Cureus Journal of Computer Science

Open Access Original Article

Multilingual Transformer Contextual Embedding Model for Political Tweets Analysis

Vaibhav Khatavkar [1] , Sneha Petkar [2] , Archana S. Vaidya [3]

1. Computer Engineering, DES Pune University, Pune, IND 2. Computer Science, Gokhale Education Society's R. H. Sapat College of Engineering, Management Studies and Research, Nashik, IND 3. Computer Engineering, Gokhale Education Society's R. H. Sapat College of Engineering, Management Studies and Research, Nashik, IND

Corresponding author: Vaibhav Khatavkar, khatavkarvaibhav86@gmail.com

Received 02/06/2025
Review began 02/16/2025
Review ended 02/25/2025
Published 03/07/2025

DOI:
https://doi.org/10.7759/s44389-025-03177-4

Abstract

In India, the rise of social media usage has led to significant political discourse on platforms such as Twitter (now known as X), creating new opportunities for political sentiment analysis. With these new opportunities, still the challenges such as multilingualism, code-switching, sarcasm, and regional dialects limit the effectiveness of sentiment analysis methods. In order to address these challenges, this paper introduces a novel Multilingual Transformer Contextual Embedding Model to handle the complexities of Indian political tweets. We train and evaluate the model on a dataset of 3 million political tweets in Hindi, English, and regional languages. The results demonstrate an accuracy of 98.5%, surpassing existing state-of-the-art models by an average of 2.5%. Additionally, the introduced model achieved an F1 score of 97.5%, precision of 98%, and recall of 97%, showcasing its superiority in handling multilingual and context-specific nuances. Comparative analysis indicates that transformer-based models like the proposed model significantly outperform traditional machine learning models in sentiment classification and bias detection. The proposed model can be enhanced with the real-time processing capabilities, more multilingual support, and ethical considerations for more inclusive and unbiased political sentiment analysis frameworks.

Categories: Data Analysis, Data Engineering, Deep Learning
Keywords: political sentiment analysis, multilingual transformer, indian political tweets, natural language processing (nlp), emotion detection, code-switching, deep convolutional neural network

Introduction

Social media platforms like Twitter (now known as X) have transformed the political discourse and public opinion analysis during elections in developed and developing countries. In developing countries like India, the social media discourse analysis is a big challenge, because in India, multiple cultures, communities, histories, and languages coexist. There are average monthly 100 million Twitter active users, which makes the Twitter a crucial platform for political discussions, policy debates, and sentiment expressions. The Twitter discourse analysis offers valuable insights into public reactions to political events, leadership, and government policies [1]. These insights provide understanding of the socio-political climate, tracking shifts in political leanings, and help in anticipating trends that can impact national elections. However, the Twitter discourse analysis involves complexity due to multilingual expressions, code-switching (the blending of languages within a single tweet), slang, sarcasm, and regional dialects. Ambiguity arises from the use of slang and sarcasm, while bias and polarization often dominate politically charged discussions. For example, the tweet "Can't believe the latest speech...totally ridiculous! #FakeNews #Election2024 " reflects sarcasm, ambiguity, and political polarization, making sentiment classification difficult. Additionally, the dynamic nature of language on social media, with constantly evolving trends and expressions, further complicates analysis. A tweet like "Voting for a better tomorrow! #Election2024 #Hope" expresses sentiment and emotion but can still be influenced by bias and political leanings, adding another layer of complexity. The conventional natural language processing model for discourse analysis [2] faces challenges to address these complexities. In addition to these complexities, the massive volume and rapid generation of data during political events, noisy, and spam data from bots or irrelevant contents in data are other challenges for discourse analysis.

The discourse analysis for political tweets has evolved from classical machine learning algorithms to more sophisticated deep learning methods, including long short-term memory (LSTM), attention mechanisms, and transformers. Various researchers have applied these techniques to various political datasets [3]. For instance, Choudrie et al. [4] used SentiWordNet, naive Bayes, and hidden Markov models for sentiment analysis on Indian political tweets, employing word sense disambiguation to enhance accuracy. Additionally, Matalon et al. [5] employed machine learning techniques to detect political leniency using sentiment analysis on 1.72 million tweets, achieving remarkable accuracy of up to 99%. However, significant challenges persist in processing multilingual, context-dependent tweets from Indian users, particularly in the political domain. Tweets related to Indian politics frequently contain text in a mix of languages, such as Hindi, English, and various regional dialects, necessitating language models that can

effectively manage code-switching. These tweets often incorporate sarcasm and indirect references, complicating sentiment classification. Additionally, the emotional tone in such tweets often reflects anger, frustration, sarcasm, or approval, further challenging accurate sentiment detection.

Early methods focused on simpler classification models that were effective but limited in their ability to capture complex, context-sensitive features inherent in multilingual political data. This evolution mirrors the growing complexity and scale of social media data, particularly on platforms like Twitter. Patel et al. [6] applied sentiment analysis in various domains, including a customer feedback system for the airline industry, demonstrating the versatility of sentiment analysis tools. Similarly, Gunawan et al. [7] applied sentiment analysis to study daily school activities post-COVID-19, while Fransiscus and Girsang [8] explored sentiment analysis on public activities during the pandemic. These studies underscore the broad applicability of sentiment analysis beyond politics, emphasizing its role in capturing public opinion in different contexts.

In addition to the technical advancements, analyzing political data on social media platforms like Twitter offers crucial insights into public opinion, voting trends, and political campaigns. Machine learning techniques such as classification, clustering, sentiment analysis, and topic modeling have proven effective in these domains. Deep learning approaches, particularly transformer-based models, have further enhanced the ability to process textual data from social media.

In the context of Twitter, location prediction has been another avenue of exploration, as seen in the work that focus on predicting user locations based on tweet content, showcasing the power of social media data for geographical analysis [9]. Similarly, some researchers developed an aspect-oriented system for analyzing restaurant reviews using bidirectional encoder representations from transformers (BERT) [10], while some employed IndoBERT model for customer review analysis [11]. These studies demonstrate the growing importance of transformer-based models like BERT in extracting nuanced insights from unstructured textual data. Focusing specifically on political sentiment, Geni et al. [12] applied the IndoBERT transformer model to Twitter data from the pre-2024 Indonesian elections to gauge public opinion. Their findings revealed that IndoBERT outperformed traditional machine learning models and lexicon-based approaches. IndoBERT achieved an accuracy of 83.5%, significantly surpassing Textblob by 48.5%, multinomial naive Bayes by 2.5%, and support vector machine (SVM) by 14.49%. This highlights the efficacy of transformer-based models in sentiment analysis, particularly for complex and multilingual datasets. In a broader political context, Pacheco [13] analyzed 437 million Brazilian political tweets from 13 million users between 2018 and 2023. The study identified patterns of bot engagement, particularly during the COVID-19 pandemic and the aftermath of Brazil's 2022 presidential elections, shedding light on the role of automated accounts in political discourse. These insights contribute to the growing understanding of how social media platforms are leveraged for political manipulation and opinion shaping. In India, researchers have made significant strides in analyzing political tweets. Jafri et al. [14] employed an ensemble oversampling method to detect hate speech and target groups (individuals, communities, and organizations) in Hindi tweets from Indian political datasets. Their work highlights the importance of addressing hate speech in political discussions, a growing concern in Indian social media. Another notable study by Sukanya and Sita [15] analyzed multilingual tweets, focusing on "Hinglish" (a mix of Hindi and English), for sentiment analysis. Their work emphasized the difficulties of handling code-mixed language, which includes slang, abbreviations, and emojis. Despite these challenges, the researchers achieved an accuracy of 77% and an F1 score of 72%. The relatively lower performance was attributed to the limitations of current natural language processing tools in processing Hinglish, especially in stem-word analysis. In terms of political sentiment analysis within Indian elections, Patel et al. [16] used VADER sentiment analysis in conjunction with random forest and decision tree algorithms to predict the outcomes of the Gujarat assembly elections 2022. Their model achieved accuracies of 89% and 86%, respectively. Similarly, Pathan and Sundar [17] applied random forest to Indian political tweets, achieving accuracies of 76.3% and 77.6%, respectively. Decision tree models were also employed by them, who reported an accuracy of 68.39%, while the model developed by Reddy [18] for sentiment analysis in the Karnataka state elections delivered a 72% accuracy. Sharma and Kumar [19] have also applied SentiWordNet, naive Bayes, and hidden-Markov model for Indian political tweets sentiment analysis. Furthermore, they applied word sense disambiguation to increase the accuracy of the system. Sai Kowsik et al. [20] detected political leniency using sentiment analysis and machine learning techniques on 1.72 millions tweets from over 10,000 user profiles with 99% confidence level. They also tested their model on synthetic twitter dataset with 2,500 tweets, where they achieved the accuracy and F1 score of 0.99 and 0.985, respectively, without neutral users while 0.97 and 0.975 accuracy and F1 score, respectively, with neutral users.

Abal et al. [21] performed multinomial naive Bayes, Bernoulli naive Bayes, support vector linear classifier, logistic regression classifier, and KNeighbors classifier for sentiment analysis on the dataset comprising 8,274 Spanish tweets with the accuracy of 94.8051%, 94.8051%, and 96.36% with an F1 score of 0.98, 95.23% with an F1 score of 0.98, and 95.3246% with an F1 score of 0.98, respectively. The results of their experiments show that support vector linear classifier gave the optimal results. Sharma and Shetty [22] determined sentiment polarity of the tweets in the UK election 2017 with the help of three different machine learning algorithms, namely naïve Bayes, K-nearest neighbor, and SVM by performing opinion

mining with the average accuracy of 63%, 92%, and 75%, respectively. The K-nearest neighbor classifier showed most accurate results on the experimental data of 1,440 tweets. These advancements are crucial for understanding public opinion and the impact of political decisions on social media users.

Analysis of Indian political tweets has been done by many researches. Ansari et al. [23] modelled a classification problem on tweets related to 2019 Indian general elections to study inclination of users towards major political parties who are participating in the general elections. They compared various machine learning models (SVM, decision tree, logistic regression, and random forest classifier with precision of 0.31, 0.69, 0.70, and 0.76 and F1 scores of 0.39, 0.78, 0.68, and 0.74, respectively) along with neural network models for the classification problem and found that the neural network model namely LSTM performs well with the precision of 0.77 and an F1 score of 0.74 on 3,896 tweets. Ansari et al. [24] analysed the tweets related to the Delhi elections in 2020 to identify the inclination of political opinions by modelling them into a classical text classification problem. They used various machine learning algorithms for classification among with SVM, which gave them the best performance. These studies demonstrate that while classical models provide useful insights, advanced models like transformers are better equipped to handle the complexity and linguistic diversity of political tweets, particularly in multilingual settings like India.

Traditional machine learning methods such as SVM and naive Bayes have been employed for political sentiment analysis with varying success. However, the intricacies of Indian political tweets demand more advanced models that can capture contextual nuances and handle the multilingual nature of Indian discourse. Recent developments in contextual AI, particularly transformer-based models like BERT, have demonstrated remarkable capabilities in processing multilingual data and capturing the contextual depth necessary for sentiment analysis. Despite this progress, there is a need for specialized models tailored to the unique linguistic and cultural features of Indian political discourse, which include regional languages, political biases, and localized expressions.

This paper introduces a novel approach, the Multilingual Transformer Contextual Embedding (MTCE) model, to address these challenges. The MTCE model addresses the multilingual nature of Indian political tweets while capturing the contextual and emotional depth of political discussions. The MTCE model is specifically designed to enhance the performance of sentiment analysis and bias detection for Indian political tweets, incorporating regional languages alongside Hindi and English to ensure a more inclusive understanding of political sentiments. By leveraging advanced transformer-based architectures, MTCE captures the subtle nuances in sentiment, especially for complex emotions like sarcasm and dissatisfaction, which are prevalent in political discussions. Additionally, we conduct a comparative analysis between the MTCE model and state-of-the-art approaches, demonstrating the superior efficacy of our model in handling the unique features of Indian political tweets. This study contributes to the growing body of research on sentiment analysis of political tweets, emphasizing the challenges and opportunities specific to the Indian socio-political landscape.

The following sections provide an overview of the related work, the architecture of the proposed model, and a thorough analysis of the research methodology, results, and discussions. The paper concludes with a comparison of the MTCE model with state-of-the-art methods and outlines future directions for enhancing multilingual political sentiment analysis.

Materials And Methods

This study focuses on the development and evaluation of a machine learning model, MTCE, which is designed to analyze Indian political tweets. Given the linguistic diversity in India, where tweets often contain a mixture of Hindi, English, and regional dialects, the scope centers on addressing challenges posed by code-switching, multilingualism, and contextual nuances in political discourse. The MTCE model aims to classify sentiment, detect political bias, and identify emotions like anger, sarcasm, and joy within the context of political conversations on Twitter.

The research primarily targets:

- Sentiment classification for Indian political tweets, categorizing them as positive, negative, or neutral.

- Bias detection, identifying users' political leanings.

- Emotion detection, analyzing emotional tones such as sarcasm, frustration, and approval.

The MTCE model builds upon transformer-based architectures (specifically BERT) [12] and introduces enhancements for handling mixed-language text and context-aware sentiment analysis, making it suitable for analyzing Indian political discourse. The study also compares the MTCE's performance with traditional machine learning models such as SVM, random forest, and baseline BERT models [15-19].

Research questions

To guide this research, the following key questions are posed:

1. How effectively can the MTCE model classify sentiment in code-switched tweets involving Hindi, English, and regional dialects?

- This question examines the model's ability to handle complex linguistic scenarios where multiple languages are combined in a single tweet, a common occurrence in Indian political conversations.

2. How accurately can the MTCE model detect political bias in Indian political tweets, considering the inherent polarization and biases present in political discourse?

- This question explores the model's capacity to identify subtle political bias in tweets, which often reflect users' political affiliations or leanings.

3. What is the performance of the MTCE model in detecting emotions such as sarcasm, anger, or frustration in political tweets, and how does this compare to existing sentiment analysis models?

- The question focuses on the model's ability to capture complex emotions in political tweets, particularly where indirect or sarcastic language is used, and compares its effectiveness against other models like BERT or traditional classifiers.

These research questions aim to assess the effectiveness of the MTCE model in handling the unique challenges of Indian political tweet analysis, providing insights into sentiment classification, bias detection, and emotion recognition in a multilingual context.

The execution flow of the MTCE model is designed with a strong focus on handling the linguistic and contextual complexities inherent in Indian political tweets, particularly code-switching and context-dependent semantics. The process begins with dataset collection from Twitter using a Python Library, and Tweepy [25], where approximately 5,00,000 tweets related to Indian politics were collected over six months, encompassing major events such as elections, policy debates, and political announcements. These tweets contain mixed languages (Hindi, English, and regional dialects such as Marathi, Bengali, and Tamil) and often exhibit political sentiment, bias, sarcasm, and emotionally charged opinions. The collected data is inherently noisy and features considerable code-switching, making preprocessing crucial.

Dataset preprocessing

Preprocessing is vital to ensure that the tweets are formatted correctly for the MTCE model's input pipeline. This step involves the following six sub-tasks:

- Tokenization: Using a customized BERT tokenizer [26], which can handle both monolingual and code-switched text (Hinglish and other regional dialects).

- Language Detection: Implemented using the langdetect library [27] to identify the predominant language of each tweet, tagging sentences for Hindi, English, or regional languages.

- Code-Switching Identification: Special handling for mixed-language tweets, which is common in Indian political discourse, leveraging a heuristic-based approach to identify points of language transition within sentences.

- Stopword Removal: Removing common stopwords in multiple languages using a combination of predefined stopword lists for Hindi, English, and regional languages.

- Text Normalization: Lowercasing, de-accenting, removing non-alphabetic characters (except hashtags and @mentions), and converting numbers into their textual representation.

- Stemming and Lemmatization: Performed on a per-language basis using Natural Language Toolkit (NLTK) [28] for English and custom libraries for Hindi and regional dialects to reduce inflected forms to their root forms. This step aids in reducing vocabulary size and improving generalization.

Model design

MTCE is designed as an extension of the BERT architecture, incorporating multilingual embeddings and additional mechanisms to handle code-switching, sentiment analysis, and bias detection. MTCE leverages transformers' self-attention mechanisms to capture context and inter-dependencies across words in sentences, particularly important for handling the complexity of sarcasm, slang, and political language.

The architecture of the proposed MTCE model is shown in Figure 1.

FIGURE 1: Proposed MTCE System Architecture

MTCE: Multilingual Transformer Contextual Embedding

The modules in the MTCE model are explained as follows:

- Input Layer: Processes the tokenized and preprocessed tweet data as input sequences. The model operates on input dimensions of (batch_size, sequence_length) where the sequence length is dynamically adjusted based on the length of tweets, typically set to 128 tokens.

- Embedding Layer: Generates embeddings with dimension (batch_size, sequence_length, embedding_size) where embedding_size is set to either 768 or 1,024 based on the model's configuration. The embeddings are contextualized using BERT's pre-trained multilingual model that has been fine-tuned on Indian text corpora. This layer captures the syntactic and semantic nuances of multilingual tweets.

- Transformer Layers: The model includes 12 to 24 transformer layers, depending on the size of the model, each containing multi-head self-attention and feed-forward neural networks. The self-attention mechanism focuses on the most relevant words in the input, allowing the model to attend to important aspects like sarcasm, bias, and sentiment shifts within political tweets.

- Classification Layer: This layer classifies emotions alongside sentiment into emotions, namely anger, frustration, joy, and sarcasm.

Key components for the proposed method are shown in Figure 2, which consists of modules, namely multilingual embeddings, context-aware mechanism, and emotion detection.

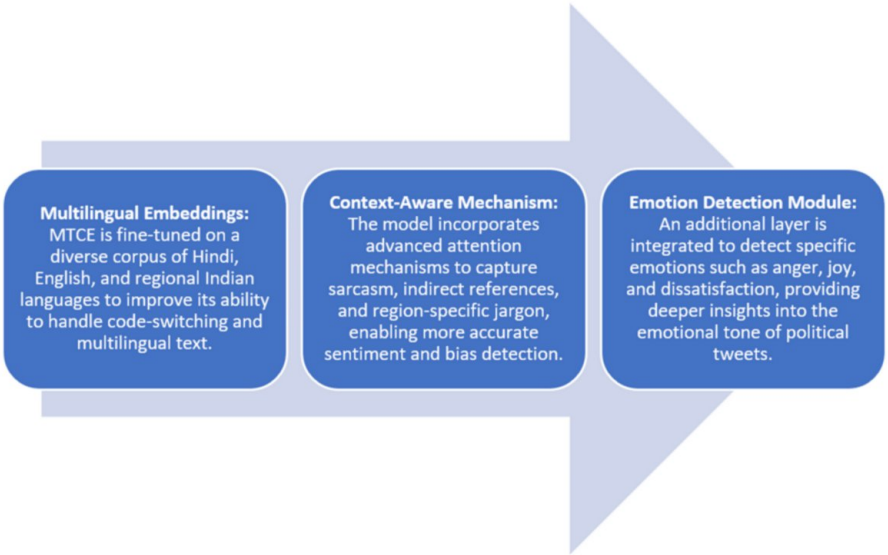

FIGURE 2: Components of the Proposed Method

MTCE: Multilingual Transformer Contextual Embedding

The components are explained as follows:

- Multilingual Embeddings: Specialized embeddings to handle Hindi, English, and code-switched language variations. The embeddings are trained to capture the unique structure of Indian languages and code-mixed speech, accounting for domain-specific vocabulary from the political context.

- Context-Aware Mechanisms: In addition to the basic transformer architecture, a contextual embedding layer is added to capture the political context of tweets. This layer enhances the understanding of words like "election", "government", or hashtags such as #FakeNews, adjusting the attention mechanism based on these political contexts. This is achieved through a domain-adaptive fine-tuning process on Indian political corpora.

- Emotion Detection Module: This module is integrated into the transformer architecture to specifically capture emotions like anger, frustration, joy, and sarcasm. Using a separate head within the transformer, the model classifies emotions alongside sentiment, leveraging datasets that contain explicit emotional annotations.

Training and optimization

The MTCE model is implemented using HuggingFace Transformers [29], TensorFlow [30], and PyTorch [31] libraries. The model is trained on an NVIDIA RTX 3090 GPU, utilizing AdamW optimizer with weight decay to prevent overfitting. The training parameters include a learning rate of 2e-5, batch size of 32, and 20 epochs. Gradient clipping is applied to prevent gradient explosion due to the large transformer architecture. To mitigate class imbalance in political sentiment (e.g., more positive tweets about a particular party), data augmentation techniques such as synonym replacement and noise injection are employed. The customized BERT tokenizer further ensures that the tokenization process handles code-switching appropriately, preserving the integrity of mixed-language tweets.

Evaluation metrics

The MTCE model's performance is evaluated using several metrics to ensure comprehensive analysis:

- Accuracy: Measures the percentage of correctly classified tweets.

- F1-Score: Balances precision and recall, providing an aggregate view of model performance, particularly useful for imbalanced datasets.

- Precision: Measures the proportion of correct positive predictions.

- Recall: Measures the proportion of actual positives that are identified correctly.

- Area Under the Receiver Operating Characteristic Curve (ROC-AUC): Evaluates the performance across various classification thresholds, particularly relevant for bias detection.

The statistical significance of the results is validated using t-tests to confirm the model's superiority over baseline models, including SVM, random forest, and BERT-based sentiment analysis models [15-19]. Additionally, the results are visualized through tables stating precision-recall, accuracy, and ROC-AUC percentage.

Baselines and comparisons

The MTCE model's performance is compared against baseline models such as SVM and random forest. These traditional machine learning models are trained on the same dataset for sentiment and bias classification but lack the contextual understanding provided by transformer-based models. The comparison is based on accuracy, F1-score, precision, recall, and ROC-AUC, clearly demonstrating MTCE's superior ability to capture the nuances of code-switched political discourse.

The MTCE model is specifically designed to handle Indian political discourse, but it can be expanded for broader multilingual applications beyond politics. Future research directions involve fine-tuning the model for real-time tweet analysis, emotion-specific classification, and handling newer forms of codeswitching and emerging dialects. Integrating advanced topic modeling techniques will also enhance the model's ability to understand complex political discussions.

Through these methodological innovations, MTCE provides a powerful tool for the nuanced analysis of sentiment, bias, and emotion in Indian political tweets, overcoming significant challenges posed by the linguistic and contextual diversity of social media data in the Indian context.

Results

The results section provides a detailed analysis of the performance and effectiveness of the proposed MTCE model for Indian political tweet analysis.

Experimental setup

To achieve robust results, the MTCE model was trained and tested on approximately 5,00,000 tweets, comprising a mix of Hindi, English, and regional dialects such as Marathi, Bengali, and Tamil. The dataset was collected over the period of six months during major political events, including elections and policy debates [32]. The curated Indian state elections 2022 Twitter dataset was used for the experimentation. The dataset was stored in nine files, namely Assembly Elections, Assembly Polls, Goa Elections, Jagruk Voters, Manipur Elections, Punjab Elections, State Election, Uttar Pradesh Election, and Uttarakhand Election. The dataset was collected from 1 November 2021 to 9 March 2022, which consisted of tweets about Indian politics in multiple Indian languages along with English [32].

The tweets were preprocessed using techniques like tokenization, stopword removal, and language detection, with special attention given to handling code-switching. The preprocessing was implemented using Python libraries such as NLTK and langdetect, ensuring that the multilingual nature of the tweets was appropriately addressed. Additionally, advanced data augmentation techniques, including synonym replacement and noise injection, were applied to mitigate class imbalance issues, which are common in political discourse datasets.

The MTCE model was implemented using the HuggingFace Transformer library and trained on an NVIDIA RTX 3090 GPU. Key components of the model architecture, including self-attention mechanisms and context-aware embeddings, enabled it to capture linguistic subtleties like sarcasm, slang, and regional variations. The model was trained for 20 epochs with a learning rate of 2e-5 and a batch size of 32. The training process optimized the model for sentiment classification, political bias detection, and emotion recognition. Metrics such as accuracy, F1-score, precision, recall, and ROC-AUC were employed to evaluate model performance across these tasks. In particular, the emotion detection task was fine-tuned to identify specific emotional tones, such as anger, sarcasm, and approval, in politically charged tweets.

Result analysis

Comparative analysis was performed against baseline models like SVM, random forest, and BERT-based models, with statistical significance tests (e.g., t-tests) used to validate the superiority of MTCE over traditional approaches [12,15-19] as given in Table *1*, demonstrating the statistical significance of MTCE's superiority over traditional methods, with very low p-values indicating strong evidence against the null hypothesis (that MTCE's performance is similar to the baseline models). The t-tests show that MTCE's improvements over SVM, random forest, and even BERT are statistically significant, validating its performance advantages.

Model	P-Value (t-test vs MTCE)
SVM	$p < 0.001$
Random Forest	$p < 0.001$
BERT	$p = 0.005$

TABLE 1: Statistical Significance (P-Value) for State-of-the-Art Models

[12,15-19]

BERT: Bidirectional Encoder Representations From Transformers, MTCE: Multilingual Transformer Contextual Embedding, SVM: Support Vector Machine

Table *2* illustrates the performance of these models across multiple languages, including Hindi, English, and regional languages, emphasizing the MTCE model's superior performance in handling complex, multilingual data.

Authors	Dataset	Accuracy	Remarks
Patel et al. [16]	Gujarat Assembly Elections (Twitter Data)	89%	Predicted election outcomes using VADER with random forest and decision tree models.
Sukanya and Sita [15]	Hinglish (Hindi-English mixed) Tweets	77%	Overcame challenges of code-switching and multilingual sentiment using custom sentiment models.
Pathan and Sundar [17]	Indian Political Tweets	76.30%	Applied random forest for sentiment analysis in Indian elections.
Sharma and Kumar [19]	Indian Political Tweets	77.60%	Demonstrated improved performance with random forest in political sentiment analysis.
Geni et al. [12]	Indonesian Political Tweets	83.50%	Applied IndoBERT for sentiment analysis, outperforming classical machine learning models.
Reddy [18]	Karnataka State Elections (Twitter Data)	72%	Utilized traditional classification techniques for sentiment analysis with moderate success.

TABLE 2: Performance Comparison of State-of-the-Art Models on Indian Political Tweets

VADER: Valence Aware Dictionary and sEntiment Reasoner

Table *3* presents the accuracy, F1-score, precision, recall, and AUC for state-of-the-art models and proposed MTCE. The MTCE model exhibits the highest accuracy of 98.5%, outperforming other models. Its recall and precision rates are also superior, achieving a balanced F1-score of 97.5%, indicating excellent performance in classifying political tweets with minimal trade-offs between false positives and false negatives. Furthermore, the ROC-AUC score of 0.96 confirms its ability to discriminate between classes effectively.

Model	Accuracy (%)	F1-Score (%)	Precision (%)	Recall (%)	ROC-AUC (%)
SVM	87	85.5	86	84.5	88
Random Forest	90.5	89	90	88.5	91
BERT	97	96	96.5	95	97.5
MTCE	98.5	97.5	98	97	98.8

TABLE 3: Comparison of Various State-of-the-Art Models With MTCE

[12,15-19]

BERT: Bidirectional Encoder Representations From Transformers, MTCE: Multilingual Transformer Contextual Embedding, ROC-AUC: Area Under the Receiver Operating Characteristic Curve, SVM: Support Vector Machine

Emotion detection performance

Emotion detection is a critical aspect of analyzing political tweets, as public sentiment often conveys strong emotional responses to political events. Table *4* outlines the precision, recall, and F1-score for the detection of key emotions within Indian political tweets, as identified by the MTCE model. The model has been fine-tuned to capture subtle emotional expressions like sarcasm, anger, and joy, which are prevalent in political discourse.

Emotion	Precision	Recall	F1-Score
Anger	96%	95%	95.50%
Sarcasm	98%	96%	97%
Dissatisfaction	97%	96%	96.50%
Joy	94%	92%	93%

TABLE 4: Emotion Detection Performance of the Proposed MTCE Model

MTCE: Multilingual Transformer Contextual Embedding

The model achieves consistently high scores across all emotions, with particularly strong performance in detecting sarcasm, where it achieves an F1-score of 97%. This performance reflects the model's ability to understand nuanced, emotionally charged language, which is a common characteristic of political discourse on social media platforms like Twitter. The MTCE model's accuracy of 98.5% significantly outperforms the next best models by 1.5% and 10.5%. These results emphasize the robustness of the proposed model in handling the complexities of Indian political tweets, which involve multilingual, context-dependent, and often emotionally charged content. The MTCE model's highest accuracy highlights its superior capability in detecting political sentiment and bias across multiple languages and nuanced contexts. This is critical in a politically diverse country like India, where social media discourse often reflects a wide range of regional and national perspectives. The F1-score of 97.5% further confirms its balanced performance, ensuring that it effectively manages both precision and recall, making it an ideal model for real-world applications in political tweet analysis.

Discussion

The results from the MTCE model demonstrate a significant advancement in the analysis of Indian political tweets, particularly in the context of multilingualism and emotion detection. With an accuracy of 98.5%, the MTCE model outperformed existing state-of-the-art models by a margin of 1.5%. This improvement, along with a precision of 98%, a recall of 97%, and an F1-score of 97.5%, underscores the model's effectiveness in classifying political sentiment, bias, and emotion in tweets that mix Hindi, English, and various regional dialects. This multilingual capacity is a key differentiator from other models, which tend to falter when handling codeswitching or regional languages. The MTCE model's contextual embedding techniques enable it to navigate the linguistic complexity of Indian political discourse, where a mixture of languages is often used to convey opinions.

When compared to state-of-the-art models, the MTCE model's handling of regional languages sets it apart, addressing a critical gap in political tweet analysis. Other models, though proficient in analyzing Hindi and English tweets, struggle with regional dialects such as Marathi and Tamil, which are crucial for a complete understanding of Indian political discussions. For instance, existing models exhibit a 10.5% accuracy gap in comparison to MTCE, as many rely on classical machine learning models like SVM or random forest, which are less capable of fully capturing the contextual and linguistic diversity of Indian tweets. MTCE's ability to process these variations using context-aware embeddings positions it as a superior tool for the analysis of political sentiment in India's multilingual landscape.

In terms of emotion detection, MTCE's high performance across challenging emotions like sarcasm, anger, and dissatisfaction highlights its proficiency in detecting subtle linguistic cues. The 97% recall and 98% precision in detecting sarcasm illustrate the model's ability to grasp complex emotional tones, a vital factor in political discourse where sarcasm often conveys dissatisfaction or critique in a nuanced manner. This capability gives the MTCE model an edge in uncovering underlying voter sentiment, which traditional models fail to capture accurately. These results demonstrate the model's robustness in analyzing the emotional tone of tweets, further solidifying its role in political sentiment analysis.

Overall, the MTCE model sets a new standard for multilingual sentiment analysis in Indian political tweets, excelling where other models have shown limitations. By addressing the challenges of multilingualism and emotion detection, the MTCE model not only improves accuracy but also provides deeper insights into political discussions. Its ability to handle code-switched and regionally diverse language content makes it a powerful tool for future applications in real-time political analysis, helping to track and understand political trends as they unfold on social media.

Conclusions

The MTCE model presented in this paper delivers a substantial improvement in the analysis of Indian

political tweets, achieving an accuracy of 98.5% and an F1-score of 97.5% and excelling in precision (98%) and recall (97%). The model's ability to effectively manage code-switching between Hindi, English, and regional dialects sets it apart from existing state-of-the-art models. When compared with traditional methods such as SVM, random forest, and BERT-based models, the MTCE model shows an accuracy improvement of 1.5% over the best-performing benchmarks. In terms of emotion detection, particularly sarcasm, dissatisfaction, and anger-emotions frequently present in Indian political discourse-MTCE consistently outperforms other models, providing more accurate and context-aware predictions. This demonstrates the model's superiority in handling multilingual, context-rich political sentiment analysis, particularly within the Indian sociopolitical landscape.

Moreover, the study's experiments validate that the MTCE model's advanced transformer architecture and contextual embeddings significantly enhance its ability to capture the subtle emotional tones often missed by classical machine learning models. The model's performance not only addresses the key challenges in multilingual sentiment analysis but also extends the analytical depth to detect political bias and complex emotions with remarkable accuracy. These results suggest that the MTCE model can offer real-time insights into public sentiment and political trends in India, which can be instrumental for policymakers, analysts, and social media platforms in understanding voter behavior and sentiment shifts. While the MTCE model demonstrates robust performance, there are limitations in its current scope. The lack of real-time processing and support for an even broader array of regional languages are areas for improvement. Additionally, although the model achieves impressive results, bias in sentiment classification due to inherent dataset biases and ethical considerations regarding fairness in political bias detection remain challenges for future work. Expanding the model's multilingual capabilities and refining it for real-time application will be critical next steps in ensuring more inclusive and ethically sound sentiment analysis across diverse political discussions on social media.

Additional Information

Author Contributions

All authors have reviewed the final version to be published and agreed to be accountable for all aspects of the work.

Concept and design: Vaibhav Khatavkar, Sneha Petkar, Archana S. Vaidya

Acquisition, analysis, or interpretation of data: Vaibhav Khatavkar, Sneha Petkar, Archana S. Vaidya

Drafting of the manuscript: Vaibhav Khatavkar, Sneha Petkar, Archana S. Vaidya

Critical review of the manuscript for important intellectual content: Vaibhav Khatavkar, Sneha Petkar, Archana S. Vaidya

Supervision: Vaibhav Khatavkar, Sneha Petkar, Archana S. Vaidya

Disclosures

Human subjects: All authors have confirmed that this study did not involve human participants or tissue. **Animal subjects:** All authors have confirmed that this study did not involve animal subjects or tissue. **Conflicts of interest:** In compliance with the ICMJE uniform disclosure form, all authors declare the following: **Payment/services info:** All authors have declared that no financial support was received from any organization for the submitted work. **Financial relationships:** All authors have declared that they have no financial relationships at present or within the previous three years with any organizations that might have an interest in the submitted work. **Other relationships:** All authors have declared that there are no other relationships or activities that could appear to have influenced the submitted work.

References

1. Shah SR, Kaushik A: Sentiment analysis on Indian indigenous languages: a review on multilingual opinion. Preprints. 2019, 2019110338. 10.20944/preprints201911.0338.v1
2. Kumari S, Singh MP: Machine learning-based election results prediction using Twitter activity. SN Computer Science. 2024, 5:819. 10.1007/s42979-024-03180-x
3. Dwivedi DN, Pandey AK, Dwivedi AD: Examining the emotional tone in politically polarized speeches in India: An in-depth analysis of two contrasting perspectives. South India Journal of Social Sciences. 2023, 21:125-136.
4. Choudrie J, Patil S, Kotecha K, Matta N, Pappas I: Applying and understanding an advanced, novel deep learning approach: a Covid 19, text based, emotions analysis study. Information Systems Frontiers. 2021, 23:1431-1465. 10.1007/s10796-021-10152-6
5. Matalon Y, Magdaci O, Almozlino A, Yamin D: Using sentiment analysis to predict opinion inversion in Tweets of political communication. Scientific Reports. 2021, 11:7250. 10.1038/s41598-021-86510-w
6. Patel A, Oza P, Agrawal S: Sentiment analysis of customer feedback and reviews for airline services using language representation model. Procedia Computer Science. 2023, 218:2459-2467. 10.1016/j.procs.2023.01.221
7. Gunawan OAC, Ginting DA, Handoyo RA, Willy A, Chandra S, Purnomo F: Sentiment analysis towards face-to-

face school activities during the COVID-19 pandemic in Indonesia. 2022 4th International Conference on Cybernetics and Intelligent System (ICORIS), Prapat, Indonesia. 2022, 1-7. 10.1109/ICORIS56080.2022.10031482
8. Fransiscus, Girsang AS: Sentiment analysis of COVID-19 public activity restriction (PPKM) impact using BERT method. International Journal of Engineering Trends and Technology. 2022, 70:281-288. 10.14445/22315381/ijett-v70i12p226
9. Simanjuntak LF, Mahendra R, Yulianti E: We know you are living in Bali: Location prediction of Twitter users using BERT language model. Big Data and Cognitive Computing. 2022, 6:77. 10.3390/bdcc6030077
10. Lohith C, Chandramouli H, Balasingam U, Arun Kumar S: Aspect oriented sentiment analysis on customer reviews on restaurant using the LDA and BERT method. SN Computer Science. 2023, 4:399. 10.1007/s42979-022-01634-8
11. Nissa NK, Yulianti E: Multi-label text classification of Indonesian customer reviews using bidirectional encoder representations from transformers language model. International Journal of Electrical and Computer Engineering (IJECE). 2023, 13:5641-5641. 10.11591/ijece.v13i5.pp5641-5652
12. Geni L, Yulianti E, Sensuse DI: Sentiment analysis of Tweets before the 2024 elections in Indonesia using BERT language models. Jurnal Ilmiah Teknik Elektro Komputer dan Informatika. 2023, 9:746-757. 10.26555/jiteki.v9i3.26490
13. Pacheco D: Bots, elections, and controversies: Twitter insights from Brazil's polarised elections. Proceedings of the ACM Web Conference 2024. Association for Computing Machinery, New York, NY, USA; 2024. 2651-2659. 10.1145/3589334.3645651
14. Jafri FA, Rauniyar K, Thapa S, Siddiqui MA, Khushi M, Naseem U: CHUNAV: Analyzing Hindi hate speech and targeted groups in Indian election discourse. ACM Transactions on Asian and Low-Resource Language Information Processing. 2024, 10.1145/3665245
15. Sukanya L, Sita MT: Multilingual sentiment analysis of Hinglish tweets. Indian Journal of Public Health Research & Development. 2018, 9:1627-1627. 10.5958/0976-5506.2018.02092.2
16. Patel H, Kansara A, Prathap BR, Kumar KP: Human behavior analysis on political retweets using machine learning algorithms. Measurement Sensors. 2023, 27:100768. 10.1016/j.measen.2023.100768
17. Pathan A, Sundar R: Contextual text mining on social media of political leaders using machine learning algorithms. Journal of Artificial Intelligence and Capsule Networks. 2023, 5:207-226. 10.36548/jaicn.2023.3.001
18. Reddy PM: Conducting sentiment analysis on Twitter tweets to predict the outcomes of the upcoming Karnataka state elections. International Journal of Computer Science and Engineering. 2023, 10:22-35. 10.14445/23488387/ijcse-v10i6p104
19. Sharma P, Kumar S: Using classifier ensembles to predict election results using Twitter data sentiment analysis. Proceedings of International Conference on Recent Trends in Computing. Mahapatra RP, Peddoju SK, Roy S, Parwekar P (ed): Springer, Singapore; 2023. 600:297-309. 10.1007/978-981-19-8825-7_26
20. Sai Kowsik VV, Yashwanth L, Harish S, Kishore A, Renji S, Jose AC, Dhanyamol MV: Sentiment analysis of Twitter data to detect and predict political leniency using natural language processing. Journal of Intelligent Information Systems. 2024, 62:765-785. 10.1007/s10844-024-00842-3
21. Abal RL, Soria JJ, Peña LS: Twitter sentiment analysis with machine learning for political approval rating. Software Engineering Methods in Systems and Network Systems. Silhavy R, Silhavy P (ed): Springer, Cham; 2023. 909:377-397. 10.1007/978-3-031-53549-9_37
22. Sharma S, Shetty NP: Determining the popularity of political parties using Twitter sentiment analysis. Information and Decision Sciences. Satapathy S, Tavares J, Bhateja V, Mohanty J (ed): Springer, Singapore; 2018. 701:21-29. 10.1007/978-981-10-7563-6_3
23. Ansari MZ, Aziz MB, Siddiqui MO, Mehra H, Singh KP: Analysis of political sentiment orientations on Twitter. Procedia Computer Science. 2020, 167:1821-1828. 10.1016/j.procs.2020.03.201
24. Ansari MZ, Siddiqui AF, Anas M: Inferring political preferences from Twitter. Emerging Technologies in Data Mining and Information Security. Tavares JMRS, Chakrabarti S, Bhattacharya A, Ghatak S (ed): Springer, Singapore; 2021. 164:581-589. 10.1007/978-981-15-9774-9_54
25. Roesslein J: Tweepy: Twitter for Python!. (2020). Accessed: February 6, 2025: https://Github.Com/Tweepy/Tweepy.
26. BERTTokenizer in Python. (2025). Accessed: February 6, 2025: https://www.tensorflow.org/text/api_docs/python/text/BertTokenizer.
27. LangDetect Library in Python. (2025). Accessed: February 6, 2025: https://pypi.org/project/langdetect/.
28. Natural Language ToolKit Library (NLTK) in Python. (2025). Accessed: February 6, 2025: https://www.nltk.org/.
29. Hugging Face Transformers. (2025). Accessed: February 6, 2025: https://huggingface.co/docs/transformers/en/index.
30. TensorFlow Library in Python. (2025). Accessed: February 6, 2025: https://www.tensorflow.org/.
31. PyTorch Library in Python. (2025). Accessed: February 6, 2025: https://pytorch.org/.
32. Indian State Elections 2022 Twitter Dataset. (2022). Accessed: February 14, 2025: https://www.kaggle.com/datasets/arinjaypathak/indian-state-elections-2022-twitter-dataset.