SSN_IT_HATE@LT-EDI-2025: Caste and Migration Hate Speech Detection

Maria Nancy C[1], Radha N [2], Swathika R[3]
[1]Annai Veilankanni's College of Engineering, Nedungundram , India
[2,3] Sri Sivasubramaniya Nadar College of Engineering, Kalavakkam, India
nancycse13@gmail.com[1]
radhan@ssn.edu.in[2]
swathikar@ssn.edu.in[3]

Abstract

This paper proposes a transformer-based methodology for detecting hate speech in Tamil, developed as part of the shared task on Caste and Migration Hate Speech Detection. Leveraging the multilingual BERT (mBERT) model, we fine-tune it to classify Tamil social media content into caste/migration-related hate speech and nonhate speech categories. Our approach achieves a macro F1-score of 0.72462 in the development dataset, demonstrating the effectiveness of multilingual pretrained models in low-resource language settings. The code for this work is available on github Hate-Speech-Deduction.

1 Introduction

Hate speech poses a threat to marginalized communities, especially those affected by caste discrimination and migration. In India, these sensitive issues often fuel online hate, commonly expressed in regional languages like Tamil. Addressing such content is vital to fostering respectful digital spaces. Automated detection of hate speech in Tamil presents challenges due to its low-resource nature, complex morphology, frequent code-mixing with English, and informal writing style. Existing tools and datasets often prioritize high-resource languages, leaving Dravidian languages underrepresented. We propose a multilingual transformer-based system to identify caste- and migration-related hate speech in Tamil social media. Using a curated, annotated dataset, we fine-tune the bert-base-multilingual-cased model with BERT tokenization, cross-entropy loss, and evaluate performance via standard metrics. Predictions on unseen test data gauge generalization ability. This study contributes to ethical AI by addressing identity-based harm in underrepresented languages. By applying advanced NLP methods, we aim to promote safer, more inclusive online platforms for vulnerable groups.

2 Literature Survey

Hate speech detection has emerged as a vital area of natural language processing (NLP), focusing on identifying abusive, derogatory, or inciting content across multiple platforms and languages. Early work in this field primarily employed statistical machine learning models, such as Support Vector Machines (SVM), Naive Bayes, and Logistic Regression, with hand-crafted features like n-grams and TF-IDF vectors (Zampieri et al., 2020). With the advent of deep learning, models such as Convolutional Neural Networks (CNNs) and Long Short-Term Memory networks (LSTMs) demonstrated superior performance, especially on noisy and informal texts common in social media (Badjatiya et al., 2017). However, these models often struggled with contextual understanding and multilingual settings. Transformer-based models like BERT (Devlin et al., 2019) revolutionized NLP by introducing contextualized embeddings and enabling transfer learning. These models improved the performance of hate speech classification in multiple languages. (Sutejo and Lestari, 2018) For low-resource languages, multilingual variants like mBERT and XLM-RoBERTa (Conneau et al., 2020) are particularly valuable. In the Indian context, (Bhattacharya et al., 2021) studied hate speech in Tamil and Malayalam, addressing challenges such as code-mixing, orthographic variation, and dialect diversity. (Patankar et al., 2022) developed transformer-based systems for abusive comment detection in Tamil, and (Roy et al., 2022) proposed a deep ensemble framework for detecting hate in multiple Dravidian languages. A landmark overview by (Rajiakodi et al., 2024) detailed the objectives, dataset structure, and methodological landscape of the LT-EDI shared task, with specific focus on caste and migration hate speech in Tamil. Comprehensive surveys, such as (Fortuna and Nunes, 2018), have discussed key limitations in early hate speech

Proceedings of the Fifth Workshop on Language Technology for Equality, Diversity, Inclusion, pages 84–89
September 9, 2025 ©2025 Association for Computational Linguistics

research, advocating for more context-aware models. (Abro et al., 2020) emphasize the importance of robust datasets and suggest improvements in annotation quality and domain adaptation. Several recent works have emphasized ethical concerns, bias mitigation, and fairness in hate speech classifiers (Parker and Ruths, 2023). (Jahan and Oussalah, 2023) highlight ethical pitfalls and recommend model transparency and explainability. Multimodal approaches (Gomez et al., 2019);(Wu and Bhandary, 2020) combining text with images or audio have shown that non-textual features such as tone, pitch, and visual cues can enhance hate detection. While promising, these approaches are more resource-intensive and less feasible in text-only shared tasks. Further, enhancements in semantic understanding like sentiment integration (Zhou et al., 2021) and contextual embeddings (Malik et al., 2022) have contributed to improved classification accuracy. Techniques like SMOTE for balancing class distributions (Kovács et al., 2021) also play a critical role when dealing with imbalanced hate speech datasets. Taken together, these contributions form the basis of our methodological decisions for this shared task submission.

3 Proposed Methodology

3.1 Dataset Description

The dataset comprises Tamil-language texts collected from various social media platforms, reflecting real-world discourse and often containing informal, emotionally charged, or contextually nuanced language. Each entry in the dataset is annotated with a binary label indicating whether the content includes hate speech directed at caste or migration groups, or not. In total, the dataset includes 5512 training samples, 787 development samples, and 1576 test samples. These texts range from short phrases to longer posts, with many exhibiting informal spelling, colloquial expressions, and a high frequency of code-mixing between Tamil and English. Such linguistic diversity presents both opportunities and challenges for automatic classification. Our proposed system for caste and migration hate speech detection in Tamil is built upon a fine-tuned multilingual BERT (mBERT) model. We considered both mBERT and XLM-RoBERTa for this task, as both are widely used multilingual models that perform well on low-resource languages. However, we chose to work with mBERT because it is lighter and faster to train, which made it a better fit for our available resources. mBERT has also been shown to work well in similar tasks involving Tamil and other Dravidian languages. While XLM-RoBERTa might offer slightly better results in some cases, our initial experiments showed that mBERT still gave strong performance and was more efficient overall. Given these factors, we felt mBERT was the more practical choice for this study. The architecture includes multiple stages, starting from data preprocessing to model inference. This section elaborates on the overall workflow, including preprocessing, tokenization, model architecture, training, and evaluation.

3.2 Text Preprocessing

The first stage involves preprocessing the raw Tamil text from the dataset to standardize and clean the input. Since Tamil is a case-insensitive script, we did not apply any lowercasing, as it does not affect the language and may discard meaningful formatting in code-mixed text. Instead, we focused on removing URLs, mentions, hashtags, special characters, and redundant white spaces using regular expressions to clean the input without distorting its structure. However, we retained casing for English words in code-mixed content, as it may carry emphasis or mark named entities, which could be useful for classification. One of the key challenges in this task is the presence of code-mixed content, where users often switch between Tamil and English within a single sentence. In many cases, Tamil words are also transliterated using Roman script, making them harder to detect using standard tokenizers. In our current approach, we did not apply any special preprocessing for code-mixing or transliteration. Instead, we relied on the multilingual capabilities of mBERT, which is pretrained on multiple scripts and languages, including English and Tamil. While this provides some level of generalization, we acknowledge that the model may not fully capture the nuances of code-switched text or romanized Tamil. Additionally, redundant white spaces are stripped to produce cleaner and more consistent input for tokenization.

3.3 Tokenization

Once preprocessed, each text instance is tokenized using the bert-base-multilingual-cased tokenizer from the HuggingFace Transformers library. The tokenizer breaks the text into sub-word units, adds special tokens like [CLS] and [SEP], and generates input IDs, attention masks, and token type IDs. All sequences are padded or truncated to a fixed maximum length of 256 tokens to ensure uniformity during batch processing.

3.4 Model Architecture

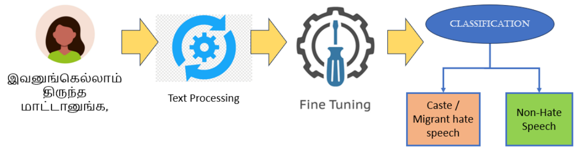

Figure 1: Workflow Diagram for Tamil Hate Speech Detection using mBERT

Figure 1 shows the proposed workflow of the model. We utilize the Bert for Sequence Classification model, which adds a classification head on top of the BERT encoder. The base model, bert-base-multilingual-cased, has 12 transformer layers and supports over 100 languages, including Tamil. The classification head is a fully connected layer that outputs two logits corresponding to the binary labels: caste/migration-related hate speech and nonhate speech.

3.5 Training Configuration

The model is fine-tuned in the labeled training set using a batch size of 32 and for 3 epochs. The optimizer used is Adam W with a learning rate of 2e-5 and weight decay to prevent overfitting.A linear learning rate scheduler was used to gradually reduce the learning rate during training. Although this scheduler supports warm-up, we did not apply it, as the number of warm-up steps was set to zero. The loss function used is CrossEntropyLoss, suitable for binary classification tasks. To facilitate efficient training and evaluation, the dataset is loaded using a custom PyTorch Dataset class and a Data-loader with shuffling enabled for the training set. Each batch is transferred to the GPU if available, ensuring accelerated computation.

3.6 Evaluation Strategy

After each epoch, the model is evaluated on the development set using a forward pass and argmax over output logits to generate predicted labels. Accuracy, precision, recall, and F1-score are calculated via the sklearn library to measure performance. In the final phase, the trained model predicts labels for the unseen test set, and results are saved in CSV format per the shared task protocol. This approach helps the model learn linguistic patterns and remain robust to the informal, noisy, and context-rich nature of Tamil social media. Leveraging mBERT's multilingual capabilities and fine-tuning on annotated domain-specific data, our system offers a practical solution for detecting hate speech in under-resourced languages. We evaluated model performance through experiments on the LT-EDI 2025 dataset, targeting hate speech against caste and migrant communities.

4 Experiment and Results

All experiments were carried out using PyTorch and Hugging Face Transformers on an NVIDIA Tesla V100 GPU, following the fine-tuning phase.

4.1 Evaluation Metrics

We used accuracy, precision, recall, and macro F1-score as our evaluation metrics. Among these, macro F1-score was prioritized due to the class imbalance and ethical weight of the task.

4.2 Development Set Results

Our best-performing model achieved a macro F1-score of 0.7246 on the development set. This indicated strong performance across both classes hate and non-hate speech despite the informal and code-mixed nature of the data.

4.3 Confusion Matrix

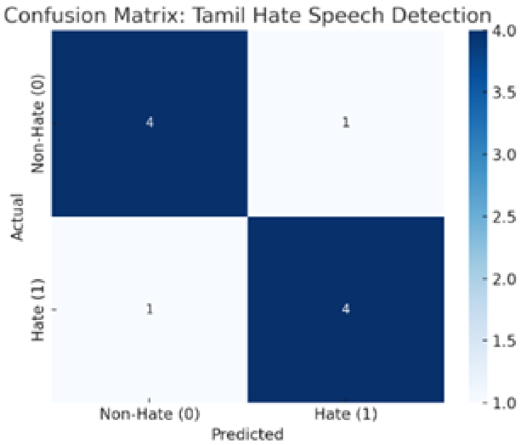

Figure 2: Confusion Matrix: Tamil Hate Speech detection against Caste / Migrated people

Figure 2 shows the model made balanced predictions across both classes. It successfully identified most hate speech posts while keeping false positives relatively low. The matrix also reveals some instances where non-hate speech was incorrectly classified as hate, likely due to emotionally charged but non-derogatory language.

4.4 Class-wise Performance

In evaluating different machine learning approaches (refer to Table 1 and Table 2) for the detection of Tamil hate speech, both the Logistic Regression and Multinomial Naive Bayes models demonstrated moderate performance, with overall F1-scores of 0.66 and 0.64, respectively.

Class	Precision	Recall	F1-Score
Non-Hate (0)	0.68	0.70	0.69
Hate (1)	0.63	0.61	0.62
Accuracy			**0.66**
Macro Avg	0.66	0.65	0.65
Weighted Avg	0.66	0.66	0.66

Table 1: Logistic Regression Classification Report.

Class	Precision	Recall	F1-Score
Non-Hate (0)	0.71	0.67	0.69
Hate (1)	0.58	0.63	0.60
Accuracy			**0.65**
Macro Avg	0.65	0.65	0.64
Weighted Avg	0.66	0.65	0.65

Table 2: Multinomial Naive Bayes Classification Report.

Class	Precision	Recall	F1-Score
Non-Hate (0)	0.73	0.73	0.73
Hate (1)	0.72	0.72	0.72
Accuracy			**0.73**
Macro Avg	0.725	0.725	0.725
Weighted Avg	0.725	0.725	0.724

Table 3: BERT Model Classification Report.

The Logistic Regression model achieved slightly better balance between precision and recall across both classes, with a macro average F1-score of 0.65, while Multinomial Naive Bayes trailed close behind. These traditional models showed a tendency to perform better on the Non-Hate class, while struggling slightly with correctly identifying hate speech, as reflected in the lower F1-scores for class 1. In contrast, the transformer-based model (refer Table 3) significantly outperformed these baselines and achieved an F1-score of 0.7246. This improvement highlights the strength of deep learning architectures, especially in capturing complex linguistic patterns and contextual relationships that are common in nuanced languages like Tamil. The transformer model maintained balanced precision and recall across both classes, which contributed to its stronger overall performance compared to the other model. Based on this observation we can conclude that while traditional classifiers can serve as useful baselines, transformer-based models are far more effective for tasks requiring deeper semantic understanding, such as hate speech detection in low-resource languages.

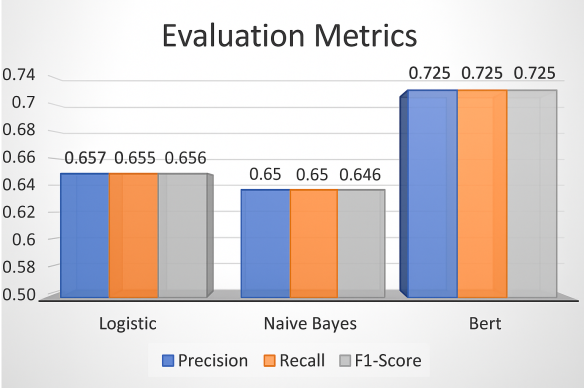

Figure 3: Evaluation Metrics using different methodologies

Figure 3 shows that among the models, BERT outperforms the others, achieving the highest values across all three metrics, with a consistent score of 0.725 for precision, recall and F1-score. Logistic regression follows, showing slightly lower but balanced scores: 0.657 for precision, 0.655 for recall, and 0.656 for F1-score. Naive Bayes, while close to Logistic Regression, performs slightly less effectively, particularly in F1-score, which stands at 0.646, compared to 0.65 for both precision and recall. This comparison highlights BERT as the most effective model in terms of classification performance for the given dataset.While the BERT-based model outperformed traditional classifiers with a macro F1-score of 0.7246, it's important to interpret these results in context. Detecting hate speech especially in low-resource, code-mixed languages like Tamil is inherently challenging due to informal language, slang, and subtle expressions of bias. The score reflects moderate success, indicating that the model captures many hateful patterns but still struggles with nuanced or indirect speech. Therefore, this result should be viewed as a strong baseline rather than a final solution. Future work can build on this by incorporating linguistic context, domain-specific pretraining, or more advanced multilingual models.

4.5 Error Analysis

To get a better understanding of where the model struggles, we looked closely at some of the examples it got wrong in the development set. One of the most common issues was with posts that used sarcasm or indirect language to express hate. These types of messages didn't contain obvious offensive words, so the model often misclassified them as non-hate. Another challenge came from code-mixed posts especially those that switched between Tamil and English. In many cases, the hateful meaning was embedded in the Tamil part, but the English portion made the message sound neutral. This seemed to confuse the model. We also noticed that slang, spelling variations, and informal language, which are common on social media, made it harder for the model to correctly identify hate speech.In some cases, the model predicted hate where there was none. These false positives often included posts with strong emotions or criticism, but not targeted hate. The model likely relied on certain keywords or tone, misinterpreting emotional expression as harmful content.Overall, these errors show that while the model performs well on average, it still has trouble with nuance, subtlety, and cultural context especially in a language like Tamil. Understanding these mistakes not only helps explain the results but also points to areas we could improve in the future, like handling sarcasm, improving code-mixed understanding, or training with more context rich data.

5 Limitations

While our model performs well, it has notable limitations. Tamil social media posts often blend languages and include slang or informal expressions, which can confuse the model, especially when hate is subtly or sarcastically conveyed. The dataset used is small and only labels posts as hate or non-hate, overlooking the nuances in harmful expression. Since the model relies on mBERT, it struggles with cultural context and its predictions can be difficult to interpret. This raises concerns about fairness and bias, particularly if it learns problematic patterns from the training data. Social media often mirrors societal biases, which the model may unintentionally reinforce. Though we didn't perform an in-depth bias or fairness analysis in this study, exploring variations across caste, gender, or identity groups is a vital direction for future work. A deeper investigation into dataset and model biases could support fairer and more responsible deployment.

6 Conclusion

Our work shows that a multilingual model like mBERT can be fine-tuned to effectively detect caste and migration-related hate speech in Tamil social media posts. Even with limited data and the challenges of informal, mixed-language text, the model achieved good performance. This approach highlights the potential of using existing language models to support low-resource languages and address real social issues. We hope our method encourages more research in this area and helps make online spaces safer and more inclusive for everyone.

6.1 Future Work

In the future, we plan to explore multimodal approaches that combine text with audio or visual cues, as these could help capture more subtle or sarcastic forms of hate speech. We're also looking to expand our dataset and include more specific labels such as distinguishing between caste-based and migration-related content to enhance the model's accuracy. Additionally, we aim to incorporate cultural context, improve the explainability of model decisions, and address potential biases. These steps are crucial for building systems that are not only more accurate but also more trustworthy and practical for real world use.

References

S. Abro, S. Shaikh, Z. H. Khand, Z. Ali, S. Khan, and G. Mujtaba. 2020. Automatic hate speech detection using machine learning: A comparative study. In *Proceedings of the 2020 Conference on Hate Speech Detection*.

Pinkesh Badjatiya, Shashank Gupta, Manish Gupta, and Vasudeva Varma. 2017. Deep learning for hate speech detection in tweets. In *Proceedings of the 26th International Conference on World Wide Web Companion*, pages 759–760.

Punyajoy Bhattacharya, Prateek Mishra, Indira Bhattacharya, and Amitava Das. 2021. Hate speech detection in low-resource languages: A case study on tamil and malayalam. In *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, pages 93–101.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, et al. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT 2019*, pages 4171–4186.

Paula Fortuna and Sérgio Nunes. 2018. A survey on automatic detection of hate speech. *ACM Computing Surveys (CSUR)*, 51(4):1–30.

Ricardo Gomez, Michail Zervakis, and Björn Schuller. 2019. Multimodal hate speech detection: Combining text and visual features. In *Proceedings of the 2019 International Conference on Multimodal Interaction*, pages 602–606.

Md Shad Jahan and Mourad Oussalah. 2023. A review of nlp-based hate speech detection. *Information Processing & Management*, 60(2):102057.

Gábor Kovács, András Szabó, and Richárd Farkas. 2021. Addressing data scarcity in hate speech detection with external resources. In *Proceedings of the 11th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 25–34.

Junaid S Malik, N S Muralidhar, and Rakesh Tiwari. 2022. A comparative study of deep learning models for hate speech detection. *Procedia Computer Science*, 199:266–273.

Sage Parker and Derek Ruths. 2023. Assessing bias and fairness in hate speech detection systems. *ACM Transactions on the Web (TWEB)*, 17(1):1–23.

Siddhesh Patankar, Shubham Suryawanshi, and Bharathi Raja Chakravarthi. 2022. Abusive comment detection in tamil. In *Proceedings of the Second Workshop on Speech and Language Technologies for Dravidian Languages*, pages 66–72.

Saranya Rajiakodi, Bharathi Raja Chakravarthi, Rahul Ponnusamy, Prasanna Kumar Kumaresan, Sathiyaraj Thangasamy, Bhuvaneswari Sivagnanam, and Charmathi Rajkumar. 2024. Overview of shared task on caste and migration hate speech detection. In *Proceedings of the Fourth Workshop on Language Technology for Equality, Diversity and Inclusion (LT-EDI 2024)*, Malta. EACL.

Prasenjit Kumar Roy, Soham Ghosh, and Animesh Mukherjee. 2022. Deep ensemble framework for hate speech detection in dravidian languages. In *Proceedings of the DravidianLangTech@ACL 2022*, pages 153–158.

Tony L Sutejo and Dwi P Lestari. 2018. Indonesian hate speech detection using deep learning. In *2018 International Conference on Asian Language Processing (IALP)*, pages 254–257. IEEE.

Chien-Sheng Wu and Ujjwal Bhandary. 2020. Hate speech detection in videos using multimodal cues. In *Proceedings of the 2020 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, pages 1–6. IEEE.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2020. Semeval-2020 task 12: Multilingual offensive language identification in social media (offenseval 2020). In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1425–1447.

Xinyu Zhou, Xilun Chen, and Yaqing Wang. 2021. Enhancing hate speech detection with sentiment knowledge. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics*, pages 2764–2774.