JustATalentedTeam@DravidianLangTech 2025: A Study of ML and DL approaches for Sentiment Analysis in Code-Mixed Tamil and Tulu Texts

Ponsubash Raj R, Paruvatha Priya B, Bharathi B
Department of Computer Science and Engineering
Sri Sivasubramania Nadar College of Engineering
ponsubashraj2370043@ssn.edu.in
paruvathapriya2370053@ssn.edu.in
bharathib@ssn.edu.in

Abstract

The growing prevalence of code-mixed text on social media presents unique challenges for sentiment analysis, particularly in low-resource languages like Tamil and Tulu. This paper explores sentiment classification in Tamil-English and Tulu-English code-mixed datasets using both machine learning (ML) and deep learning (DL) approaches. The ML model utilizes TF-IDF feature extraction combined with a Logistic Regression classifier, while the DL model employs FastText embeddings and a BiLSTM network enhanced with an attention mechanism. Experimental results reveal that the ML model outperforms the DL model in terms of macro F1-score for both languages. Specifically, for Tamil, the ML model achieves a macro F1-score of 0.46, surpassing the DL model's score of 0.43. For Tulu, the ML model significantly outperforms the DL model, achieving 0.60 compared to 0.48. This performance disparity is more pronounced in Tulu due to its smaller dataset size of 13,308 samples compared to Tamil's 31,122 samples, highlighting the data efficiency of ML models in low-resource settings. The study provides insights into the strengths and limitations of each approach, demonstrating that traditional ML techniques remain competitive for code-mixed sentiment analysis when data is limited. These findings contribute to ongoing research in multilingual NLP and offer practical implications for applications such as social media monitoring, customer feedback analysis, and conversational AI in Dravidian languages.

1 Introduction

With the growing prevalence of code-mixed text on social media, there is an increasing need for effective NLP tools to analyze and interpret such data. Code-mixing, the blending of words or phrases from multiple languages within the same sentence, reflects real-world multilingual communication but introduces complexities in text analysis. Sentiment analysis of code-mixed data is particularly challenging due to the lack of standardized grammar, varying transliterations, and diverse language structures.

This paper focuses on addressing these challenges in Tamil and Tulu code-mixed sentiment analysis by comparing two different methodologies: a machine learning (ML)-based approach and a deep learning (DL)-based approach. While the ML model leverages character-level n-grams and a logistic regression classifier, the DL model employs a BiLSTM architecture enhanced with attention mechanisms to capture semantic and contextual relationships in the text.

Although the performance of the DL approach is comparable to that of the ML approach in the Tamil code-mixed dataset, the difference in performance is found to be larger in the case of the Tulu code-mixed dataset, likely due to the smaller dataset. By evaluating the performance of these methodologies, this paper provides insights into the strengths and limitations of each approach, offering guidance for future work in code-mixed NLP tasks.

The paper begins with a review of related work in sentiment analysis for code-mixed text, highlighting previous research in this domain. In Section 3, the proposed methodologies, detailing the ML and DL approaches used for analysis are presented. Finally, the results are discussed, providing key insights and inferences drawn from the comparative evaluation.

2 Related Work

The study of sentiment analysis in Dravidian code-mixed text has recently gained attention, addressing the unique challenges of low-resource languages and their complex linguistic patterns. Chakravarthi et al. (2020a) focused on developing and evaluating models for Tamil-English and Tulu-English datasets, highlighting the need for effective tools to

Proceedings of the Fifth Workshop on Speech, Vision, and Language Technologies for Dravidian Languages, pages 351–355

analyze multilingual social media content. Hegde et al. (2022) introduced a valuable dataset of annotated YouTube comments and evaluated sentiment analysis using machine learning models such as logistic regression, random forest, and BERT, providing a foundation for future research in this area.

Expanding on these efforts, Hegde et al. (2023) presented findings from a shared task on sentiment analysis in code-mixed texts. The growing interest in sentiment analysis for Dravidian code-mixed languages is further reflected in initiatives aimed at benchmarking models for multilingual datasets. Kumar et al. (2024) discuss these initiatives, highlighting advancements in sentiment analysis for social media content.

To enhance sentiment classification in code-mixed texts, Puranik et al. (2021) employed pre-trained models like ULMFiT and multilingual BERT, fine-tuning them on the code-mixed dataset, its transliteration (TRAI), an English translation (TRAA) of the transliterated data, and a combination of all three. Similarly, Balaji et al. (2020) analyzed different feature extraction techniques, including count vectorization, LSTM, and BERT embeddings, comparing their effectiveness across various machine learning models. While sentiment analysis in high-resource languages like English has been extensively studied, research on Dravidian code-mixed languages, particularly Tamil and Tulu, remains in its early stages. Ponnusamy et al. (2023) emphasize the need for robust models capable of handling informal grammar and mixed scripts, underscoring the importance of tailored approaches for low-resource languages.

Early-stage research on sentiment analysis in Dravidian languages has paved the way for further exploration of text classification and emotion detection. Rachana et al. (2023) discuss the increasing interest in analyzing code-mixed text, encouraging the development of more inclusive and diverse language processing tools. Likewise, Chakravarthi et al. (2020b) focus on sentiment analysis in Tamil-English and Malayalam-English code-mixed texts, addressing the complexities introduced by mixed scripts and informal syntax in social media data.

Various machine learning approaches have been employed to enhance sentiment classification in multilingual online content. Kanta and Sidorov (2023) examined sentiment detection in Tamil-English and Tulu-English code-mixed social media text, using machine learning techniques to improve emotion recognition in such contexts. Similarly, Bharathi and Samyuktha (2021) and Varsha et al. (2022) explored machine learning-based approaches for sentiment analysis, with the latter also incorporating transformer models to refine classification performance.

The proposed models in the paper and their predictions are submitted for the Shared Task on Sentiment Analysis in Tamil and Tulu: DravidianLangTech@NAACL 2025 Durairaj et al. (2025). Given the frequent blending of Dravidian languages with English on social media platforms, sentiment analysis remains a challenging task. Recent studies have leveraged both traditional machine learning and deep learning methods to refine emotion classification in multilingual contexts, improving sentiment detection in mixed-language user-generated content. As research in this field progresses, the development of comprehensive datasets and advanced modeling techniques will be crucial in enhancing sentiment analysis for low-resource languages.

3 Proposed Methodology

This section describes the methodology followed for sentiment classification in code-mixed Tamil and Tulu texts using both Machine Learning (ML) and Deep Learning (DL) approaches. The dataset split for training and testing is shown in Table 1.

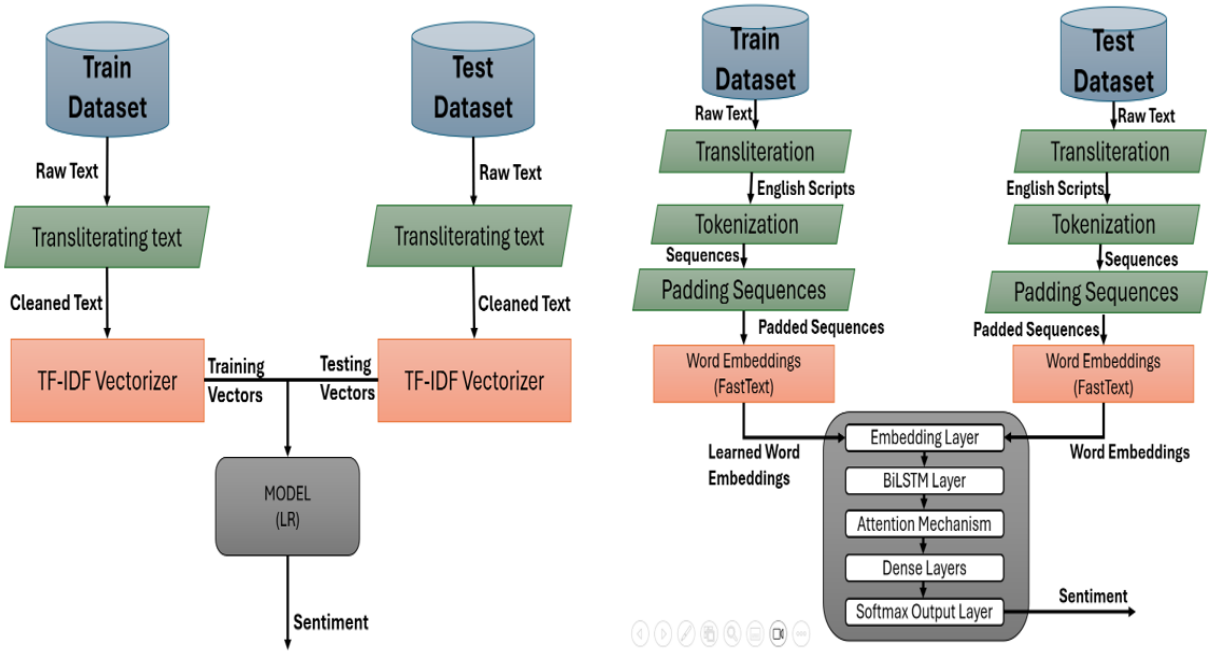
3.1 Preprocessing

Since the dataset contains code-mixed Tamil-English and Tulu-English text, transliteration is performed to convert all text into the English script for uniform processing. Following this, tokenization is applied to split text into individual words or subwords, enabling better feature extraction.

The text is then lowercased to maintain consistency. Additionally, punctuation marks and special characters are eliminated to reduce noise, and normalization techniques are used to handle repeated characters. Once preprocessed, the cleaned text is used as input for both ML and DL models, with different feature extraction techniques applied in each approach.

3.2 Machine Learning Approach

In the ML approach, numerical feature representations of text are generated using the TF-IDF (Term Frequency-Inverse Document Frequency) vectorizer, which captures the importance of words within the dataset. The TF-IDF vectorization is



(a) Machine Learning Approach

(b) Deep Learning Approach

Figure 1: Architecture diagram of the proposed methodology using (a) machine learning and (b) deep learning approaches.

Table 1: Data Distribution

Language	Training Set	Validation Set
Tamil	31122	3843
Tulu	13308	1643

performed separately on the training and testing datasets, using a character-level analyzer with an n-gram range of (1,4) to effectively capture word-level and subword-level patterns which is vital for code-mixed texts.

Once the text is converted into feature vectors, a Logistic Regression (LR) classifier is trained using these numerical representations. The model is trained with the 'liblinear' solver. The trained model is then used to predict sentiment labels for the test dataset. The performance is evaluated using accuracy and macro F1-score, with the given validation dataset. This approach relies on statistical feature extraction and performs effectively with relatively smaller datasets, making it well-suited for sentiment classification in code-mixed text. The overall architecture of the ML approach is shown in Figure 1a.

3.3 Deep Learning Approach

The DL approach leverages word embeddings and sequential modeling to better capture the contextual relationships within the text. The cleaned text is first tokenized and converted into sequences, which are then mapped to word embeddings using FastText, trained on the dataset to generate dense vector representations for words, including out-of-vocabulary words. The sequences are padded to 100 tokens. The embeddings are initialized with a dimension of 300 and a window size of 5.

These embeddings serve as input to a Bidirectional LSTM (BiLSTM) model with an attention mechanism, allowing the model to capture both forward and backward dependencies in the text. The BiLSTM layer consists of 128 hidden units and includes a dropout rate of 0.3 to prevent overfitting. The attention layer refines the model's focus on the most informative words, followed by fully connected dense layers with ReLU activation and L2 regularization. The final softmax layer classifies the text into sentiment categories. The model is trained using sparse categorical cross-entropy loss and optimized with the Adam optimizer, set with an initial learning rate of 0.001 with decay. The overall architecture of the Deep Learning approach is shown in Figure 1b.

Table 2: Performance analysis of the proposed system using validation data

Language	Model	Macro F1 Score	Accuracy
Tamil	Logistic Regression	0.46	0.54
Tamil	BiLSTM with Attention	0.43	0.58
Tulu	Logistic Regression	0.60	0.69
Tulu	BiLSTM with Attention	0.48	0.65

While the DL approach can capture deeper semantic relationships, its performance is dependent on the availability of large labeled datasets. In this study, the ML approach outperforms the DL model, particularly for the Tulu dataset, indicating that traditional feature-based methods may still be effective in low-resource settings where deep learning models struggle with limited training data. Additionally, the smaller dataset size may have led to suboptimal generalization in the deep learning model.

4 Results

The experimental results from Table 2 reveal that the ML model (TF-IDF + Logistic Regression) outperforms the DL model (FastText + BiLSTM + Attention) for both Tamil and Tulu datasets in terms of macro F1 scores. For Tamil, the ML model achieved a score of **0.46** compared to **0.43** for the DL model, while for Tulu, the ML model performed significantly better with a score of **0.60** compared to **0.48** for the DL model. The disparity in performance is more pronounced for Tulu, likely due to its smaller dataset size (13,308 samples) compared to Tamil (31,122 samples). These results highlight the effectiveness of ML models in handling limited data, whereas the DL model struggled due to its reliance on larger datasets for effective representation learning. The code used for these experiments is available on GitHub.[1]

5 Conclusions

The findings demonstrate that ML models are better suited for sentiment analysis of code-mixed texts, particularly in low-resource settings, as they effectively leverage n-gram-based features without requiring extensive labeled data. DL models, while theoretically capable of capturing richer contextual and semantic relationships, underperform with limited data availability. This is because deep learning models require a large amount of training data to learn meaningful representations and avoid overfitting, whereas smaller datasets fail to provide sufficient examples for learning complex patterns. This study emphasizes the need for larger and more diverse datasets to fully realize the potential of DL models for code-mixed text analysis and suggests exploring transfer learning or pre-trained multilingual models to improve performance in low-resource scenarios. Overall, ML models remain a practical and reliable approach for code-mixed sentiment analysis, especially for underrepresented languages like Tulu.

6 Limitations

This study is limited by the dataset size, particularly for Tulu ($13,308$ samples), which affects model generalizability. It focuses only on Tamil-English and Tulu-English code-mixed texts, limiting applicability to other Dravidian languages. The ML model uses TF-IDF, which lacks contextual understanding, while the DL model (FastText + BiLSTM + Attention) is not fine-tuned on Dravidian corpora. Transformer-based models like mBERT and XLM-R are not explored, and transliteration variations are not explicitly handled. Addressing these limitations in future work could improve sentiment classification in Dravidian code-mixed texts.

[1] https://github.com/JustATalentedGuy/JustATalentedTeam_NAACL

References

Nitin Nikamanth Appiah Balaji, B Bharathi, and J Bhuvana. 2020. Ssncse_nlp@ dravidian-codemix-fire2020: Sentiment analysis for dravidian languages in code-mixed text. In *FIRE (Working Notes)*, pages 554–559.

B Bharathi and GU Samyuktha. 2021. Machine learning based approach for sentiment analysis on multilingual code mixing text. In *FIRE (Working Notes)*, pages 1038–1043.

Bharathi Raja Chakravarthi, Vigneshwaran Muralidaran, Ruba Priyadharshini, and John P McCrae. 2020a. Corpus creation for sentiment analysis in code-mixed tamil-english text. *arXiv preprint arXiv:2006.00206*.

Bharathi Raja Chakravarthi, Ruba Priyadharshini, Vigneshwaran Muralidaran, Shardul Suryawanshi, Navya Jose, Elizabeth Sherly, and John P McCrae. 2020b. Overview of the track on sentiment analysis for dravidian languages in code-mixed text. In *Proceedings of the 12th Annual Meeting of the Forum for Information Retrieval Evaluation*, pages 21–24.

Thenmozhi Durairaj, Bharathi Raja Chakravarthi, Asha Hedge, Hosahalli Lakshmaiah Shashirekha, Rajeswari Natarajan, Sajeetha Thavareesan, Ratnasingam Sakuntharaj, Kalyanasundaram Krishnakumari, Charumathi Rajkumar, Poorvi Shetty, and Harshitha S Kumar. 2025. Overview of the Shared Task on Sentiment Analysis in Tamil and Tulu: DravidianLangTech@NAACL 2025. In *Proceedings of the Fifth Workshop on Speech, Vision, and Language Technologies for Dravidian Languages*. Association for Computational Linguistics.

Asha Hegde, Mudoor Devadas Anusha, Sharal Coelho, Hosahalli Lakshmaiah Shashirekha, and Bharathi Raja Chakravarthi. 2022. Corpus creation for sentiment analysis in code-mixed tulu text. In *Proceedings of the 1st Annual Meeting of the ELRA/ISCA Special Interest Group on Under-Resourced Languages*, pages 33–40.

Asha Hegde, Bharathi Raja Chakravarthi, Hosahalli Lakshmaiah Shashirekha, Rahul Ponnusamy, Subalalitha Cn, SK Lavanya, Durairaj Thenmozhi, Martha Karunakar, Shreya Shreeram, and Sarah Aymen. 2023. Findings of the shared task on sentiment analysis in tamil and tulu code-mixed text. In *Proceedings of the Third Workshop on Speech and Language Technologies for Dravidian Languages*, pages 64–71.

Selam Kanta and Grigori Sidorov. 2023. Selam@ dravidianlangtech: Sentiment analysis of code-mixed dravidian texts using svm classification. In *Proceedings of the Third Workshop on Speech and Language Technologies for Dravidian Languages*, pages 176–179.

Lavanya Sambath Kumar, Asha Hegde, Bharathi Raja Chakravarthi, Hosahalli Shashirekha, Rajeswari Natarajan, Sajeetha Thavareesan, Ratnasingam Sakuntharaj, Thenmozhi Durairaj, Prasanna Kumar Kumaresan, and Charmathi Rajkumar. 2024. Overview of second shared task on sentiment analysis in code-mixed tamil and tulu. In *Proceedings of the Fourth Workshop on Speech, Vision, and Language Technologies for Dravidian Languages*, pages 62–70.

Kishore Kumar Ponnusamy, Charmathi Rajkumar, Prasanna Kumar Kumaresan, Elizabeth Sherly, and Ruba Priyadharshini. 2023. Vel@ dravidianlangtech: Sentiment analysis of tamil and tulu. In *Proceedings of the Third Workshop on Speech and Language Technologies for Dravidian Languages*, pages 211–216.

Karthik Puranik et al. 2021. Iiitt@ dravidian-codemix-fire2021: Transliterate or translate? sentiment analysis of code-mixed text in dravidian languages. *arXiv preprint arXiv:2111.07906*.

K Rachana, M Prajnashree, Asha Hegde, and HL Shashirekha. 2023. Mucs@ dravidian-langtech2023: Sentiment analysis in code-mixed tamil and tulu texts using fasttext. In *Proceedings of the Third Workshop on Speech and Language Technologies for Dravidian Languages*, pages 258–265.

Josephine Varsha, B Bharathi, and A Meenakshi. 2022. Sentiment analysis and homophobia detection of youtube comments in code-mixed dravidian languages using machine learning and transformer models. In *FIRE (Working Notes)*, pages 124–137.