ItsAllGoodMan@LT-EDI-2025: Fusing TF-IDF and MuRIL Embeddings for Detecting Caste and Migration Hate Speech

Amritha Nandini KL, Vishal S, Giri Prasath R, Anerud Thiyagarajan, Sachin Kumar S

Amrita School of Artificial Intelligence, Coimbatore
Amrita Vishwa Vidyapeetham, India
{amrithanandini2003, vishalatmadurai, giriprasath017, anerud68511}@gmail.com, s_sachinkumar@cb.amrita.edu

Abstract

Caste and migration hate speech detection is a critical task in the context of increasingly multilingual and diverse online discourse. In this work, we address the problem of identifying hate speech targeting caste and migrant communities across a multilingual social media dataset containing Tamil, Tamil written in English script, and English. We explore and compare different feature representations, including TF-IDF vectors and embeddings from pretrained transformer-based models, to train various machine learning classifiers. Our experiments show that a Soft Voting Classifier that make use of both TF-IDF vectors and MuRIL embeddings performs best, achieving a macro F1 score of 0.802 on the test set. This approach was evaluated as part of the Shared Task on Caste and Migration Hate Speech Detection at LT-EDI@LDK 2025[1], where it ranked 6th overall.

1 Introduction

In India, caste and migration-based hate speech is a pervasive problem that has long been the focus of political discussion and still can be observed a lot in online forums and digital spaces. This carries severe real-world consequences, including psychological harm, the increase in social divisions, and the potential for inciting offline violence, particularly in linguistically diverse regions (Singh, 2025). This online discrimination takes place in numerous forms, including direct hate speech, derogatory remarks targeting specific castes and migration groups, cyberbullying of individuals based on their identity, threats of violence or social ostracism, and exclusion from online communities and groups.

With the boom of social media, such biases have found new avenues to spread, often under the disguise of free speech and anonymity. The spreading of hate speech targeting caste and migration groups not only normalizes discrimination but also reinforces harmful stereotypes, further marginalizing already vulnerable communities. Given the vast volume of online discourse and the rapid spread of harmful content, there is a pressing need to develop a system that is capable of recognizing and addressing such biases in real time.

Existing research on detection of hateful speech has primarily focused on widely spoken languages such as English. The complexity of code-mixed and regional language discourse prevalent in multilingual societies like India is often overlooked. Tamil, a widely spoken Dravidian language, frequently appears in code-mixed forms with English and other regional languages, making hate speech detection in Tamil code-mixed text a challenging task. The lack of sufficient annotated datasets with code-mixed text further complicate the problem.

Recent studies have explored a variety of transformer-based, machine learning, and data augmentation approaches for the detection of hate speech in Tamil, particularly in code-mixed and multilingual contexts. Using an ensemble of models such as XLM-RoBERTa, multilingual-cased BERT, and MuRIL, one of the best-performing models in the LT-EDI-EACL 2024 shared task achieved an F1-score of 0.82 (Singhal and Bedi, 2024). Another submission to this shared task evaluated 12 pre-trained transformer models in Indian multilingual language settings and found that MuRIL-Large was the most effective, with an F1-score of 0.81, which was obtained by ensembling the top-performing models (Pokrywka and Jassem, 2024). Another method experimented with transformers, FastText, and TF-IDF; mBERT did the best with an F1-score of 0.80 (Alam et al., 2024). To counter the difficulty of detecting masked abusive language in regional languages,(S et al., 2025) developed a system for Tamil and Malayalam by using supervised learning techniques on RoBERTa

[1] https://codalab.lisn.upsaclay.fr/competitions/21884

Proceedings of the Fifth Workshop on Language Technology for Equality, Diversity, Inclusion, pages 90–94

text embeddings. Researchers have also investigated multimodal hate speech detection that incorporates text, speech, and video data in addition to shared tasks. A study that compared several Tamil language models, such as Tamil-BERT, LaBSE, Hate-MuRIL, and MuRIL-Large-Cased, concluded that Tamil-BERT was the most successful (Mohan et al., 2025).

Another paper focused on detecting offensive language in Tamil-English code-switching by highlighting the potential of a hybrid system using both KANs and standard classifiers to improve detection accuracy (Jaidev et al., EasyChair, 2024). An averaging ensemble approach resulted in an accuracy score of 90.67% in identifying hate speech with mixed Tamil-English codes by using conventional machine learning approaches such as Support Vector Machine, Naive Bayes and ensemble methods (FHA et al., 2023). Different neural networks were explored, out of which a hybrid CNN-BiLSTM model adjusted for data imbalance, performed best for identifying offensive language in Dravidian languages (K et al., 2021).

Despite the substantial social impact of hate speech related to caste and migration, little research has been done on identifying it. The majority of current research focuses on hate speech in general, which leaves a gap in addressing these particular and culturally relevant types of online abuse. In this work, we explore machine learning models to detect hate speech related to caste and migration in Tamil code-mixed text using TF-IDF and pretrained model embeddings to identify the most effective approaches for handling this task.

2 Data

The dataset used for this task includes text samples from social media platforms, including posts that are general and those that are specifically related to caste or migration hate speech, along with the labels that correspond to these posts for the purpose of identifying hate speech (Rajiakodi et al., 2025). The dataset includes three different language representations: English, Tamil, and Tanglish (a code-mixed Tamil and English). The provided train and development datasets were merged into a single training dataset. An overview of this combined dataset's distribution across classification labels can be found in Table 1.

Label	Count
Caste/Migration Hate Speech	2399
Not Caste/Migration Hate Speech	3900

Table 1: Dataset distribution across classification labels on train and development datasets combined

3 Methodology

This section describes the methodology followed which includes data pre-processing, feature extraction and model training used in this study. The codebase is available at our GitHub repository[2].

3.1 Data Preprocessing

A number of text preprocessing procedures were used to make sure the dataset was clean and appropriate for classification. Initially, the text's hashtags were taken out and processed independently. A word segmentation model was employed to separate hashtags into meaningful components because they frequently contain compound words without spaces. To preserve their semantic meaning, the processed hashtags were subsequently added back to the main body of text.

In order to anonymize the users tagged, while preserving the conversation's structure, user mentions (such as @username) were replaced with the placeholder <USER>. The emoji library was also used to translate emojis into their textual descriptions, guaranteeing that the text retained the emojis' sentiment and meaning. Lastly, to standardize the input format, extra whitespace and newline characters were eliminated. By removing noise from the text, these preprocessing techniques assisted in preserving the most important linguistic information.

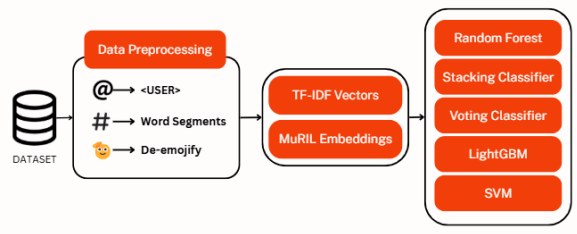

Figure 1: Methodology

3.2 Feature Extraction

Transformer-based embeddings and statistical methods were the two main strategies used for fea-

[2]https://github.com/amri-tah/ItsAllGoodMan-LT-EDI-2025

ture extraction for our task. Significant patterns in text have been captured using TF-IDF vectors which was used to train machine learning models. Furthermore, using the contextual understanding offered by these pre-trained language models, embeddings from mBERT, XLM-RoBERTa, and MuRIL were extracted and utilized as input features for machine learning models. To further investigate the effect of hybrid feature representations on classification performance, we experimented contatenating the most effective embeddings, TF-IDF and MuRIL.

3.3 Traditional Machine Learning Models

A range of machine learning models, including ensemble-based and individual classifiers, were investigated. The individual classifiers that were employed included XGBoost, logistic regression, decision trees, SVM, Random Forest, gradient boosting, and LightGBM.

A voting-based method aggregated three best performing classifiers, by averaging probability scores (soft voting) and by choosing the most often predicted class (hard voting). The strengths of several top-performing base models were combined using the stacking approach, and a logistic regression model was employed as the last decision layer.

3.4 Hyperparameter Tuning

Grid Search was used for hyperparameter tuning, in order to maximize the model performance. Various machine learning models such as random forest, logistic regression, etc, were tested with various learning rates, depth values, and weight adjustments. To evaluate model stability and make sure the models don't overfit, a 10-fold cross-validation technique was also applied.

4 Results

The results of several machine learning models on different text representations have been explored in this section. Initially, machine learning models were trained using TF-IDF vectors and embeddings from a number of pre-trained models, including mBERT, Tamil BERT, LaBSE, XLM-Roberta, and MuRIL (base and large). Of these, TF-IDF vectors and MuRIL large embeddings outperformed the others on the validation split. Following this, a combination TF-IDF vectors and MuRIL embeddings were used to to further improve accuracy and F1-scores.

Table 2 presents the classification performance of the best performing machine learning models using text representations: TF-IDF vectors, MuRIL embeddings, and their combination. The evaluation metrics considered are accuracy and macro F1-score, where higher values indicate better performance.

Model	Accuracy	Macro F1
TF-IDF Vectors		
Random Forest	**0.80**	**0.77**
Stacking Classifier	0.79	0.76
Voting Classifier (Soft)	0.77	0.73
MuRIL Embeddings		
Stacking Classifier	**0.78**	**0.75**
XGBoost	0.77	0.74
Voting Classifier	0.77	0.73
TF-IDF + MuRIL Embeddings		
Voting Classifier (Soft)	**0.79**	**0.77**
XGBoost	0.78	0.77
LightGBM	0.78	0.76

Table 2: ML Models for Each Embedding Type

The validation of our models trained were done using the 20% of the dataset provided to us for training. Using this validation set, Random Forest classifier outperformed the other models trained on TF-IDF vectors, achieving a macro F1-score of 0.77 and an accuracy of 0.80, whereas Stacking Classifier performed the best for models trained on MuRIL embeddings, with a macro F1-score of 0.75 and an accuracy of 0.78. Voting classifier and XGBoost trained on MuRIL embeddings gave similar results with an accuracy of 0.77 and F1-scores of 0.73 and 0.74, respectively. On combining MuRIL representations with TF-IDF vectors, an overall improvement in classification performance can be observed.

Overall classification performance was improved by combining MuRIL based embeddings with TF-IDF vectors and training it on Soft Voting Classifier with an accuracy score of 0.79 and F1 score of 0.77. Following closely behind, XGBoost and LightGBM returned comparable results with an accuracy of 0.78 and F1 scores of 0.77 and 0.76 respectively. These findings imply that improving classification performance requires using both contextual embeddings and traditional statistical features. When combined with TF-IDF, MuRIL embeddings helped to improve performance, but they did not outperform TF-IDF-based models on

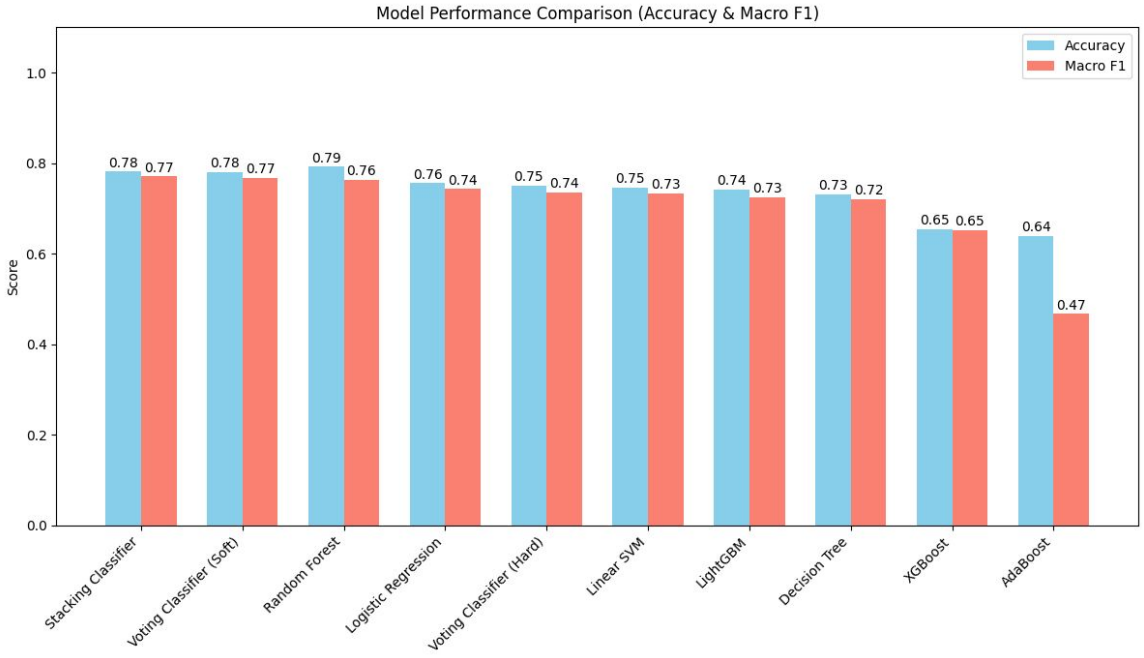

Figure 2: MuRIL + TF-IDF Model Training

their own. Overall, these results show how well hybrid feature representations work to produce reliable text classification outcomes.

In comparison to the best individual feature-based models (TF-IDF only and MuRIL only), the combination of MuRIL embeddings and TF-IDF vectors performed better on the held-out test set, obtaining the highest macro F1-score of 0.802. This supports how well contextual and statistical text representations work together to classify hate speech.

The term importance based on the frequency of explicit hateful words and keywords is useful in identifying whether a speech is hateful or not and is obtained by using statistical features such as TF-IDF. However, this feature alone might fail in some cases since it does not understand the semantic meaning behind these words, especially in code-mixed and multilingual contexts. This is where MuRIL embeddings comes into play, which, when combined with TF-IDF, has proven to be a good feature representation for the hate speech classification task.

5 Conclusion

This paper presents our system for the LT-EDI@LDK 2025 Shared Task on detecting caste and migration hate speech across Tamil, Tanglish, and English. We experimented with both TF-IDF vectors and transformer embeddings (especially MuRIL) as input features for a range of machine learning classifiers.

Our experiments clearly showed that combining traditional TF-IDF vectors with the contextual understanding from MuRIL embeddings produced the best outcome. Specifically, a Soft Voting Classifier using this hybrid TF-IDF + MuRIL feature set achieved the highest macro F1-score of 0.802 on the competition's test data. Using both TF-IDF and MuRIL together produced a better score than using either one individually. This likely happened because the two methods capture different kinds of useful information. TF-IDF finds key hate terms through frequency, while MuRIL understands the context and nuance, essential for the code-mixed and multilingual text we analyzed.

Our system using this method placed 6th overall in the shared task. This work shows that blending statistical text features with modern contextual embeddings offers a solid path forward for effectively detecting hate speech in complex, real-world linguistic scenarios like those found in Indian social media.

6 Limitations

Our model performances have been primarily validated on the provided LT-EDI@LDK 2025 dataset, therefore the generalization of the models on the full diversity of online caste and migration hate speech might be constrained.

References

Md Alam, Hasan Mesbaul Ali Taher, Jawad Hossain, Shawly Ahsan, and Mohammed Moshiul Hoque. 2024. CUET_NLP_Manning@LT-EDI 2024: Transformer-based approach on caste and migration hate speech detection. In *Proceedings of the Fourth Workshop on Language Technology for Equality, Diversity, Inclusion*, pages 238–243, St. Julian's, Malta. Association for Computational Linguistics.

Shibly FHA, Uzzal Sharma, and HMM. Naleer. 2023. Development of an efficient method to detect mixed social media data with tamil-english code using machine learning techniques. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, 22(2).

K Jaidev, Munnangi Pranish Kumar, Jampala Sai Chandana, Charishma Chowdary, and Sachin Kumar. EasyChair, 2024. Offensive text detection: Exploring traditional classifiers, ensemble models, and kolmogorov arnold networks in code-mixed tamil-english text. EasyChair Preprint 15581.

Sreelakshmi K, Premjith B, and Soman Kp. 2021. Amrita_CEN_NLP@DravidianLangTech-EACL2021: Deep learning-based offensive language identification in Malayalam, Tamil and Kannada. In *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, pages 249–254, Kyiv. Association for Computational Linguistics.

Jayanth Mohan, Spandana Reddy Mekapati, Premjith B, Jyothish Lal G, and Bharathi Raja Chakravarthi. 2025. A multimodal approach for hate and offensive content detection in tamil: From corpus creation to model development. volume 24, New York, NY, USA. Association for Computing Machinery.

Jakub Pokrywka and Krzysztof Jassem. 2024. kubapok@LT-EDI 2024: Evaluating transformer models for hate speech detection in Tamil. In *Proceedings of the Fourth Workshop on Language Technology for Equality, Diversity, Inclusion*, pages 196–199, St. Julian's, Malta. Association for Computational Linguistics.

Saranya Rajiakodi, Bharathi Raja Chakravarthi, Rahul Ponnusamy, Shunmuga Priya MC, Prasanna Kumar Kumaresan, Sathiyaraj Thangasamy, Bhuvaneswari Sivagnanam, Balasubramanian Palani, Kogilavani Shanmugavadivel, Abirami Murugappan, and Charmathi Rajkumar. 2025. Findings of the Shared Task on Caste and Migration Hate Speech Detection. In *Proceedings of the Fifth Workshop on Language Technology for Equality, Diversity and Inclusion*, Italy. Fifth Conference on Language, Data and Knowledge (LDK2025).

Nishanth S, Shruthi Rengarajan, S Ananthasivan, Burugu Rahul, and Sachin Kumar S. 2025. ANSR@DravidianLangTech 2025: Detection of abusive Tamil and Malayalam text targeting women on social media using RoBERTa and XGBoost. In *Proceedings of the Fifth Workshop on Speech, Vision, and Language Technologies for Dravidian Languages*, pages 711–715, Acoma, The Albuquerque Convention Center, Albuquerque, New Mexico. Association for Computational Linguistics.

Dhyan Singh. 2025. Dalits' encounters with casteism on social media: a thematic analysis. volume 28, pages 335–353. Routledge.

Kriti Singhal and Jatin Bedi. 2024. Transformers@LT-EDI-EACL2024: Caste and migration hate speech detection in Tamil using ensembling on transformers. In *Proceedings of the Fourth Workshop on Language Technology for Equality, Diversity, Inclusion*, pages 249–253, St. Julian's, Malta. Association for Computational Linguistics.