments on hate speech detection in Hinglish. The HASOC shared tasks [12] further expanded research in this area by providing benchmark datasets for hate speech and offensive language detection across several Indo-European languages, including Hindi.More recent work has also emphasized the importance of richer annotation schemes and multilingual learning strategies. Kapil et al. [13] proposed the Hindi Hate Speech Dataset (HHSD), a large annotated corpus built using a hierarchical multi-layer annotation framework. The dataset distinguishes between hateful and non-hateful content while also capturing finer-grained information such as explicit versus implicit hate, thematic categories, and targeted entities. Their experiments show that transformer-based models combined with multi-task learning across related languages such as Urdu and Bangla can significantly improve classification performance.Although these studies have advanced hate speech detection for Hindi and code-mixed text, several challenges remain. Social media content often includes phonetic spellings, informal abbreviations, and evolving slang that rarely appear in curated datasets. Even models trained on transliterated data may struggle with these variations. Furthermore, most previous work evaluates individual models independently, with relatively limited exploration of ensemble strategies that combine the strengths of different architectures. To address these limitations, the present study constructs a unified dataset and examines transformer-based models alongside ensemble techniques for hate speech detection in Hindi and Hinglish text.

Dataset Description

Hate speech detection systems rely heavily on the availability of high-quality labelled datasets. In order to obtain sufficient data coverage for Hindi and Hinglish text, this study constructs a unified dataset by integrating multiple publicly available sources. These datasets were originally collected from social media platforms such as X (formerly Twitter), Facebook, and Instagram, where users frequently communicate using informal and code-mixed language. Similar datasets for Hindi and code-mixed hate speech detection have been explored in previous studies [11, 12].

The dataset aggregation process combines three primary sources. The first source is the India Hate Speech Superset available on Hugging Face, which contains professionally annotated samples from social media platforms. The second source is the IndoHateSpeech dataset hosted on Mendeley Data, which focuses on code-mixed Hindi and English interactions collected from Instagram. The third source is the PRISM Hinglish dataset available on Kaggle, which provides a large number of labelled samples containing Romanized Hindi expressions.

Since these datasets were created independently, their labelling schemes and formats differed. A preprocessing pipeline was developed to standardize the data into a unified structure consisting of two primary fields: the text content and a binary label representing hate speech. All labels were converted into a consistent format where class 1 represents hate speech and class 0 represents non-hateful or normal text.

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1s, 2026, Pages: 3286-3292

Several cleaning steps were applied during the preprocessing stage. URLs and hyperlinks were removed to eliminate irrelevant content. User mentions were replaced with neutral tokens to prevent the model from learning user-specific patterns. Emoji characters and excessive punctuation were removed to reduce vocabulary noise, and duplicate entries were eliminated to avoid data leakage during training.

After completing the preprocessing and unification process, the final dataset contained more than 93,000 labelled samples. Approximately 30% of the dataset corresponds to hate speech, while the remaining 70% represents normal content. This class distribution reflects realistic social media environments, where harmful content typically appears less frequently than neutral conversations. Similar class distributions have also been reported in other hate speech datasets [13].

The unified dataset provides sufficient linguistic diversity for training modern machine learning models and allows comprehensive evaluation across multiple modelling approaches.

Methodology

The methodology used in this work consists of several stages designed to see how well we can detect hate speech in Hindi and Hinglish posts. We began with the traditional machine learning models to establish baseline performance. Deep learning architectures are then examined to capture the manner in which sequential patterns emerge in these informal posts. Finally, multilingual transformer models are fine-tuned and evaluated to determine how well contextual language representations improve classification performance [5, 7].

Data Preprocessing

Raw social-media text is noisy - links, the use of the @ character, strange spacing, etc, so I had to cleanse it before feeding it to any model. I reduced all letters to lowercase and removed URLs, symbols, and unnecessary whitespace. Their @handles and URLs were changed to neutral placeholders so that the models would not know anything specific to the platform.

Since Hinglish is everything about the casual phonetic combination of English and Hindi, I used the tokenizer to divide the text into subwords. This approach allows the models to capture subword patterns that frequently appear in code-mixed Hindi–English text.

Baseline ML Models

Traditional machine learning algorithms were first evaluated to provide baseline performance for the task. The cleaned text data was transformed into numerical feature vectors using Term Frequency–Inverse Document Frequency (TF–IDF) representations [8]. Both word-level and character-level n-grams were explored to capture spelling variations commonly found in Hinglish text.

Two commonly used classifiers were selected for evaluation: Logistic Regression and Linear Support Vector Machines (SVM). Model training was performed using stratified cross-validation so that the class distribution remained consistent across the training and validation splits. The performance of these models served as a reference point for comparison with more advanced deep learning and transformer-based approaches.

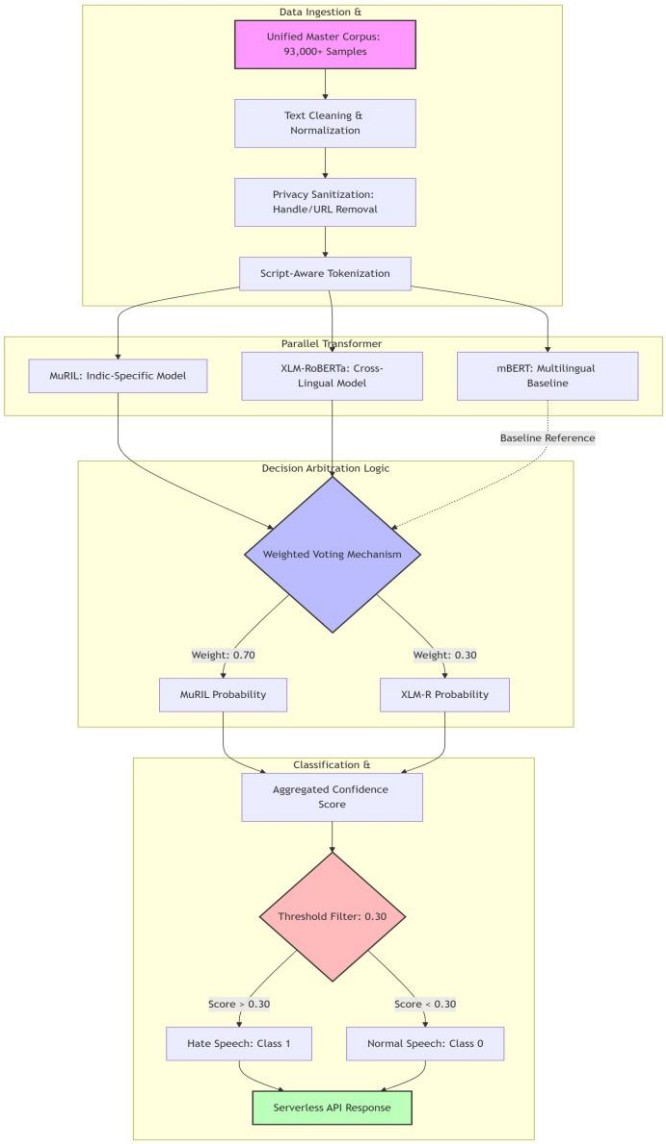

Figure 1: Proposed hate speech detection pipeline using transformer ensemble architecture.

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1s, 2026, Pages: 3286-3292

Deep Learning Architectures

In addition to feature-based models, neural network architectures were explored to capture sequential relationships within the text. A Bidirectional Long Short-Term Memory (BiLSTM) network was used to model contextual dependencies between words in a sentence [10]. Because the model processes the sequence in both forward and backward directions, it can incorporate information from both preceding and following words when generating representations.

The deep learning models utilized an embedding layer to convert tokens into dense vector representations. Dropout regularization was applied during training to reduce overfitting and improve generalization performance.

Transformer Models

Transformer architectures have become the dominant approach for many natural language processing tasks, particularly text classification problems [4]. Their self-attention mechanism allows the model to capture contextual relationships between words more effectively than traditional machine learning or sequential neural models.

In this study, three multilingual transformer models were selected for hate speech detection: Multilingual BERT (mBERT) [5], MuRIL [6], and XLM-RoBERTa [7].

These models were adapted to the classification task through transfer learning. Pretrained language representations were fine-tuned using the labeled dataset constructed for this study. During training, model parameters were optimized using cross-entropy loss.

Weighted Ensemble Strategy

To further improve classification performance, a weighted ensemble strategy was implemented by combining the probability outputs of the best-performing transformer models. Specifically, the predictions of MuRIL and XLM-RoBERTa were aggregated using a weighted probability combination:

$$P\text{ensemble} = w1\ P\text{MuRIL} + w2\ P\text{XLM-R}$$

where w_1 and w_2 represent the contribution weights of each model. A grid search procedure was performed to identify the optimal weight combination and classification threshold.

The final ensemble configuration assigned a higher weight to MuRIL due to its stronger performance on Hindi and transliterated text, while XLM-RoBERTa contributed complementary cross-lingual representations. This ensemble approach allows the system to leverage the strengths of both models and improve overall hate speech detection accuracy.

Experimental Setup

All experiments were implemented in Python using common machine learning and deep learning libraries. Traditional machine learning models were developed with the *Scikit-learn* library. Deep learning and transformer models were implemented using the *PyTorch* framework together with the *HuggingFace Transformers* library.

Training and evaluation were performed on a workstation equipped with an Intel Core i5 processor, 16GB RAM, and an NVIDIA RTX 3050 GPU with 6GB of VRAM. GPU acceleration was used during transformer fine-tuning and deep learning training in order to reduce training time.

The dataset was divided using stratified cross-validation so that the proportion of hate and non-hate samples remained consistent across training and validation folds. Model performance was measured using standard classification metrics including Precision, Recall, F1-score, and Accuracy.

For transformer models, fine-tuning was performed using the AdamW optimizer with a learning rate of 2×10^{-5} [5]. Training ran for several epochs, and early stopping was applied to reduce overfitting.

The main hyperparameters used during training are listed in Table 1.

Table 1: Key Training Hyperparameters

Parameter	Value
Maximum Sequence Length	128
Batch Size	8
Learning Rate	2*e*−5
Number of Epochs	5
Optimizer	AdamW
Cross Validation	3-Fold
Loss Function	Cross Entropy with Class Weights
Decision Threshold	0.30

Evaluation Metrics

The performance of the classification models was assessed using common evaluation metrics used in binary classification tasks [1]. These metrics include F1-score, Precision, Recall, and Accuracy. Together, they provide a balanced view of model performance by measuring prediction quality as well as the ability to correctly identify hate speech instances.

Precision reflects how many of the samples predicted as hate speech are actually correct:

$$Precision = \frac{TP}{TP + FP}$$

Recall measures the ability of the model to detect hateful content among all actual hate speech samples:

$$Recall = \frac{TP}{TP + FN}$$

The F1-score combines Precision and Recall into a single metric using their harmonic mean:

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

Accuracy indicates the overall proportion of correctly classified samples in the dataset:

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

These metrics were calculated on the validation folds of the unified Hindi–Hinglish dataset described earlier.

Results and Performance Evaluation

We tested the proposed dataset and modeling pipeline with the help of a variety of methods, including traditional machine learning methods, certain deep-learning models, and multilingual transformer models. F1-score was primarily used to measure the model's performance, which provided a balanced picture of precision and recall, which comes in handy when doing NLP tasks.

My team utilized the traditional machine-learning models as a starting point. Among these, the Linear Support Vector Machine using combined word and character TF–IDF features, in fact, provided us with the best baseline score, with an F1 -score of 0.7193.

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1s, 2026, Pages: 3286-3292

We further ventured into deep learning to determine whether sequential models could extract additional context from the text. The BiLSTM model that included an attention layer achieved an F1-score of 0.7259; a relatively small increase over the traditional baseline, but nonetheless promising.

The performance was picked up once again by transformer models. The multilingual BERT (mBERT) model [5] achieved an F1-score of 0.7514, while XLM-RoBERTa [7] reached 0.7603. MuRIL [6] was the best, giving an F1-score of 0.7620, which is likely due to the fact that it is already trained on Indian languages and text transliteration.

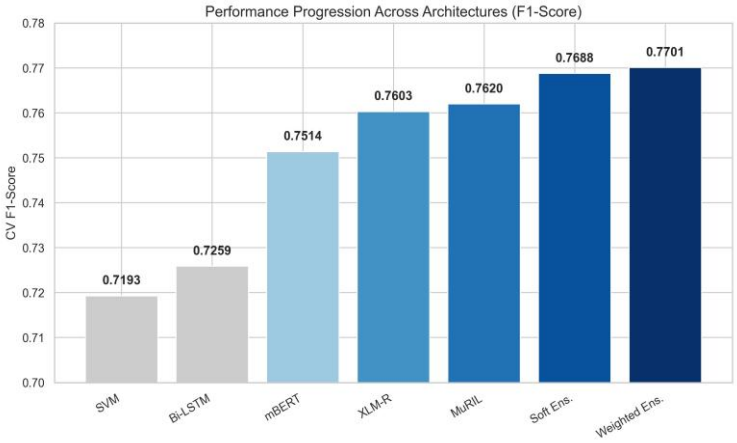

Figure 2: Performance progression across evaluated architectures based on cross-validation F1-score.

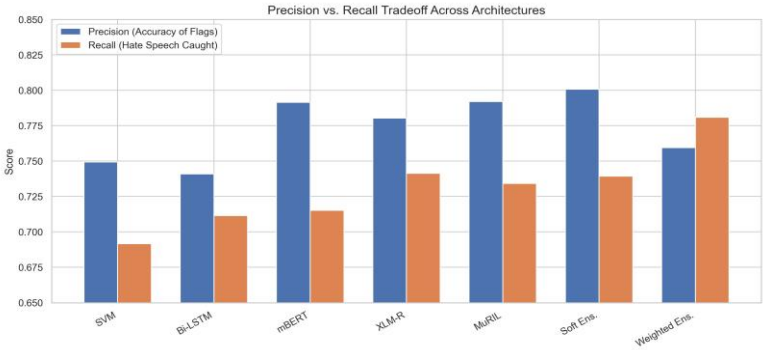

Figure 3: Precision–Recall comparison across evaluated model architectures.

To further improve classification performance, an ensemble approach was developed by combining the predictions of MuRIL and XLM-RoBERTa. Model weights and decision thresholds were tuned using grid search. The resulting weighted ensemble achieved the highest overall performance with an F1-score of 0.7701.

Table 2 presents a comparison of the performance of the evaluated models.

Table 2: Performance Comparison of Evaluated Models

Model	F1-score	Precision	Recall	Accuracy
Linear SVM	0.7193	0.7493	0.6916	0.8295
BiLSTM	0.7259	0.7409	0.7115	0.8361
mBERT	0.7514	0.7914	0.7152	0.8567
XLM-RoBERTa	0.7603	0.7803	0.7413	0.8589
MuRIL	0.7620	0.7920	0.7342	0.8688
Weighted Ensemble	**0.7701**	0.7595	0.7809	0.8638

Error Analysis: Although the proposed weighted ensemble model achieved the highest overall performance, a detailed error analysis was conducted to better understand the limitations of the system. The analysis focused on identifying similarities in the false positive and false negative predictions produced by the final ensemble model.

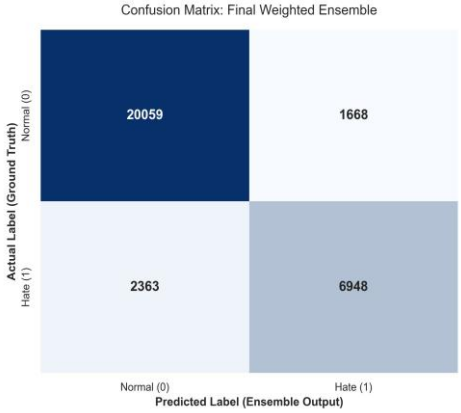

Figure 4: Confusion matrix of the final weighted ensemble model.

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1s, 2026, Pages: 3286-3292

A total of 4,031 misclassified samples were identified during validation. Among these errors, false negatives (instances of hate speech that were not detected by the system) constituted approximately 58.6% of the total errors, while false positives (non-hateful text incorrectly classified as hate speech) represented approximately 41.4%. Most of the hate speech cases that were overlooked were not explicit. The message can be passed across through sarcasm rather than abusive words, through indirect remarks or culturally specific terms. In these cases, the malicious intent is only seen once the context around the situation is taken into account and thus, it is hard to detect using automated systems. As an illustration, a comment like *"tum log toh bade deshbhakt ho, bas keyboard ke peeche se"* can seem to be neutral on the surface but have sarcastic or even insulting meaning depending on the context.

The other problem that was repeated was the altered spelling of offensive words. To avoid the use of moderation filters, users of social media often alter abusive words or phrases by introducing punctuation or repeating characters or changing spelling phonetic representations. Examples include variations such as *"ch***ya"*, *"chutiyaa"*, or *"chu.tiya"*. These variations can disrupt tokenization and reduce the effectiveness of transformerbased models [5]. The number of false positive predictions was seen in posts with strong or aggressive wordings but not necessarily hateful. Communication on politics, religion or social topics generally contains words or phrases which are emotionally charged even when the speaker is not actually abusing but merely criticizing. As an example, we can refer to sentences like *"yeh policy bilkul bekaar hai"* or *"yeh decision galat hai"* which contain negative statements but do not mean hate speech.

It was also found that there existed differences between the predictions made by MuRIL and XLM-RoBERTa. The two models differed in the output in about 28 percent of the incorrectly classified samples. The above observation implies that all models focus on varying linguistic signals, and this is one of the reasons why the combination of these models in an ensemble yields better performance.

Findings

The experimental results and analysis lead to several important findings regarding hate speech detection in Hindi and Hinglish online posts. First, Traditional ML models based on TF-IDF features provide a reasonable baseline, but they start struggling to capture linguistic variations when the text mixes languages. These models rely heavily on lexical patterns and therefore fail to recognize semantically similar expressions that appear in different phonetic forms.

Second, deep learning models like BiLSTM understand context way better than the older/traditional ML models. But they still struggle when the sentence gets long or when multiple languages are mixed together.

Third, transformer-based language models significantly outperform both traditional machine learning and sequential neural architectures [4]. Because transformers have a self-attention mechanism, they can understand the context between words much better. That's really helpful when trying to catch more nuanced forms of hate speech.

Fourth, region-specific transformer models such as MuRIL perform particularly well on Hindi and transliterated text [6]. This result highlights the importance of domainspecific pre-training when working with multilingual and culturally specific datasets.

Finally, when I combined the results of MuRIL and XLM-RoBERTa using a weighted ensemble, it produces the best overall performance. By exploiting the complementary nature of the two models, the result have improved recall value without precision dropping too much, resulting in the highest F1-score

Limitations

Although the proposed ensemble model has resulting promises, a number of constraints exist. The first of them is that the system cannot easily pick hate speech that is actually indirect. When the one speaks sarcasm, metaphors, and cultural jokes rather than the obvious abusive words, it tends to go under the model. This kind of cases tends to need more contextual knowledge or exogenous knowledge.

Secondly, the data in this research is mostly dedicated to the Hindi and Hinglish text gathered on the particular social media sites. Although the dataset is quite large, it is likely to be not representative of the range of linguistic ways of expression in different online communities.

Third, transformer models need large amounts of computational resources to be trained and inferred. Due to this fact they are not so viable in live moderation where speed and server budget are the key factors of consideration.

Lastly, the binary classification structure applied in the research makes the detection of hate speech simplified in two categories; hate and non-hate. Practically, there is also such a spectrum of online toxicity as offensive language, harassment, and abusive speech. Future studies can also consider the multi-class classification frameworks to represent these differences.

Future Scope

Even though the system does a decent job in spotting hate speech in Hindi and Hinglish, there are definitely a few angles we still need to explore.

One area for future work is to improve preprocessing methods for code-mixed language. Social media users tend to alter the spelling of offensive words with phonetic spellings, repeated letters or weird punctuation to avoid detection. Developing smarter normalization techniques for these variations could clean up the noise before tokenization and increase the accuracy of the model.

Another direction is the use of other contextual information during classification. Pulling in external knowledge bases or contextual language models may be helpful in identifying implicit hate speech, sarcasm or subtle abusive expressions that current models miss.

The data set in this study is mainly concerned with Hindi and Hinglish. Expanding it to other regional languages and dialects such as Bengali and Urdu could make multilingual moderation systems stronger. Cross lingual transfer learning may allow a model trained on one language to generalise to other Indic languages even if labelled data is scarce.

Future work may also look into multimodal hate speech detection. Online content nowadays are full of images, memes, and short videos to accompany text. Integrating text-based models with image and video analysis will help moderation systems identify harmful content in multiple formats.

Conclusion

This paper described the issue of hate speech detection on Hindi and Hinglish social media posts. A single dataset with over 93,000 labeled samples was formed through the merging of various publicly available datasets. This data enabled the comparison of various modeling techniques in dealing with code-mixed language. During the experiments, several classes of models were tested, such as traditional machine learning algorithms, deep learning architectures, and multilingual transformer models. The finding indicates that transformer models are superior to conventional machine learning models as well as sequential deep learning models in performing this task [4].

MuRIL was the best performing single model. This has probably been attributed to pretraining on Indian languages and transliterated text. XLM-RoBERTa was successful too because it was able to capture multilingual contextual information.

The weighted ensemble of MuRIL and XLM-RoBERTa was then built to enhance the accuracy of detection. The ensemble model had the highest overall performance with the F1-score of 0.7701.

These findings show that multilingual structure and region-specific language models can enhance the detection of hate speech in the context of code-mixed languages . The suggested framework shows how to create practical tools on the construction of automated moderation systems to be used in multilingual social media environments.

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1s, 2026, Pages: 3286-3292

References

[1] Schmidt, A., and Wiegand, M. (2017). A Survey on Hate Speech Detection using Natural Language Processing. Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media.

[2] Zampieri, M., Malmasi, S., Nakov, P., Rosenthal, S., Farra, N., and Kumar, R. (2019). SemEval-2019 Task 6: Identifying and Categorizing Offensive Language in Social Media. Proceedings of the 13th International Workshop on Semantic Evaluation.

[3] Patwa, P., Sharma, S., Pykl, S., Guptha, V., Kumari, G., Akhtar, M. S., Ekbal, A., Das, A., and Chakraborty, T. (2021). Overview of the HASOC Subtask at FIRE 2021: Hate Speech and Offensive Content Identification in English and Indo-Aryan Languages. Forum for Information Retrieval Evaluation.

[4] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A., Kaiser, Ł., and Polosukhin, I. (2017). Attention Is All You Need. Advances in Neural Information Processing Systems (NeurIPS).

[5] Devlin, J., Chang, M., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Proceedings of NAACL-HLT.

[6] Khanuja, S., Bansal, D., Mehtani, S., Khosla, S., Dey, A., Gopalan, B., Margam, D., Aggarwal, P., Nagipogu, R., Dave, S., and others (2021). MuRIL: Multilingual Representations for Indian Languages. Proceedings of ACL.

[7] Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., Grave, E., Ott, M., Zettlemoyer, L., and Stoyanov, V. (2020). Unsupervised Cross-lingual Representation Learning at Scale. Proceedings of ACL.

[8] Joulin, A., Grave, E., Bojanowski, P., and Mikolov, T. (2017). Bag of Tricks for Efficient Text Classification. Proceedings of EACL.

[9] Davidson, T., Warmsley, D., Macy, M., and Weber, I. (2017). Automated Hate Speech Detection and the Problem of Offensive Language. Proceedings of the International AAAI Conference on Web and Social Media (ICWSM).

[10] Badjatiya, P., Gupta, S., Gupta, M., and Varma, V. (2017). Deep Learning for Hate Speech Detection in Tweets. Proceedings of the 26th International World Wide Web Companion Conference (WWW Companion).

[11] Bohra, A., Vijay, D., Singh, V., Akhtar, S., and Shrivastava, M. (2018). A Dataset of Hindi-English Code-Mixed Social Media Text for Hate Speech Detection. Proceedings of the Second Workshop on Computational Modeling of People's Opinions, Personality, and Emotions in Social Media.

[12] Mandl, T., Modha, S., Majumder, P., Patel, D., Dave, M., Mandlia, C., and Patel, A. (2019). Overview of the HASOC Track at FIRE 2019: Hate Speech and Offensive Content Identification in Indo-European Languages. Proceedings of the Forum for Information Retrieval Evaluation (FIRE).

[13] Kapil, P., Sharma, S., Akhtar, M. S., and Ekbal, A. (2023). HHSD: Hindi Hate Speech Detection Using Hierarchical Multi-Task Learning. Proceedings of the International Conference on Computational Linguistics.