DravidianCodeMix 2025: Empirical Analysis of Classical Machine Learning Approaches in Tamil, Malayalam, and Tulu Code-Mixed Offensive Content Classification

Shudapreyaa R S[1], Surya U S[1], Swetha M[1] and Sandeep P S[1]

[1]*Kongu Engineering College, Tamil Nadu, India*

Abstract

A critical responsibility for maintaining healthy online communication, particularly on social media platforms where multilingual interactions are prevalent, is identifying offensive language in texts that contain Dravidian code. Creating machine learning models to categorize offensive information in Tamil, Malayalam, and Tulu—three Dravidian languages—is the main goal of this project. We tested and implemented a variety of algorithms to find the best method for each language. According to experimental results, Linear SVC performed best for Tulu, while Random Forest produced better results for Malayalam, and Logistic Regression outperformed the other models for Tamil. These results show that no single algorithm is uniformly dominant across all Dravidian languages, underscoring the significance of language-specific algorithm selection in offensive language identification.

1. Introduction

Code-mixed communication, in which users combine regional languages with English, has increased in popularity due to the rapid development of social media platforms. Several of these, including Dravidian languages like Tamil, Malayalam, and Tulu, are frequently used in casual online conversations that blend several scripts and vocabularies. Although it allows for more freedom of expression, this also creates opportunities for abusive and offensive language, which can hurt people and interfere with constructive online relationships. The lack of annotated datasets, translation problems, and spelling variances makes it especially difficult to identify such harmful information in code-mixed texts[1]. By automatically identifying objectionable content, machine learning approaches present a viable way to overcome these issues. In this study, we use various algorithms to assess offensive language recognition in Dravidian code-mixed languages[2]. We find that Linear SVC achieves the maximum accuracy for Tulu, Random Forest produces better results for Malayalam, and Logistic Regression performs best for Tamil. These findings demonstrate the necessity of using language-specific strategies when developing efficient systems for identifying harmful languages in a variety of multilingual situations.

2. Literature Survey

Every day, millions of comments are left on the uploaded postings due to the rise of netizen culture and social media. The usage of derogatory language in user comments has dramatically increased. Online comments that contain abusive language initiate cyberbullying[3], which targets both a group of people (a certain nation, age, and religion) and an individual (a politician, celebrity, or product). Automated detection and analysis of abusive language in online comments are crucial. In the literature, there have been multiple attempts to identify abusive language in the English language. NB, SVM, IBK, Logistic, and JRip are five different machine learning models, while CNN, LSTM, BLSTM, and CLSTM[4] are

Forum for Information Retrieval Evaluation, December 17-20, 2025, India

*Corresponding author.

shudapreyaa@gmail.com (S. R. S); suryaofficial2021@gmail.com (S. U. S); swetham.22cse@kongu.edu (S. M); sandeepps.22cse@kongu.edu (S. P. S)

0000-0043-2073-426X (S. R. S); 0000-0123-27744-246X (S. U. S); 0000-0123-27744-246X (S. M); 0000-0123-27744-246X (S. P. S)

four deep learning models that we use in this study to recognize abusive language in Urdu and Roman Urdu comments.

Tanjim Mahmud et al.[5] have proposed a system that creates sophisticated machine learning and deep learning models for identifying child abusive texts in the Bengali language on online platforms; this study tackles the pressing problem of child abuse in digital communications. The main objective of this project is to develop a useful tool for precisely recognizing abusive content to aid in the prevention of child abuse. This model differentiates between abusive and non-abusive material by combining deep learning methods with natural language processing (NLP) approaches.

Dhanyashree G et al. [6] highlighted that while social networks serve as major platforms for engagement and communication, they are also increasingly misused for gender-based abuse, particularly targeting women with demeaning and harassing remarks. Their study, focusing on Malayalam and Tamil YouTube comments, aimed to identify explicit abuse, implicit bias, stereotypes, and coded language. To address this, they evaluated multiple machine learning models, including Support Vector Machines (SVM), Logistic Regression (LR), and Naive Bayes classifiers, for categorizing comments into abusive and non-abusive groups.

Anwar Hossain et al[7]. have proposed in this system that one of the major problems significantly impacting society is the spread of hate speech on social media, which contributes to an increase in violence, discrimination, and societal disintegration. Because of adversarial manipulations and cultural, linguistic, and contextual complexities, the task of identifying hate speech is inherently complex. In this work, we methodically examine how well LLMs perform in identifying hate speech in a variety of geographical contexts and multilingual datasets. Our research offers a novel assessment approach that takes into account three factors: robustness to adversarially created text, geography-aware contextual detection, and binary classification of hate speech.

3. Materials and Methods

3.1. Taskset Description

The study explores training datasets from three Dravidian code-mixed languages—Tamil, Malayalam, and Tulu—made up of real social media comments that freely mix English and native scripts. While all three datasets contain offensive and non-offensive labels, the amount of detail in these annotations differs across languages. The Tamil dataset is the most detailed, with separate categories for targeted insults aimed at individuals, groups, or other entities, along with untargeted insults and comments written in other languages [8]. The Malayalam dataset is balanced but more challenging to work with because of its rich morphology, shifting dialects, and frequent code-switching, which can make offensive expressions harder to detect [9]. The Tulu dataset is smaller in size but still meaningful, offering useful insights for low-resource language research by capturing typical patterns of neutral, targeted, and untargeted offensive speech found online. When comparing models across these datasets, each language shows a different best performer: Logistic Regression works well for Tamil due to its strength with sparse text features, Random Forest handles Malayalam effectively by modeling its non-linear linguistic patterns, and Linear SVC suits Tulu because of its stability with limited training data. Overall, the results emphasize how linguistic characteristics, dataset size, and annotation depth play a crucial role in determining which machine learning approach is most effective for detecting offensive content in Dravidian code-mixed text.

Tamil Dataset: Labels include Not Offensive, Targeted Insult (Individual, Group, Other), Untargeted, and Other Language.

Tulu Dataset: Classified into Not Offensive, Offensive Targeted, and Offensive Untargeted.

Malayalam Dataset: Similar to Tamil, with categories for Not Offensive, Targeted Insults, Untargeted, and Other Language.

Table 1
Sample training texts from Tamil dataset

Text	Label
movie vara level la Erika poguthu	Not offensive
I love Ajith Kumar Vivegam movie inki mjy bht achi lgi	Not-Tamil
Dei Rajini pavam da ne varaven poraven ellam karithuppatha koraiya pesittu erukkan 1:58	Offensive_Targeted_Insult_Group
Naaa arasiyaikkulu varuvathu uruthi.....Rajini tells this dialog since 96. Semme joke.	Offensive_Targeted_Insult_Individual
Correct. Enga apa military da oodi vilaiyada solli tharuvayaa	Offensive_Targeted_Insult_Other
Intha maari comments ku like kekuravangala india va vittu veliya annupanum	Offensive_Untargeted

Table 2
Sample training texts from Malayalam dataset

Text	Label
Ee cinemayude story kollam enn thonnunnu	Not offensive
Mere naam ka kachra karo sab mil k	Not-Malayalam
Trailer kandal ariyam nalla oombiya padam aannun	Offensive_Targeted_Insults_Group
Al veruppikkal Mammotty ude vere oru adipoli veruppikkal padama-yarikku ith	Offensive_Targeted_Insults_Individual
Nalla assal moonjjal thank you lola	Offensive_Untargeted

Table 3
Sample training texts from Tulu dataset

Text	Label
Nama yeth parndala Steady Ulla	Not offensive
Sonu nigam da Shreya ghoshal background g panka...	Offensive_Untargeted
Savda comedy kapikad	Offensive_Targeted

3.2. Data Collection

The Data Collection module is in charge of compiling code-mixed textual data in Tamil, Malayalam, and Tulu from a variety of sources, including forums, social networking sites, and publicly accessible datasets. This module guarantees that the dataset has a variety of content categories, including slang, dialects, and samples that are both offensive and non-offensive. Training machine learning models that can reliably detect harmful content in multilingual and code-mixed environments requires a carefully curated dataset.

3.3. Data Pre-processing

Preprocessing was essential for preparing the text for classification and involved the following steps:

3.3.1. Lowercasing

All characters were converted to lowercase to maintain consistency and avoid treating words like 'Good' and 'good' as different tokens.

3.3.2. Noise Removal

Unwanted elements such as URLs, mentions (@username), hashtags, numbers, punctuation, and special characters were removed to reduce irrelevant .

3.3.3. Language Filtering

Only Tamil and English characters were retained by restricting the text to the corresponding Unicode ranges, ensuring that irrelevant scripts and symbols were excluded.

3.3.4. Whitespace Normalization

Multiple spaces were collapsed into a single space, and leading/trailing whitespace was removed, making the text cleaner and more uniform.

3.4. Feature Extraction

In order to make machine learning models understand preprocessed text, the Feature Extraction module converts it into numerical representations. Word embeddings, term frequency-inverse document frequency (TF-IDF) vectors, and n-gram generation are examples of common methods. In order to capture contextual nuances, this module may also include linguistic or semantic elements unique to code-mixed Dravidian texts. Effective feature extraction improves the model's capacity to reliably distinguish between offensive and non-offensive content.

3.5. Model Training and Selection

To process the feature-rich dataset, the Model Training and Selection module uses a variety of machine learning algorithms. We use language-specific data to train and evaluate algorithms like Linear SVC, Random Forest, and Logistic Regression [10]. They evaluate each model using performance indicators such as F1-score, recall, accuracy, and precision. To ensure optimal performance in offensive language identification, the best-performing algorithm is chosen for each language based on these evaluations.

4. Results and Discussion

The results show that abusive language detection behaves differently across Tamil, Malayalam, and Tulu, proving that a single model cannot work equally well for all. Logistic Regression gave the best results for Tamil because the dataset was relatively balanced, which made it easier for a linear model to separate offensive and non-offensive text while still capturing small differences between categories. For Malayalam, Random Forest performed better as its ensemble of decision trees could manage the imbalance across classes and deal with the greater variation in offensive expressions. It also showed more robustness to spelling changes and noisy code-mixed text, which are common in Malayalam. In the case of Tulu, the dataset was much smaller, and complex models tended to overfit. Linear SVC was more effective here since its margin-based classification and ability to handle sparse TF–IDF features helped it generalize better in low-resource conditions. Overall, these findings highlight that the choice of model depends strongly on the size, balance, and linguistic characteristics of each dataset, and that tailoring algorithms to language-specific needs leads to more reliable offensive language detection.

4.1. Performance Metrics

To give a thorough evaluation of the efficacy of the offensive language detection models, their performance was assessed using common classification measures, such as accuracy, precision, recall, and F1-score. While precision shows the percentage of offensive texts successfully detected out of all those projected to be offensive, accuracy gauges how accurate predictions are overall. The F1-score offers a balanced metric that combines precision and recall, whereas recall evaluates the model's capacity to recognize every instance of actual offensive behavior. The findings of the experiment showed that Linear SVC was the best for Tulu, Random Forest was the best for Malayalam, and Logistic Regression

had the best performance metrics for Tamil. These measurements show that the system can consistently differentiate objectionable content from non-offensive text in code-mixed Dravidian datasets, underscoring the significance of choosing language-specific methods.

Table 4
Performance of Logistic Regression Classifier across Different Offensive and Non-Offensive Categories in Tamil Text

Classifier	Class label	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Logistic Regression	Not-Offensive	0.76	0.80	0.97	0.87
Logistic Regression	Offensive Targeted Insult Individual	0.76	0.45	0.13	0.21
Logistic Regression	Offensive Targeted Insult Group	0.76	0.41	0.14	0.21
Logistic Regression	Offensive Untargeted	0.76	0.45	0.24	0.31
Logistic Regression	Offensive Targeted Insult Other	0.76	0.00	0.00	0.00
Logistic Regression	Other Language	0.76	0.87	0.63	0.73

Table 5
Performance of Random Forest Classifier across Different Offensive and Non-Offensive Categories in Malayalam Text

Classifier	Class label	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Random Forest	Not-Offensive	0.94	0.95	0.99	0.97
Random Forest	Offensive Targeted Insult Individual	0.94	0.83	0.49	0.62
Random Forest	Offensive Targeted Insult Group	0.94	0.78	0.23	0.36
Random Forest	Offensive Untargeted	0.94	0.68	0.40	0.50
Random Forest	Other Language	0.94	0.87	0.71	0.78

Table 6
Performance of Linear SVC Classifier across Different Offensive and Non-Offensive Categories in Tulu Text

Classifier	Class label	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Linear SVC	Not-Offensive	0.90	0.94	0.94	0.94
Linear SVC	Offensive Targeted	0.90	0.85	0.67	0.75
Linear SVC	Offensive Untargeted	0.90	0.82	0.91	0.87

5. Error Analysis

Looking more closely at the results, we found that all three models struggled with classes that had very few training examples. In Tamil, Logistic Regression performed well for major categories but completely failed to detect the "Offensive Targeted Insult Other" class, where recall dropped to 0.0 due to insufficient data for the model to learn meaningful patterns. Similarly, the Random Forest classifier for Malayalam showed good overall accuracy but performed poorly on group insult categories because of dataset imbalance. For Tulu, Linear SVC handled the small dataset better than other models but still struggled with subtle targeted insults. Many errors also arose from slang, spelling variations, and context-dependent words. Overall, these issues highlight that data scarcity and ambiguous expressions remain key challenges, emphasizing the need for richer and more balanced datasets in the future.

6. Limitations

The suggested offensive language detection method performs admirably, but there are still a number of drawbacks. The models mostly rely on annotated datasets, which are small and do not fully represent the variety of regional dialects, slang, and code-mixed expressions in Tamil, Malayalam, and Tulu. Sentences with ambiguous context may cause misclassification since the models may find it difficult to discern between offensive and non-offensive word usage. Furthermore, real-time social media feeds with extremely informal or changing language patterns may cause the algorithm to perform worse. These drawbacks emphasize the necessity of using bigger, more varied datasets, as well as sophisticated context-aware methods to increase generality and accuracy.

7. Conclusion

A machine learning-based method was created in this study to identify abusive language in texts that contain Dravidian code, specifically in Tamil, Malayalam, and Tulu. With Linear SVC for Tulu, Random Forest for Malayalam, and Logistic Regression for Tamil, the studies showed how important it is to choose algorithms that are specific to a given language. The outcomes demonstrate how well feature extraction, meticulous preprocessing, and model training can address the difficulties posed by code-mixed and multilingual data.The suggested system offers a solid basis for automatic content filtering, even though some mistakes still occur because of unclear context, slang, and small datasets. All things considered, this work promotes safer online communication and establishes the framework for next advancements, such as context-aware algorithms and larger datasets for more reliable foul language identification.

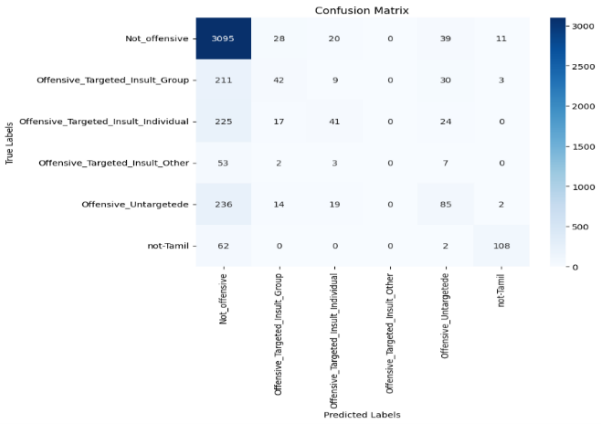

Figure 1: Confusion matrix for the Tamil dataset

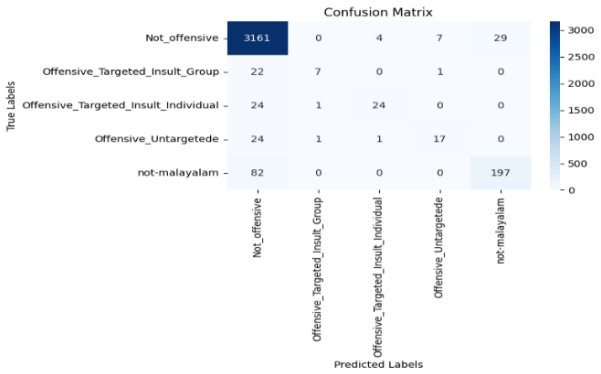

Figure 2: Confusion matrix for the Malayalam dataset

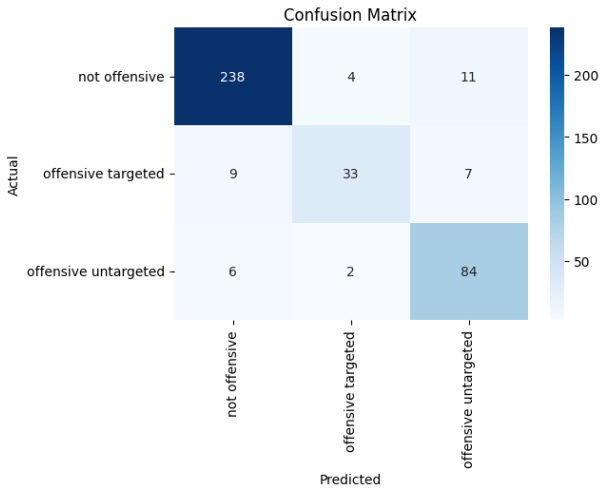

Figure 3: Confusion matrix for the Tulu dataset

Project Repository

The full source code for this project is available on
GitHub: GitHub Repository- SURYAULAGANATHAN

Declaration on Generative AI

In the course of preparing this manuscript, the author(s) employed the generative AI tool ChatGPT. Its use was limited to performing checks for grammar and spelling. Following this, the author(s) conducted a thorough review and revision of the text and assume full responsibility for the final published content.

References

[1] B. R. Chakravarthi, M. B. Jagadeeshan, V. Palanikumar, R. Priyadharshini, Offensive language identification in dravidian languages using mpnet and cnn, International Journal of Information Management Data Insights 3 (2023) 100151. URL: https://www.sciencedirect.com/science/article/pii/S2667096822000945. doi:https://doi.org/10.1016/j.jjimei.2022.100151.

[2] S. N, B. R. Chakravarthi, T. Durairaj, B. Bharathi, S. C. Navaneethakrishnan, P. K. Kumaresan, A. M D, P. R. Hegde, D. Vikram, Overview of the shared task on offensive language identification in dravidian code-mixed languages, in: Forum of Information Retrieval and Evaluation FIRE-2025, 2025.

[3] P. Fortuna, S. Nunes, A survey on automatic detection of hate speech in text, ACM Computing Surveys 51 (2018) 1–30.

[4] H. Mubarak, A. Abdelali, K. Darwish, Arabic offensive language on twitter: Analysis and experiments with classifiers, in: Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT), Association for Computational Linguistics, 2020.

[5] T. Mahmud, T. Akter, M. K. Uddin, M. T. Aziz, et al., Machine learning techniques for identifying child abusive texts in online platforms, in: 2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT), IEEE, 2024. doi:10.1109/ICCCNT61001.2024.10724830.

[6] G. Dhanyashree, K. Kalpana, A. Lekhashree, K. Arivuchudar, R. Arthi, B. Sahitya, J. Pavithra, S. Johnson, Linguaists@dravidianlangtech 2025: Abusive tamil and malayalam text targeting women on social media, in: Proceedings of the Fifth Workshop on Speech, Vision, and Language

Technologies for Dravidian Languages, Association for Computational Linguistics, 2025, pp. 682–687. URL: https://aclanthology.org/2025.dravidianlangtech-1.116.

[7] A. Hossain Zahid, M. K. Roy, S. Das, Evaluation of hate speech detection using large language models and geographical contextualization, arXiv preprint arXiv:2502.19612 (2025).

[8] B. R. Chakravarthi, R. Priyadharshini, N. Jose, T. Mandl, P. K. Kumaresan, R. Ponnusamy, J. P. McCrae, E. Sherly, et al., Findings of the shared task on offensive language identification in tamil, malayalam, and kannada, in: Proceedings of the first workshop on speech and language technologies for Dravidian languages, 2021, pp. 133–145.

[9] B. R. Chakravarthi, R. Priyadharshini, V. Muralidaran, N. Jose, S. Suryawanshi, E. Sherly, J. P. McCrae, Dravidiancodemix: Sentiment analysis and offensive language identification dataset for dravidian languages in code-mixed text, Language Resources and Evaluation 56 (2022) 765–806.

[10] A. M. D, D. Vikram, B. R. Chakravarthi, P. R. Hegde, Overcoming low-resource barriers in tulu: Neural models and corpus creation for offensive language identification, 2025. URL: https://arxiv.org/abs/2508.11166. arXiv:arXiv:2508.11166.