Synapse@DravidianLangTech 2025: Multiclass Political Sentiment Analysis in Tamil X (Twitter) Comments: Leveraging Feature Fusion of IndicBERTv2 and Lexical Representations

Suriya KP[1], Durai Singh K[1], Vishal A S[1], Kishor S[1], Sachin Kumar S[1]
[1]Amrita School of Artificial Intelligence, Coimbatore
Amrita Vishwa Vidyapeetham, India
{cb.en.u4aie22164, cb.en.u4aie22167, cb.en.u4aie22159, cb.en.u4aie22128}
@cb.students.amrita.edu , s_sachinkumar@cb.amrita.edu

Abstract

Social media platforms like X (Twitter) have gained popularity for political debates and election campaigns in the last decade. This creates the need to moderate and understand the sentiments of the tweets in order to understand the state of digital campaigns. This paper focuses on political sentiment classification of Tamil X (Twitter) comments which proves to be challenging because of the presence of informal expressions, code-switching, and limited annotated datasets. This study focuses on categorizing them into seven classes: substantiated, sarcastic, opinionated, positive, negative, neutral, and none of the above. This paper proposes a solution to Political Multiclass Sentiment Analysis of Tamil X (Twitter) Comments - DravidianLangTech@NAACL 2025 shared task, the solution incorporates IndicBERTv2-MLM-Back-Translation model and TF-IDF vectors into a custom model. Further we explore the use of preprocessing techniques to enrich hashtags and emojis with their context. Our approach achieved Rank 1 with a macro F1 average of 0.38 in the shared task.

Keywords: Political Comments, Tamil tweets, NLP, IndicBERTv2, TF-IDF, Agentic System, Sentiment Analysis.

1 Introduction

Sentiment analysis is a method of analyzing and interpreting feelings, attitudes, and opinions in text. Aspects of sentiment analysis are an integral part of surveys on public opinion, election-future predictions, and policymaking in terms of political discourse. Public forums such as X (Twitter), have emerged as an active platform where people express their opinions in real time, every day, providing a rich resource for analyses of this type (V P et al., 2023). The spontaneity and openness of social media make it a treasure trove to study public sentiments, especially in diverse lingual settings.

Tamil is a Dravidian language, which is rich in literary traditions and predominantly spoken in Tamil Nadu, India, and some parts of Sri Lanka, Singapore and Malaysia. However, culturally important, Tamil is a low-resource language in terms of NLP, lacking annotated datasets, computational tools, and language-specific resources. That marks significant challenges in Tamil sentiment analysis. The informal nature of online speech, slang usage, code-mixing with English (Sreelakshmi et al., 2024), and the frequent use of emojis, hashtags, and abbreviations demand careful linguistic interpretation to convey its complexity.

To tackle these challenges, the shared task "DravidianLangTech@NAACL 2025" (Chakravarthi et al., 2025) will focus on political multiclass sentiment analysis of Tamil X (Twitter) comments to analyse Tamil political discourse while being aware of its linguistic complexities. This paper explores the use of IndicBERTv2, which is trained in 23 Indian languages, for deep contextual understanding and TF-IDF for lexical feature extraction, thus improving the robustness of the model.

2 Related Work

In recent years, significant research has focused on political sentiment analysis, particularly in identifying and understanding public opinion and bias within social media. (B et al., 2024) employed TF-IDF with n-gram features in ensemble models for Tamil and Tulu sentiment analysis achieving F1 scores of 0.260 and 0.550, respectively. The authors demonstrated the capability of TF-IDF in feature extraction for code-mixed tweets providing deep learning representations.

(Kumar et al., 2017) employed a BiLSTM-CNN model for sentiment analysis in Malayalam, attaining an accuracy of 0.9824 has allowed more room for the introduction of transformers in low-resource languages. (Chakravarthi et al., 2021) obtained the

Proceedings of the Fifth Workshop on Speech, Vision, and Language Technologies for Dravidian Languages, pages 751–755
May 3, 2025 ©2025 Association for Computational Linguistics

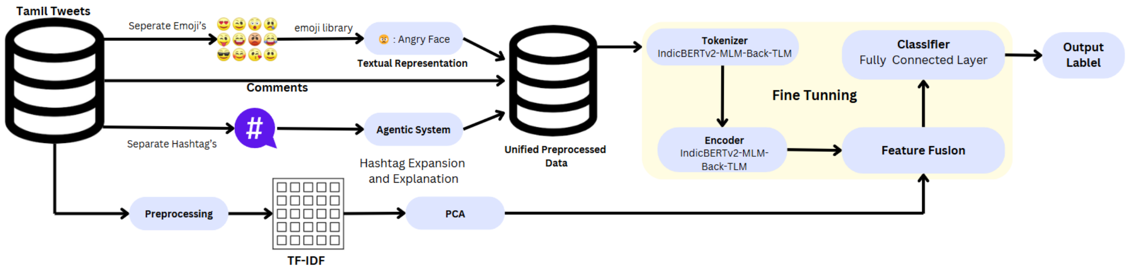

Figure 1: proposed workflow for Political Sentiment Analysis in Tamil X (Twitter) comments. The workflow comprising the modules of preprocessing, feature extraction, model training, and classification.

weighted F1 score of 0.711 in the Tamil-English sentiment analysis based on political psycholinguistic studies in multilingual settings. (Rajalakshmi et al., 2022) combined MuRIL with emoji-based sentiment analysis and proved that the introduction of emojis improved classification accuracy and said that multimodal features are essential in code-mixed sentiment analysis. Finally, (Angdresey et al., 2025) proposed a hybrid model based on tagging along with a BERT-based model, random oversampling, and Multinomial Naïve Bayes, achieving an accuracy of 85.155% and AUC of 96.80% in political sentiment analysis of YouTube comments.

3 Dataset

The dataset has been annotated for political sentiments in Tamil X (Twitter) comments and included comments, which were annotated into seven categories: Substantiated, Sarcastic, Opinionated, Positive, Negative, Neutral, and None of the above.

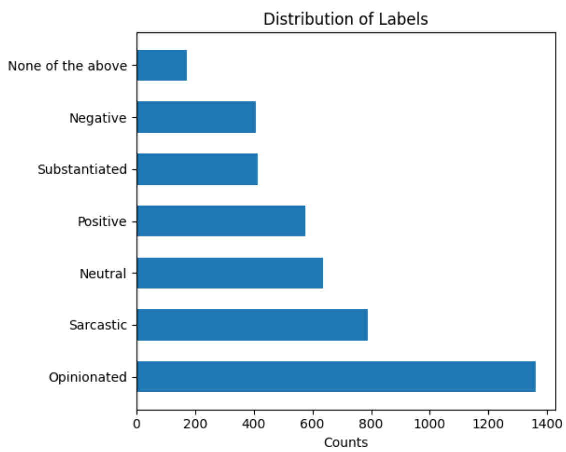

Figure 2: Class distribution in the training dataset.

In total, the dataset has 5440 comments which is divided into three subsets: 80% samples for training with a count of 4,352 samples, 10% for development with 544 samples, and 10% for testing with 544 samples. As organized, the dataset is pre-stratified by the organizers for the sake of uniformity among splits, but Figure 2 depicts the imbalance distribution of different classes in training data.

To balance this, class weights were calculated as the inverse of the class frequencies. These weights were then added to the model's training process to ensure a more balanced learning approach, allowing the model to handle underrepresented categories effectively.

4 Methodology

4.1 Pre-processing

Our pre-processing pipeline has been designed considering the informal and code-mixed nature of Tamil political tweets to ensure that there is meaningful text representation for sentiment classification. The following are the steps it contains:

Demojize: Emojis are converted into textual descriptions using the python emoji library. This step helps retain the emotions expressed through emojis, improving sentiment classification accuracy.

Agentic System: The system uses a two-step process: context retrieval and contextual generation. The most commonly used 160 hashtags were filtered, taking relevant hashtags. For each hashtag, the system queries the Serper API for three top-ranked results and captures contextually relevant text snippets to be used as context. Then, with the help of LLaMA 3.1, it forms a short one-line description depending on pre-existing rules, such as Tamil abbreviations being expended, political actors being identified, and parties being tagged. Such a method applies precise, context-sensitive hashtag explanations, enhancing the sentiment analysis for Tamil political debates.

Stopword Removal: Commonly occurring but non-informative words are removed to improve the effectiveness of feature extraction. This will be used for Term Frequency-Inverse Document Frequency (TF-IDF) representation, as it focuses on meaningful words that contribute to sentiment analysis.

4.2 Feature Extraction

After pre-processing, feature extraction techniques are employed to convert the text into some numerical representation suitable for training. The main techniques used here are:

Term Frequency-Inverse Document Frequency: TF-IDF assigns importance to the words in a document based on their frequency while reducing the weight of frequent words that appear in many documents. It improves feature representation in text classification by highlighting distinctive terms. Higher values indicate more informative words.

$$\text{TF-IDF}(txt, doc) = \text{TF}(txt, doc) \times \log\left(\frac{N'}{\text{DF}(txt)}\right) \tag{1}$$

Where N' denotes total number of document. and $\text{DF}(txt)$ denotes number of documents that contain the term txt.

TF-IDF was employed to enrich IndicBERTv2 by emphasizing important words that transformers may not catch, enhancing interpretability, capturing domain-specific phrases

Principal Component Analysis(PCA):Reduces the dimension of the data while preserving essential variance, making it useful for noise reduction and visualization.

$$X' = XW \tag{2}$$

where W consists of eigenvectors of the covariance matrix of X, capturing the most significant variance in the dataset.

PCA was applied for dimensionality reduction, noise removal, and preserving vital variance to ensure efficient and concise feature representation

5 Model and Training

In our model, we propose a hybrid approach, combining deep contextual embeddings from **IndicBERTv2-MLM-Back-Translation** (Doddapaneni et al., 2023) with features obtained from **TF-IDF** vectorization (S N et al., 2022), allowing the model to exploit both semantic knowledge and frequency-based linguistic patterns to enhance classification performance. The entire architecture of the model is shown in the figure 3.

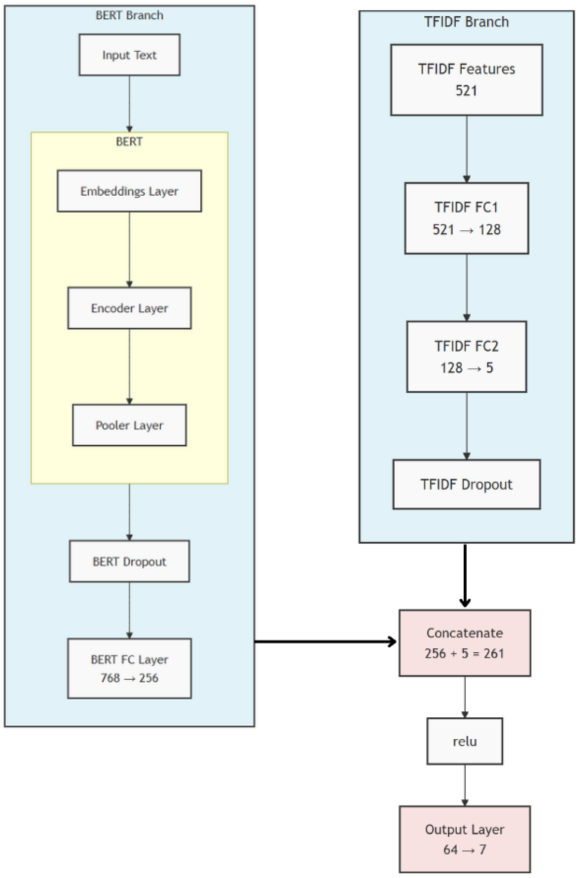

Figure 3: Architecture of the IndicBERTv2-based classification model used in the competition.

There are two main components of the system: a transformer-based encoder and a statistical feature-based discriminator. Preprocessed comments are passed to IndicBERTv2 model to process input text and extract contextual features which get concatenated with the TF-IDF features. These features are then passed through a fully connected classification network with dropout regularization to control overfitting. Finally, a softmax layer predicts one from among the seven sentiment categories.

The training setup has several optimization steps that aim to improve model generalization. We have used the **AdamW** optimizer with the learning rate set at 2e-5 with **weighted CrossEntropyLoss**, taking into account class imbalance by weighing according to inverse class frequencies. To have stable learning, we used **incremental batch sizes**. That is, the batch size is initialized to 16 for the first two epochs and then increased to 32, 48, and finally 64 for the later epochs. Seeded shuffling of dataset is performed before every epoch to maintain variance in the distribution of classes throughout all passes.

The model is fine-tuned for eight epochs, allowing both representations to move into more nuanced representations of Tamil's political discourse. To control overfitting, dropout regularization is applied with a regularization probability of 0.1 for both models' feature sets, i.e., the transformer and TF-IDF.

6 Result And Analysis

6.1 Macro Average F1-Score

We have used the F1 macro average, as required in the task, which computes the harmonic mean of precision and recall for each class and then averages across all classes, which is appropriate for handling imbalanced datasets by giving equal treatment to all classes.

$$F1_{\text{macro}} = \frac{1}{C} \sum_{i=1}^{C} \frac{2 \times P_i \times R_i}{P_i + R_i} \qquad (3)$$

Where C denotes Total number of classes and R_i and P_i denotes precision and recall of class i respectively.

This metric ensures fair performance evaluation, even when some classes are underrepresented.

6.2 Results

We conducted experiment on four different IndicBERT-V2 models as our base encoder, the results are provided in Table 1

Model	Macro-F1
IndicBERTv2-MLM-Back-TLM	0.383
IndicBERTv2-MLM-Sam-TLM	0.376
IndicBERTv2-SS	0.338
IndicBERTv2-MLM-only	0.290

Table 1: Performance of IndicBERT-V2 model variants.

The IndicBERTv2-MLM-Back-TLM model performed the best, while the same base model without TF-IDF features scored a macro-F1 score of 0.362, implying the significance of TF-IDF features.

The class-wise precision, recall, and F1 score of our final model are presented in Table 2

The findings indicate that the IndicBERTv2-MLM-Back-TLM model was excellent in classifying the 'None' class which is easier to distinguish, with precision, recall, and F1 score of 1.00, 0.92, and 0.96, respectively. It did struggle with the 'negative' and 'substantiated' classes, which exhibit greater linguistic complexity.

Class	Precision	Recall	F1-score
Negative	0.13	0.15	0.14
Neutral	0.17	0.27	0.21
None	1.00	0.92	0.96
Opinionated	0.53	0.33	0.40
Positive	0.25	0.35	0.29
Sarcastic	0.50	0.42	0.45
Substantiated	0.22	0.24	0.23

Table 2: Class-wise performance in terms of Precision, Recall, and F1-score of our model.

Our final system, based on the IndicBERTv2-MLM-Back-TLM model, achieved a macro-F1 score of 0.3773, securing the 1st rank in the shared task. Table 3 shows the top 4 teams.

Project code files are available in Github.[1]

Rank	Team Name	Macro-F1
1	**Synapse**	**0.3773**
2	KCRL	0.3710
3	byteSizedLLM	0.3497
4	Eureka-CIOL	0.3187

Table 3: Top 4 teams in the shared task.

7 Conclusion

This paper studies political sentiment analysis on Tamil X (Twitter) comments using our custom model, which has achieved Rank 1 in this shared task. Our approach incorporates IndicBert-V2-MLM-Back-Translation and TF-IDF features, this addresses the challenges of informal, code-mixed discourse through pre-processing techniques such as demojizing and agent based hashtag expansion. Our approach can effectively capture linguistic nuances, making it suitably applicable to multi-class sentiment classification in political discussions. The study demonstrates the potential of NLP in analyzing public opinion in Tamil-speaking online communities and contributes to advancing sentiment analysis for Tamil language.

[1] https://github.com/SURIYA-KP/NAACL_DravidianLangTech_2025/

8 Limitations

Despite its effectiveness, our solution has certain limitations. The dataset used for training is relatively small, and its inherent ambiguity poses challenges, even for human interpretation. For instance, the query in tamil translating to "Naam Tamilar Vs AMMK.. Who will lead in Trichy?" is labelled as negative, despite its semantic proximity to a neutral, interrogative stance, thus limiting the model's ability to generalize across diverse political contexts. Highly ambiguous or sarcastic statements are also challenges for the model, in which the sentiment is hard to determine without more profound contextual understanding. Although IndicBERTv2 performs very well, even larger pretrained models with greater computational power could refine the accuracy of sentiment classification. Future research might address these aspects by using larger annotated datasets and further advanced transformer-based architectures.

References

A. Angdresey, L. Sitanayah, and I. L. H. Tangka. 2025. Sentiment analysis for political debates on youtube comments using bert labeling, random oversampling, and multinomial naïve bayes. *Journal of Computing Theories and Applications*, 2(3):342–354.

Prathvi B, Manavi K, Subrahmanyapoojary K, Asha Hegde, Kavya G, and Hosahalli Shashirekha. 2024. MUCS@DravidianLangTech-2024: A grid search approach to explore sentiment analysis in code-mixed Tamil and Tulu. In *Proceedings of the Fourth Workshop on Speech, Vision, and Language Technologies for Dravidian Languages*, pages 257–261, St. Julian's, Malta. Association for Computational Linguistics.

B. R. Chakravarthi, R. Priyadharshini, S. Thavareesan, D. Chinnappa, D. Thenmozhi, E. Sherly, and C. Vasantharajan. 2021. Findings of the sentiment analysis of dravidian languages in code-mixed text. *arXiv preprint arXiv:2111.09811*.

Bharathi Raja Chakravarthi, Saranya Rajiakodi, Elizabeth Sherly, Thenmozhi Durairaj, Sathiyaraj Thangasamy, Ratnasingam Sakuntharaj, Prasanna Kumar Kumaresan, Kishore Kumar Ponnusamy, Arunaggiri Pandian Karunanidhi, and Rohan R. 2025. Overview of the Shared Task on Political Multiclass Sentiment Analysis of Tamil X(Twitter) Comments: DravidianLangTech@NAACL 2025. In *Proceedings of the Fifth Workshop on Speech, Vision, and Language Technologies for Dravidian Languages*. Association for Computational Linguistics.

Sumanth Doddapaneni, Rahul Aralikatte, Gowtham Ramesh, Shreya Goyal, Mitesh M. Khapra, Anoop Kunchukuttan, and Pratyush Kumar. 2023. Towards leaving no indic language behind: Building monolingual corpora, benchmark and models for indic languages. *Preprint*, arXiv:2212.05409.

S Sachin Kumar, M Anand Kumar, and KP Soman. 2017. Sentiment analysis of tweets in malayalam using long short-term memory units and convolutional neural nets. In *Mining Intelligence and Knowledge Exploration: 5th International Conference, MIKE 2017, Hyderabad, India, December 13–15, 2017, Proceedings 5*, pages 320–334. Springer.

R. Rajalakshmi, S. Selvaraj, A. Shibani, and B. R. Chakravarthi. 2022. Understanding the role of emojis for emotion detection in tamil. In *Proceedings of the First Workshop on Multimodal Machine Learning in Low-resource Languages*, pages 9–17.

Prasanth S N, R Aswin Raj, Adhithan P, Premjith B, and Soman Kp. 2022. CEN-Tamil@DravidianLangTech-ACL2022: Abusive comment detection in Tamil using TF-IDF and random kitchen sink algorithm. In *Proceedings of the Second Workshop on Speech and Language Technologies for Dravidian Languages*, pages 70–74, Dublin, Ireland. Association for Computational Linguistics.

K. Sreelakshmi, B. Premjith, Bharathi Raja Chakravarthi, and K. P. Soman. 2024. Detection of hate speech and offensive language codemix text in dravidian languages using cost-sensitive learning approach. *IEEE Access*, 12:20064–20090.

Abeera V P, Dr. Sachin Kumar, and Dr. Soman K P. 2023. Social media data analysis for Malayalam YouTube comments: Sentiment analysis and emotion detection using ML and DL models. In *Proceedings of the Third Workshop on Speech and Language Technologies for Dravidian Languages*, pages 43–51, Varna, Bulgaria. INCOMA Ltd., Shoumen, Bulgaria.