KSK@DravidianLangTech 2025: Political Multiclass Sentiment Analysis of Tamil X (Twitter) Comments Using Incremental Learning

Kalaivani K S[1], **Sanjay R**[1], **Thissyakkanna S M**[1], **Nirenjhanram S K**[1]
[1]Department of AI, Kongu Engineering College, Perundurai, Erode.
{kalaivani.cse, sanjayr.22aid}@kongu.edu
{thissyakkannasm.22aid, nirenjhanramsk.22aid}@kongu.edu

Abstract

The introduction of Jio in India has significantly increased the number of social media users, particularly on platforms like X (Twitter), Facebook, Instagram. While this growth is positive, it has also led to a rise in native language speakers, making social media analysis more complex. We took part in the shared task to classify political comments to classify social media comments from X (Twitter) into seven different categories. Tamil speaking users often communicate using a mix of Tamil and English, creating unique challenges for analysis and tracking. This surge in diverse language usage on social media highlights the need for robust sentiment analysis tools to ensure the platform remains accessible and user-friendly for everyone with different political opinions. In this study we trained four machine learning models, SGD Classifier, Random Forest Classifier, Decision Tree, and Multinomial Naive Bayes classifier to identify and classify the comments. Among these, the SGD Classifier achieved the best performance, with a training accuracy of 83.67% and a validation accuracy of 80.43%.

1 Introduction

In August 2024, India accounted for approximately 462 million active social media accounts, representing 32.2% of its population. This number is projected to grow exponentially, reaching nearly 1.2 billion users by 2029. With such massive growth, a significant portion of communication on these platforms is made by code-mixed language, an combination of native languages and English. Identifying politically inappropriate and sentimentally abusive comments within this data presents a unique and complex challenge due to the linguistic diversity and informal writing styles used by the users (Palit and Pal, 2018; Priyadharshini et al., 2021). Social media platforms possess immense power to influence public opinion in a short period, making it critical to monitor and classify political comments to ensure a safe space for users.(Kumar et al., 2021). Manual identification of such comments is not possible due to the volume of data and the complexity introduced by the surge of users and code-mixed language, which deviates substantially from standardized linguistic rules chakravarthi2020corpus. Advancements in machine learning (ML) and deep learning (DL) have enabled significant progress in NLP, particularly in analyzing multilingual and code-mixed datasets (Bojanowski et al., 2017a; Devlin et al., 2018; Chakravarthi, 2022). These technologies offer scalable, efficient solutions for text classification tasks. However, static models often struggle to adapt to the evolving nature of social media content, where trends, language usage, and sentiment expressions continuously change. To address this we have explored incremental learning to train the model. Incremental learning enables models to learn and adapt to new data without requiring complete retraining, making it highly effective for real-time applications like social media sentiment analysis (Parisi et al., 2019; Loshchilov and Hutter, 2017). In this study, we explore machine learning models for the multiclass sentiment analysis of Tamil X (formerly Twitter) comments. We categorize comments into seven sentiment classes: negative, neutral, none of the above, opinionated, positive, sarcastic, and substantiated. Our use incremental learning to efficiently adapt models to new data while maintaining performance on previously learned tasks. Various machine learning algorithms like Stochastic Gradient Descent (SGD) Classifier, Random Forest Classifier, Decision Tree, and Multinomial Naive Bayes Classifier are used. Out of 25 teams, our team ranked 18th place in the shared task(Chakravarthi et al., 2025).

2 Related Works

The study of sentiment analysis, particularly in the context of code-mixed text, has gained consider-

Proceedings of the Fifth Workshop on Speech, Vision, and Language Technologies for Dravidian Languages, pages 308–312

able attention in recent years due to the increasing prevalence of multilingual communication on social media platforms. Several researchers have explored various techniques to address the challenges of code-mixed data and sentiment classification.

(Khanuja et al., 2020) proposed a method for detecting offensive language in Hinglish (Hindi-English) code-mixed text using deep learning approaches. They highlighted the linguistic diversity and informal nature of code-mixed text as primary challenges, emphasizing the need for domain-specific embeddings.

ramesh2021dravidian focused on leveraging pre-trained multilingual language models like mBERT and XLM-R for sentiment analysis on Dravidian languages. Their experiments demonstrated the effectiveness of transfer learning for handling code-mixed languages while identifying areas for improvement in fine-tuning strategies

joshi2020lowresource explored sentiment classification in low-resource languages by combining rule-based and machine learning methods. They addressed the difficulties associated with data sparsity and introduced hybrid approaches that improved performance on small datasets

(Das et al., 2021) investigated the use of graph neural networks (GNNs) for sentiment analysis in code-mixed text. Their approach effectively captured relationships between words in multilingual sentences, offering a promising solution for sentiment classification tasks in linguistically diverse datasets.

(Ruder et al., 2019) provided a comprehensive survey on cross-lingual embeddings, discussing their applications for sentiment analysis and multilingual NLP tasks. They underscored the importance of shared embedding spaces for improving classification accuracy in code-mixed and low-resource scenarios.

Incremental learning techniques have also been explored in sentiment analysis to adapt to evolving data. (Bojanowski et al., 2017b) proposed continual learning algorithms designed to handle sequential tasks without catastrophic forgetting, making them highly suitable for real-time applications such as monitoring social media trends.Similarly,(Chen and Liu, 2018)introduced lifelong learning frameworks for text classification tasks, emphasizing the ability of models to accumulate knowledge across tasks.

These studies collectively highlight the advancements in machine learning and deep learning approaches for sentiment analysis on code-mixed text, while also emphasizing the potential of incremental learning to address dynamic and large-scale social media data. Building upon these works, this study adopts a combination of traditional classifiers and incremental learning techniques to classify Tamil X (formerly Twitter) comments into multiple sentiment categories.

3 Methodology

In this study machine learning is used to classify the training data into seven classes. This section discusses about various machine learning model used and procedures used in this study.

3.1 Dataset Used

The study uses the dataset provided by DravidianLangTech on Social Media sentiment classification (Chakravarthi et al., 2025). The dataset contains seven classses of different sentiment namely negative, neutral, none of the above, opinionated, positive, sarcastic, and substantiated. The training dataset contains 4533 rows of code-mixed tamil and the validation dataset contains 544 rows.

Label	Count
Opinionated	1,361
Sarcastic	790
Neutral	637
Positive	575
Substantiated	412
Negative	406
None of the above	171

Table 1: Training Data

Label	Count
Opinionated	153
Sarcastic	115
Neutral	84
Positive	69
Substantiated	52
Negative	51
None of the above	20

Table 2: Validation Data

3.2 Preprocessing Techniques

3.2.1 Removal of Hashtags, URLs, and Mentions:

Social media text often contains hashtags, URLs, and user mentions that do not offer any context and relevant meaning. The special character's such as '#' were removed and words such as 'www','.com' were removed keeping the content of hashtag and the URL's to train the model.

3.2.2 Whitespace Normalization:

Extra spaces which are present in the dataset were removed to ensure uniformity among the dataset.

3.2.3 Tokenization:

The cleaned up text is then split into words based on the occurence of space in between characters.

3.3 Models Used

In this study, we employed four machine learning algorithms to classify code-mixed Tamil text into seven sentiment categories: negative, neutral, none of the above, opinionated, positive, sarcastic, and substantiated. The models utilized are as follows:

3.3.1 Stochastic Gradient Descent (SGD) Classifier:

SGD is an optimization method that updates model parameters incrementally after evaluating each training example. This approach makes it efficient for large datasets and is particularly useful when a quick, approximate solution is acceptable.

3.3.2 Random Forest Classifier:

Random Forest is an ensemble learning method that builds multiple decision trees and combines their results to improve accuracy. Each tree in the forest makes a prediction, and the final output is determined by the majority vote among all trees. This method is effective for handling complex datasets with many features and is less prone to overfitting compared to individual decision trees.

3.3.3 Decision Tree Classifier:

A decision tree is a model that makes decisions by splitting data into subsets based on feature values, resembling a tree structure. It recursively splits the data at each node based on the feature that results in the best separation of classes. Decision trees are easy to understand and interpret, making them useful for problems where model transparency is important.

3.3.4 Multinomial Naive Bayes Classifier:

This probabilistic model is based on Bayes' theorem, assuming that features are conditionally independent given the class label. It calculates the probability of each class given the features and selects the class with the highest probability. Naive Bayes is particularly effective for text classification tasks, especially when the features (like words) are independent.

3.4 Training Methodology

The training data was divided into chunks of 108 batches, and the model was trained incrementally on these batches. This approach allows the model to learn from new data progressively, maintaining and building upon previous knowledge, which is particularly beneficial when dealing with large-scale datasets such as the one used in this study. Unlike traditional batch training, where the model is retrained from scratch with the entire dataset, incremental learning enables the model to update itself continuously without forgetting previously learned information. This is especially crucial in sentiment analysis, where language usage, expressions, and context evolve over time. By training in smaller chunks, the model adapts dynamically to emerging linguistic patterns and sentiment variations while preserving performance on earlier learned data. Additionally, this method reduces computational overhead, making real-time sentiment classification more efficient. Accuracy is monitored at each stage, ensuring that the model remains stable and effectively integrates new insights without suffering from catastrophic forgetting. Through continual updates, the model achieves improved generalization, making it well-suited for analyzing the ever-changing landscape of Tamil social media discourse.(dra, 2024)

4 Results and Discussion

Model performance can be assessed using a variety of metrics. In this study, we have chosen accuracy, precision, recall, and F1-score to evaluate the models. The machine learning models implemented include Stochastic Gradient Descent (SGD) Classifier, Random Forest Classifier, Decision Tree, and Multinomial Naive Bayes Classifier. The dataset is split into 100 chunks, and the models are trained in batches iteratively using incremental learning.

From Table 3, it is observed that the Stochastic Gradient Descent (SGD) Classifier achieves the

Model	Accuracy (%)	Validation Accuracy (%)
SGD Classifier	83.67	80.43
Naive Bayes	78.78	65.23
Logistic Regression	80.82	77.89
Random Forest Classifier	82.56	70.21

Table 3: Accuracy of different models on training and validation data

highest training accuracy of 83.67% and a validation accuracy of 80.43%, indicating strong generalization to unseen data. In contrast, the Naive Bayes model, with a validation accuracy of 65.23%, struggles to generalize, likely due to its assumption of feature independence, which does not hold in real-world linguistic data. The Logistic Regression model maintains balanced performance with 80.82% training accuracy and 77.89% validation accuracy, making it a stable alternative. The Random Forest Classifier, while achieving 82.56% training accuracy, exhibits a notable drop in validation accuracy (70.21%), suggesting overfitting due to its reliance on multiple decision trees.

While accuracy is an important evaluation metric, it does not fully capture the model's performance, particularly in imbalanced sentiment classes. Precision, recall, and F1-score provide deeper insights, revealing that Naive Bayes and Random Forest tend to misclassify minority sentiment categories, leading to lower recall scores. The use of incremental learning, where the dataset is processed in batches, enables the model to adapt to new linguistic trends in Tamil political discussions without catastrophic forgetting. This approach ensures the SGD Classifier maintains performance over time, making it well-suited for real-time sentiment analysis. Future improvements could explore weighted loss functions, data augmentation techniques, or deep learning-based models such as BERT or RoBERTa to enhance classification effectiveness in Tamil code-mixed sentiment analysis.

5 Conclusion

Sentiment analysis on code-mixed political data was conducted, and it was found that the Stochastic Gradient Descent (SGD) Classifier outperforms the other models by achieving a training accuracy of 83.67% and a validation accuracy of 80.43%. These results underscore the potential of SGD in handling code-mixed text in political sentiment analysis. In future work, further exploration can be carried out by incorporating more advanced techniques such as deep learning models, fine-tuning pre-trained models like BERT or RoBERTa for code-mixed data, and expanding the dataset to include more diverse political contexts to improve the model's robustness and performance. The code for our models and preprocessing methods is available here.

References

2024. *Proceedings of the EACL 2024 Workshop on Speech and Language Technologies for Dravidian Languages*.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017a. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017b. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Bharathi Raja Chakravarthi. 2022. Hope speech detection in youtube comments. *Social Network Analysis and Mining*, 12(1):75.

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.

Zhiyuan Chen and Bing Liu. 2018. Lifelong learning for sentiment analysis tasks. *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2204–2210.

S. Das et al. 2021. Graph neural networks for sentiment analysis in multilingual code-mixed text. *Knowledge-Based Systems*, 220:106901.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186. Association for Computational Linguistics.

Simran Khanuja, Kaustav Dey, El Moatez Billah Karim Nagoudi, et al. 2020. A new dataset and strong baselines for the detection of code-mixed offensive language. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1719–1726.

Manish Kumar, Vikas Chauhan, Ashok Kumar Yadav, and Yogesh Kumar Meena. 2021. Multilingual sentiment analysis on social media: Challenges and applications. *Information Processing & Management*, 58(4):102509.

Ilya Loshchilov and Frank Hutter. 2017. Sgdr: Stochastic gradient descent with warm restarts. In *Proceedings of the 5th International Conference on Learning Representations (ICLR)*.

N. Palit and K. Pal. 2018. Social media analytics: A survey on concepts, tools, and applications. *IEEE Access*, 6:12321–12345.

German I. Parisi, Ronald Kemker, Jose L. Part, Christopher Kanan, and Stefan Wermter. 2019. Continual learning in deep neural networks: An empirical model. *Neural Networks*, 113:54–71.

Ruba Priyadharshini, Bharathi Raja Chakravarthi, Elizabeth Sherly, and John P. McCrae. 2021. Sentiment analysis in tamil-english code-mixed social media text. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 20(1):1–21.

Sebastian Ruder, Ivan Vulić, and Anders Søgaard. 2019. A survey of cross-lingual embeddings and their applications. *Journal of Artificial Intelligence Research*, 65:569–631.