DOI: 10.21917/ijsc.2025.0501

EXPLORING WORD EMBEDDINGS FOR SENTIMENT ANALYSIS OF MARATHI POLITICAL TWEETS: A MACHINE LEARNING APPROACH

Swapnil P. Goje[1] and Rupali H. Patil[2]

[1]*Department of Computer Science and Applications, Dr. Vishwanath Karad MIT World Peace University, India*

[2] *Department of Computer Science, Shri Shivaji Vidya Prasarak Sanstha's Late Karmaveer Dr. P. R. Ghogrey Science College, India*

Abstract

Sentiment analysis of textual data is becoming increasingly significant in research. Many researchers are developing new technologies to enhance the accuracy and performance of sentiment analysis. This process is particularly vital in analysing customer reviews across various domains. One of new domain which was explored by the researchers is Political domain. After the inception of Smartphones and Internet availability, various political parties are using the social media to influence the people. As every people has their own opinion related to political context, they always try to put it on various social media handles like Facebook, Twitter (changed to X), Instagram, YouTube etc. As there is lots of research and resources carried out for few languages such as English, Chinese, Arabic but still few languages still lag in it like Marathi, Gujrati, Telegu, Greek etc. In view of this we have done the sentiment analysis for Marathi Tweets related to political domain using various ML models and Word embedding techniques like FastText, IndicNLP, Bag of Words and TF-IDF. We employed the hyperparameter tuning to optimize each model's performance. Among the tested embeddings, IndicNLP proved most effective, yielding superior accuracy and robustness across different machine learning models, likely due to its ability to capture linguistic intricacies specific to Indian languages. Our findings highlight the effectiveness of advanced word embeddings like IndicNLP in sentiment analysis tasks for under-resourced languages like Marathi, demonstrating their potential for broader applications in regional language processing.

Keywords:

Sentiment Analysis, Political Tweets, Word Embeddings, Machine Learning

1. INTRODUCTION

Sentiment analysis, an essential component of natural language processing (NLP), has become increasingly important with the rapid rise of user-generated content on social media. This method focuses on analysing and classifying opinions in text to identify whether they are positive, negative, or neutral. Although sentiment analysis has been extensively studied in widely used global languages like English, research on Indian languages, especially Marathi, remains relatively underexplored. Marathi, spoken by over 83 million people primarily in the state of Maharashtra, India, is one of the oldest and most richly expressive languages in the Indian subcontinent. The complex syntactic structure, morphological richness, and agglutinative nature of Marathi [1] present unique challenges for sentiment analysis, especially when applied to political discourse, where nuanced opinions and cultural context play critical roles. Marathi is considered a low-resource language compared to other Indian languages like Hindi, Bengali, and Kannada. In India, social media platforms such as Twitter have emerged as powerful arenas for political discourse, enabling users to freely share their opinions on policies, leaders, and political events. Political sentiment analysis in Indian languages like Marathi can provide valuable insights into public opinion, voter behaviour, and social trends. However, conducting sentiment analysis in Marathi presents challenges that are often absent in English and other widely studied languages. Marathi, like many other Indian languages, is characterized by free word order, inflectional morphology [2], and a large number of dialectal variations, making it difficult to directly apply sentiment analysis models that work well for English. Additionally, the lack of large, labelled datasets for Marathi political text further complicates the task, necessitating the development of specialized tools such as lemmatizers and stemmers tailored to the language.

Indian languages, including Marathi, share several characteristics that make NLP tasks like sentiment analysis more complex. These languages are resource-scarce in terms of computational tools and annotated corpora. Furthermore, Indian languages exhibit high morphological richness and are written in a variety of scripts, with Marathi primarily using the Devanagari script. Unlike English, where word boundaries are clear and sentence structures are relatively fixed, Marathi sentences often feature compound words, inflected forms, and extensive use of postpositions. These aspects require customized pre-processing techniques for sentiment analysis, such as handling code-mixed text (where Marathi is blended with English), managing spelling variations, and developing sentiment lexicons specific to the Marathi political domain. Given these challenges, the study of sentiment analysis for Marathi political tweets is essential for advancing computational tools for Indian languages and fostering a deeper understanding of the political landscape in India.

In this paper, we explored the various machine learning techniques and linguistic resources to develop an effective sentiment analysis model for Marathi political tweets. By leveraging tools such as FastText embeddings, IndicNLP resources, and custom lemmatizers for pre-processing, we aim to enhance sentiment classification performance in this underexplored language. This research contributes significantly to sentiment analysis for low-resource languages while offering valuable insights into the evolving political discourse in Maharashtra. The findings have potential implications for political campaigns, public opinion tracking, and socio-political studies.

2. RELATED WORK

In recent years, sentiment analysis research has increasingly focused on multilingual and low-resource languages, especially in linguistically rich countries like India. India's linguistic diversity presents unique challenges and opportunities for sentiment analysis, as it encompasses languages from multiple language families, including Indo-Aryan, Dravidian, and Tibeto-Burman, among others. To tackle sentiment analysis in Indian languages, researchers have employed lexicon-based methods, machine learning techniques, and hybrid approaches to address sentiment analysis in Indian languages, often leveraging language-specific resources such as annotated corpora, bilingual dictionaries, and pretrained word embeddings like fastText and Indic BERT. However, limited availability of labelled datasets, lack of standardized resources, and significant dialectal variations remain major obstacles, making sentiment analysis in Indian languages an active area of exploration and innovation.

This section provides a foundation for discussing existing methodologies, challenges, and innovations in sentiment analysis.

Patsiouras et al. [3] collected 2578 Greek Political Tweets using Twitter API keys. For Feature representation used FastText Embedding along with CNN and Transformer model for classification. The results are compared against various strategies like Figurative, Bias and Aggressiveness. In all classification types Transformer performed well than CNN.

Ying and Yufan [4] performed sentiment analysis for Travel reviews which is wrote in Chinese language. Total 8319 Travel reviews they have collected, out of which 7179 are positive and 1212 are negative reviews. For classification purpose they used RNN with Skip-Gram model of Word2Vec. This is compared against CNN, Logistic Regression, Random Forest, DT, Naïve Bayes and Traditional RNN. Proposed model performed well with 92% accuracy when compared against baseline models.

Liang-Chih Yu et al. [5] used Continuous Sentiment Contextualized Vectors (CSCV) with the use of Principal Component Analysis (PCA) along with Word2Vec and Glove. The proposed approach is compared against the CNN, LSTM, BiLSTM. All experiments are performed on Movie Reviews and Stanford Sentiments Treebank (SST-1 and SST-2) datasets. BiLSTM-EVWord2Vec performed with 88.6% accuracy on SST-2, 82.4% on Movie Review and BiLSTM-SAWE Glove gave 52.1% accuracy on SST-1 dataset.

Li et al. [6] proposed Hybrid ensemble technique of CNN and SVM with Word2Vec-TextRank on total 179,673 Amazon Reviews along with other techniques like Random Forest, Logistic Regression, Decision Tree, SVM. Proposed approach performed well with 97.13 Accuracy.

Alfreihat et al. [7] done the sentiment analysis on 58,000 Arabic Tweets with the use of NRC and NileUlex Lexicon along with ML Models like Linear SVM, Multinomial NB, Bernouli NB, SGD, Decision Tree, Random Forest and KNN. The Hybrid approach with KNN and Lexicon performed well with 88.7% accuracy. The Table.1. provided the existing research carried out related to Sentiment Analysis.

Table.1. Summary of Existing work on Sentiment Analysis

Ref	Dataset Source and Size	Methodology	Results
[9]	Political News articles	Distributed Bag of Words, Latent Dirichlet Allocation, Hard Voting Classifier and Document Embedding	Hard Voting outperforms well with 78.45% accuracy
[10]	4,445 Annotated tweets in Nepali Language.	NB, XGBoost, AdaBoost, DT, LR, SVM, DistillBERT, RoBERTa, NepaliBERT, NepNewsBERT, NepBERTa and TF-IDF Text Embedding	DistillBERT performed well with highest precision score of 0.465. SVM = MMAE score of 0.641.
[11]	19997 messages Emotion Dataset	Count Vectorization and TF-IDF for Feature Representation. SVM and Decision Tree	SVM= 88.36%
[12]	live chat of the YouTube live stream	BERT	BERT Based model with AdamW optimizer = 85% Accuracy
[13]	Tanglish (Tamil and English) comments	Created Sentiment lexicon or dictionary specifically tailored to Tanglish	Not mentioned explicitly
[14]	11,807 Bengali Reviews	Utilized DistilmBERT, mBERT, BanglaBERT, Bangla-Bert-Base, and XLM-R-base. Proposed Transformer ensemble which comprises Bangla BERT, mBERT and XLM-R-base.	Proposed Transformer ensemble performed well with 95.97% accuracy when compared against other models.
[15]	Restaurant, Laptop Reviews and CLIPEval Dataset	Proposed BERT (BERT-SCL+KEFT) framework. It is compared against BERTasp+SCAPT, BERT-SCL+KEFT, SHELLFBK and ATTLSTM	BERT-SCL+KEFT outperforms with 89.78% accuracy on Restaurant, 83.70% accuracy on Laptop Review Dataset and 87.12% Accuracy on CLIPEval dataset.
[16]	IMDB, Yelp2014, SinaWeibo, JDReview	Proposed attention-emotion-enhanced convolutional LSTM (AEC-LSTM) variants, SVM, NBSVM, CNN, ULMFit, BERT, XLNet	All Performed well on IMDB Datasets AEC-LSTM/ELSTM = 89.4% AEC-LSTM/TA = 88.7% AEC-

			LSTM/CNN = 92.8%
[17]	Movies reviews from IMDB, Product Reviews from Amazon and Tweets.	Compared proposed MAFESA with SVM, CNN, BiLSTM, BERT, GRU, Gated CNN, SRU and CSNN using GloVe Embedding	Proposed approach outperformed well on all datasets. Max Accuracy of MAFESA is 93.45%
[18]	Dravidian code-mixed and SemEval datasets	Proposed MultiSwitchNet Model which encompasses XLM-RoBERTa, DistilBERT and Convolutional Neural Network and compaired against XGBoost, BiLSTM, BERT-based, LSTM with attention, CNN-based, RoBERTa-based, mBERT, XLM-R, Udify and LASER	72.3% Accuracy with proposed MultiSwitchNet
[19]	14 Indian language Pretrained data and NER tagged data.	UPOS, XPOS tagged dataset along with FastText, Glove, MUSE, BERT, XLM and ELMO Embedding	422 embedding models for 14 Indic languages
[20]	Tweets related to 2024 Indonesia Elections related to 5 Political Leaders	Performed Data Pre-processing and used Logistic Regression Model	Overall Average Accuracy = 79%
[21]	7270 data and news articles with five attributes Serial, Headline, News, Date, and Publisher	Used ML Models Logistic Regression, MNB, SVM, SVC, RF and KNN and for feature representation Bag of Words	Logistic Regression = 85.19%
[22]	Tweets related to Gujrat and Himachal Pradesh Elections	SVM, LSTM, AdaBoost, SVM with LSTM	LSTM with SVM = 87.13% Accuracy
[23]	LIAR dataset	Naïve Bayes, Logistic Regression and Neural Network	NN = 79.00% Accuracy
[24]	2881 Marathi Concepts using Google Translator	Naive Bayes Classifiers (NBC), Random Forest, RNN, SVM, and LSTM. TF-IDF and n-gram	2304 instances for training and 577 instances for testing. RNN Outperforms well with 59% Accuracy
[25]	12116 Tokens consists Marathi, English, Code Mixed, other tokens	SVM, Naïve Bayes TF-IDF for feature representation	SVM outperforms well than NB in terms of F1 Score and Accuracy.
[26]	15900 Marathi tweets	XLM-R architecture and XLM, RoBERTa base	XLM-R Large= 83.82% Accuracy
[27]	L3CubeMahaSent, L3Cube-MahaHate, L3Cube-MahaNews	MahaBERT, MahaRoberta, Multilingual-cased, and Muril Used Parameter-Efficient Fine Tuning (PEFT) methods	L3Cube Mahasent: MahaBert-v2(LoRA) = 85.5%, L3Cube MahaHate: MahaBert-v2 -LoRA = 82.95%, L3Cube MahaNews: multilingual-cased (Adapter) =94.94%
[28]	AI4BHARAT-INDICNLP CORPORA of 10 Languages, WK and Wikipedia + Common Crawl corpus	Used Word2Vec, FastText, GloVe embeddings is trained on each languages and performance is evaluated.	In Terms of Word Similarity (Pearson correlation) FastText- Skip Gram performed well with 94.44% accuracy for Malayalam language on iNLTK Headlines Dataset. Same Embedding performed well with 99.37% on Marathi language on IndicNLP News Category Dataset.

3. METHODOLOGY

In our approach, we compared traditional feature representation methods with newer techniques. Specifically, we utilized Bag of Words, TF-IDF, FastText, and IndicNLP for feature representation. For experimentation, we employed various traditional machine learning models, including SVM, Random Forest, Logistic Regression, K-Nearest Neighbour, Gaussian Naive Bayes, and Decision Tree.

3.1 DATASET PREPARATION

The main crucial part of the research is the finding the suitable dataset or creation. As there are many sources of collecting views related to political domain are available. Indian peoples spending much time on social media as per the study [26]. After 2014 Lok

Sabha elections. Political parties from the India using various social media intensively to influence the voters. When we gone through the literature review found that most of the researchers used the Twitter for collecting the sentiments for research purpose. As our main motivation to perform this experiment related to political domain and majorly tweets made in Marathi language.

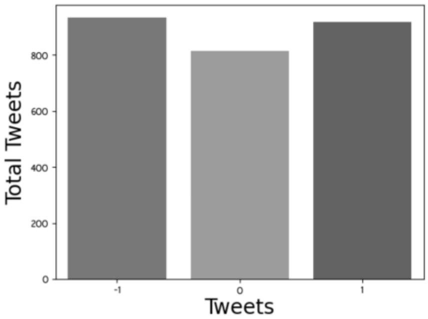

Fig.1. Dataset

Total 2667 Tweets related to political domain are considered for our research. The distribution of the tweets is given in Table.2. The tweets are collected based on few keywords which we have provided during the Tweets collection process.

Table.2. Dataset Statistics

Sentiment Class	Label	Tweets
Neutral	0	815
Positive	1	933
Negative	-1	919
Total Tweets		**2667**

3.2 PRE-PROCESSING

After collecting the dataset for preparing it to train the model and to test its accuracy, we need to remove some noisy information.

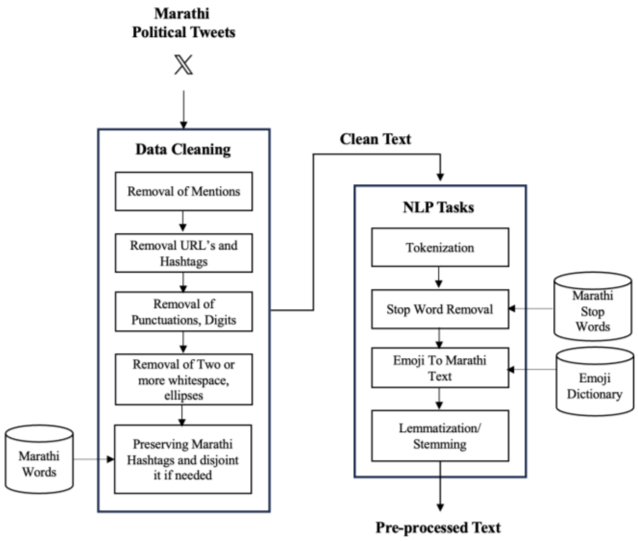

Fig.2. Pre-processing Task

The tweets contain some information which is not adding any importance for sentiment analysis, so in this phase we are removing those unnecessary noisy data from the tweet. The Data Pre-processing step is given in Fig.2. The detailed steps which we have followed during Data Pre-processing stage is elaborated in the Table.3.

Table.3. Data Pre-processing steps

Steps	Detailed Description	Example
Data Refinement	Removal of @Mentions	@miprakashbhau: दसरा मेळावा शिवतीर्थावर शिवसेनेचाच 🏳 @AUThackeray @ShivSena becomes दसरा मेळावा शिवतीर्थावर शिवसेनेचाच 🏳
	Removal of URL's, Hashtags	बापाच्या इज्जतीचा बाजार मांडणाऱ्यांना बाप पळवल्याचा त्रास होतोय. #UddhavThackeray #Shivsena https://t.co/IjtynSIKCS becomes बापाच्या इज्जतीचा बाजार मांडणाऱ्यांना बाप पळवल्याचा त्रास होतोय.
	Removal of Punctuations, English or Marathi Digits, English words with digits pattern	भारताचा विकास २०२० मध्ये गती पकडेल! India2024 becomes भारताचा विकास मध्ये गती पकडेल
	Removal of Non-Devanagari Characters	कोणीही इथे politics आणू नये becomes कोणीही इथे आणू नये
	Removal of ellipses	भयाण शांतता .... becomes भयाण शांतता
	Removal of Two or more Whitespaces	हा प्रश्न खूप महत्त्वाचा आहे becomes हा प्रश्न खूप महत्त्वाचा आहे
	Preserving Marathi Hashtags and disjoint it into proper words using dictionary.	मी #उद्धवबाळासाहेबठाकरे यांच्या बरोबर आहे becomes मी उद्धव बाळासाहेब ठाकरे यांच्या बरोबर आहे
Tokenization	The Text is splitted into tokens or individual words	मी उद्धव बाळासाहेब ठाकरे यांच्या बरोबर आहे is tokenized to ["मी", "उद्धव", "बाळासाहेब", "ठाकरे", "यांच्या", "बरोबर", "आहे"]
Stop Word removal	Removal of such words which are not useful for sentiment analysis.	मी उद्धव बाळासाहेब ठाकरे यांच्या बरोबर आहे becomes उद्धव बाळासाहेब ठाकरे बरोबर

Emoji Preservation	Emojis are considered as a important part, we preserve it using its proper Marathi Word.	मी उद्धव बाळासाहेब ठाकरे यांच्या बरोबर आहे ❤ 🏳 becomes मी उद्धव बाळासाहेब ठाकरे यांच्या बरोबर आहे प्रेम हिंदुत्व
Stemming and lemmatization	Word to base form	राजकारणी becomes राजकारण (Politics)
	Word to base form but considers the dictionary approach.	बापाचे becomes वडील (Father)

From our own created dataset of political tweets, we have created our own Lemma dictionary which consists 15293 unique words. We have manually annotated their root words and created dictionary which is used for lemmatization. We have extended the lemma dictionary by scrapping Marathi lemma from Marathi Wiktionary website using Beautiful Soup library, total 26093 words along with their lemma are scrapped using python script.

There are few words which are joined in such manner that we need to disjoint it using dictionary approach so we can preserve their sentiment importance in the given tweet. The algorithm which we have designed is given below to deal with such words.

If the words are not present in the dictionary, we have tried to found its root word by using our own created stemmer which has around 96 suffixes, by removing these suffixes we can get the root word of the given word but it’s not guarantee that it will give us the proper root word of the given word. Few of the suffixes considered are given below. We have not considered the affixes as it might remove the negation of the words.

Table.4. Sample Suffixes

◌ांच्यासारख्या	◌ाजवळ	◌ांच्याखाली
◌ांच्यासारखे	◌ाकडे	◌ांच्यापासून
◌ांच्यासारखी	◌ावर	◌ावर

The 178 stop words are handcrafted by analysing the frequency in the dataset and which are not making any contribution to the sentiment analysis. Few of the Stop words along with their frequency is listed below. These are chosen based on their linguistic importance in the sentences.

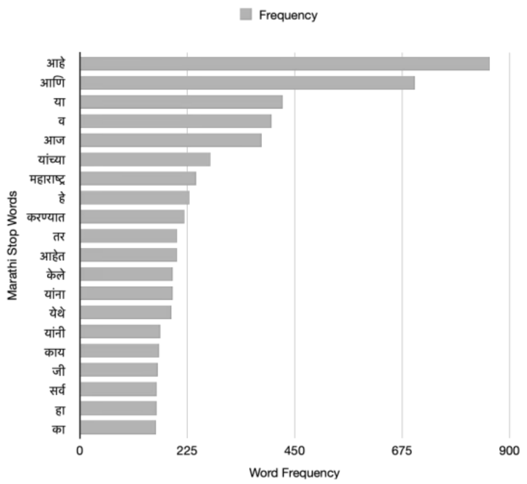

Fig.3. Sample Marathi Stop Words Frequency

	Tweets	Label	clean_text	lemmatized_text
0	कालपर्यंत शिंदे म्हणत होते, दसरा मेळाव्याबाबत ...	0	काल पर्यंत शिंदे म्हणत दसरा मेळाव्या बाबत लवकर...	काल परिणाम शिंदे म्हण दसरा मेळा बाबत लवकर कळणे...
1	हिंदुत्व, कमळाबाई, गद्दार, खोके, बापचोर, मेळाव...	-1	हिंदुत्व कमळाबाई गद्दार खोके बापचोर मेळावा चोर...	हिंदुत्व कमळाबाई गद्दार खोके बापचोर मेळा चोरी ...
2	RT @SandyNikam12: शिवसेना आणि शिवतीर्थ एक अतूट...	1	शिवसेना शिवतीर्थ अतूट नातं दसरा मेळावा	शिवसेना शिवतीर्थ अतूट नातं दसरा मेळा
3	RT @OmRajenimbalkr: धाराशिव मेडिकल कॉलेज ला रा...	0	धाराशिव मेडिकल कॉलेज राष्ट्रीय आयुर्विज्ञान आय...	धाराशिव मेडिकल महाविद्यालय राष्ट्र आयुर्विज्ञा...
4	RT @KedarDighe1: बाळासाहेबांचे विचार जिंकले! उद...	1	बाळासाहेबांचे विचार जिंकले उद्धव साहेबांचा निर...	बाळासाहेब विचार जिंकणे उद्धव साहेब निर्धार जिं...

Fig.4. Data Frame during Data Pre-processing

3.3 PROPOSED APPROACH

Once the data was cleaned, we analysed the performance of traditional machine learning models using the Scikit-learn library in Python. We employed Random Forest, Gaussian NB, Logistic Regression, Decision Tree, SVM, and K-Nearest Neighbour [29], optimizing each model’s performance through hyperparameter tuning. Their performance was evaluated against various word embedding techniques like Bag of Words (Bigram), TF-IDF, FastText, and IndicNLP. Based on our literature review, these models are among the most widely used in sentiment analysis. All experiments were conducted on a Mac M1 processor with 16GB RAM.

In our research on Marathi sentiment analysis, we have employed a diverse set of machine learning classifiers, each offering distinct advantages for this task [29]:

- **Random Forest (RF)** is chosen for its robustness and ability to handle large feature sets, like the ones generated from text embeddings, by building multiple decision trees and aggregating their outputs. It reduces overfitting and performs well with complex datasets.
- **Gaussian Naive Bayes (GNB)** is effective for text data, particularly when working with large, sparse feature vectors like TF-IDF or Bag of Words.
- **Logistic Regression (LR)** is selected for its efficiency and strong performance on linearly separable data. It is well-suited for high-dimensional data, such as word embeddings, and offers interpretable results through feature weights.
- **Decision Tree (DT)** is useful for its interpretability and ability to capture complex patterns. Though prone to overfitting, it can be an important model for understanding the feature importance in sentiment classification tasks.
- **Support Vector Machine (SVM)** is highly effective in handling high-dimensional spaces, making it particularly well-suited for text classification tasks. Its capacity to find a decision boundary with maximum margin ensures better generalization, especially with non-linear kernels like RBF or linear kernels.
- **K-Nearest Neighbour (KNN)** is included for its simplicity and ability to capture local patterns in data. While computationally expensive, it can perform well in cases where sentiment is influenced by nearby examples in the feature space.

Using this ensemble of models allows us to compare performance across linear, probabilistic, and tree-based algorithms, ensuring the selection of the best approach for Marathi sentiment analysis.

3.4 FASTTEXT AND INDICNLP WORD EMBEDDING

FastText, a word embedding model developed by Facebook, enhances the Word2Vec model by incorporating subword information. Unlike traditional embeddings, FastText represents each word as a collection of character n-grams, enabling it to manage rare and out-of-vocabulary words more efficiently, especially in morphologically rich languages such as Marathi.

IndicNLP is a suite of tools specifically designed for processing Indian languages. It provides pre-trained word embeddings that capture the linguistic nuances of various Indian languages, including Marathi, enabling better semantic understanding in tasks such as sentiment analysis. The embeddings are generated using large-scale corpora and are tailored to handle the diversity of Indian languages.

One of the most important tasks before training and testing the model is to choose the fine tune the model. Find tuning model gives you the idea about model’s performance on the basis of few parameters like its validation score, test score, mean test accuracy, train loss, AUC score etc. We have performed the fine tuning using cross validation techniques using 10-fold. For having good results, we have properly selected the hyper parameters for each model. The Table.4 provides a summary of the hyperparameters chosen for each model for the respective feature representation technique.

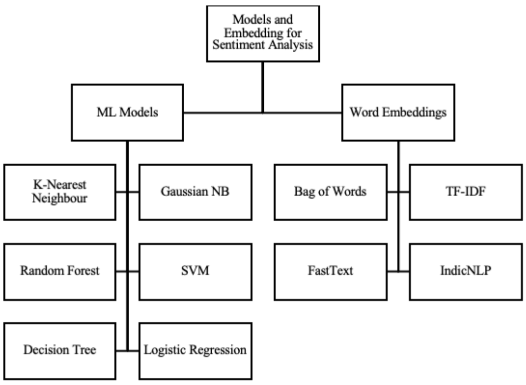

Fig.5. Overview of Sentiment Analysis models and Word Embedding used in Study

For optimization, we have employed a grid search with a range of different parameters for each estimator and chosen the best hyperparameters based on good results. The fine-tuned models indicates that model performs well for each class and isn’t biased toward one over others and it also suggests that model has a balanced understanding of the data, capturing nuance between the classes effectively. For effective sentiment analysis, we need to have a good understanding about how well the model is generalized to unseen test data. Every model’s accuracy is assessed by splitting the dataset into training and testing sets.

Table.5. Hyperparameters and tuning strategies for each model

Model	Hyper-parameters	Bag of Words	TF-IDF	FastText	IndicNLP
SVM	C: Regularization parameter, kernel: Kernel type	C: 20, kernel: rbf’	C: 1, kernel: linear	C: 20, kernel: linear	C: 10, kernel: linear
Random Forest	n_estimators: Number of decision trees	n_estimators: 100	n_estimators: 100	n_estimators: 100	n_estimators: 100
Logistic Regression	C: Regularization parameter, Solver: Optimization algorithm	C: 1 Solver: liblinear	C: 1 Solver: liblinear	C: 10 Solver: liblinear	C: 10 Solver: liblinear
K-Nearest Neighbour (KNN)	algorithm: Search strategy, n_neighbors: Number of neighbors, weights: Weight function	algorithm:ball_tree, n_neighbors: 7, weights: distance	algorithm:ball_tree, n_neighbors: 10, weights: uniform	algorithm: auto, n_neighbors: 8, weights: uniform	algorithm: auto, n_neighbors: 8, weights: distance
Gaussian NB	var_smoothing: Portion of variance added to data	var_smoothing: 1e-05	var_smoothing: 1e-05	var_smoothing: 1e-09	var_smoothing: 1e-09
Decision Tree	criterion: Split quality metric, max_depth: Max tree depth, min_samples_split: Min samples	criterion: gini, max_depth: None, min_samples_split: 10	criterion: gini, max_depth: None, min_samples_split: 10	criterion: gini, max_depth: 6, min_samples_split: 10	criterion: gini, max_depth: 4, min_samples_split: 5

The detailed process of Methodology is given in Fig.6.

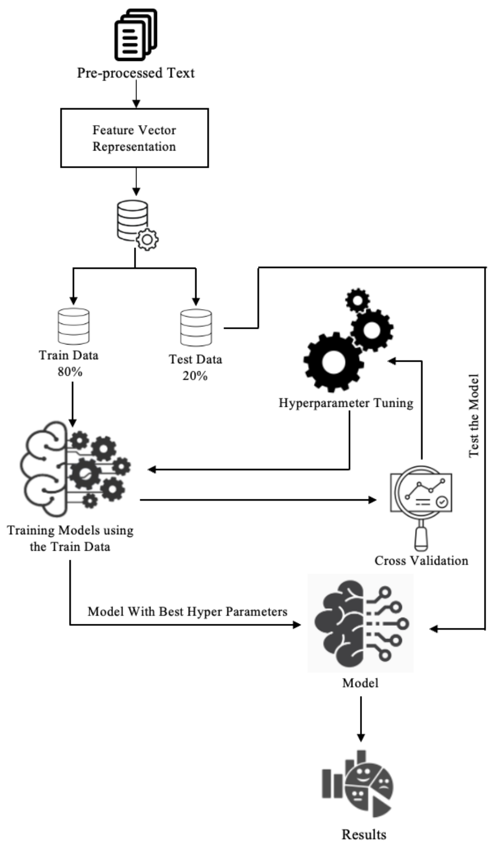

Fig.6. Proposed Methodology

3.5 MODEL TRAINING AND EVALUATION

As per the model tuned with the specified hyperparameters, we tested the models using training and testing datasets. 80% dataset is used for training the model and 20% for testing the model, with validation. To evaluate the models, we employed a cross-validation strategy with 10 folds. The trained models were evaluated based on different evaluation metrics which includes Precision, F1-Score, Recall, mean test accuracy and accuracy. Additionally, Multi-Class ROC and AUC were used, as our classification problem involves three classes: 0 (Neutral), 1 (Positive), and -1 (Negative). We also implemented a One-vs-Rest approach for generating ROC and AUC Curve.

3.6 MODEL PERFORMANCE

The different model's performance is evaluated against its feature representation and the fine-tuned hyper parameters. The various evaluation metrics which we have utilized to test the performance and performance metrics which we have utilized are given below.

The mean test accuracy is a metric that measures the model's overall performance across various data subsets, ensuring that it doesn't become overfitted to a particular subset. It offers an average performance score across multiple folds in cross-validation.

The test score serves as the ultimate indicator of how accurately the model is predicted to perform on data it has not encountered before. It demonstrates the model's capability to apply its knowledge to practical situations, preventing it from simply memorizing the training data.

Validation score: this score assists in assessing the model's performance during the tuning phase, without considering the test set. It is essential for comprehending how effectively the model performs with the training data before conducting final testing or deployment.

The area under the curve (auc) measures the overall effectiveness of the classifier by determining the total area under the receiver operating characteristic (roc) curve. A higher auc value signifies better performance, as it quantifies the model's capability to differentiate between positive and negative classes. The roc curve is a graphical representation that displays the true positive rate against the false positive rate at different threshold settings.

4. RESULTS AND DISCUSSION

This section is elaborating proposed approach performance related to the various evaluation metrics as discussed previously.

Table.6. Performance of models with FastText Embedding

Word Embedding	Evaluation Metrics	Classifiers					
		SVM	**RF**	**LR**	**KNN**	**Gaussian NB**	**DT**
FastText	Highest Fold Accuracy	81.25	68.75	**83.13**	75.63	77.50	60.63
	Mean Test Accuracy	73.00	61.00	**74.00**	70.00	69.00	56.00
	Test score	77.00	65.00	**78.00**	72.00	72.00	56.00
	Validation Score	74.04	62.54	**76.02**	68.35	71.91	58.05

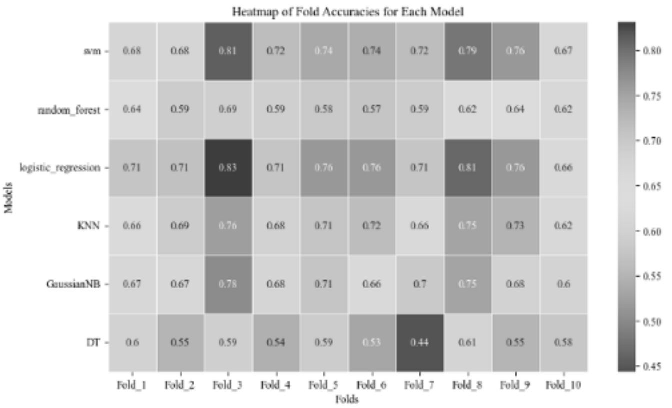

Fig.7. Heatmap of Fold Accuracies using FastText

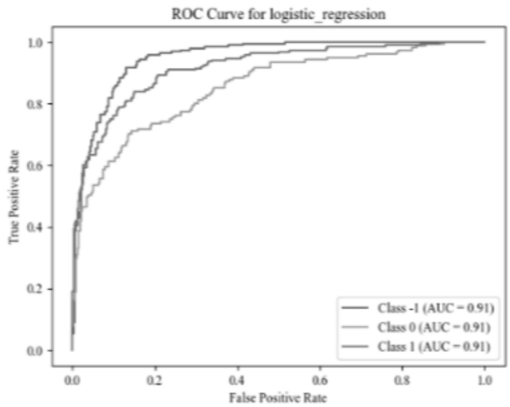

Fig.8. ROC Curve of LR

The evaluation of classifiers using FastText word embeddings shown in Table.6 shows that Logistic Regression (LR) outperformed others, achieving the highest fold accuracy at 83.13% and a mean test accuracy of 74.00%. Support Vector Machine (SVM) followed closely with a fold accuracy of 81.25%. In contrast, Decision Tree (DT) consistently underperformed, with a mean test accuracy of only 56.00%. Overall, LR demonstrated the most effective performance among the classifiers tested.

The AUC values for the models indicate that Logistic Regression (0.906) and SVM (0.905) offer the best performance in distinguishing between the three sentiment classes, with a high degree of accuracy as shown in Fig.8.

Table.7. Performance of models with IndicNLP Embedding

Word Embedding	Evaluation Metrics	Classifiers					
		SVM	**RF**	**LR**	**KNN**	**Gaussian NB**	**DT**
IndicNLP	Highest Fold Accuracy	**82.00**	68.13	80.00	75.00	76.88	63.13
	Mean Test Accuracy	**74.79**	63.47	74.36	70.60	68.60	58.91
	Test score	79.03	68.54	**80.34**	74.72	71.72	62.17

	Validation Score	**75.65**	64.79	75.46	70.41	71.16	64.04

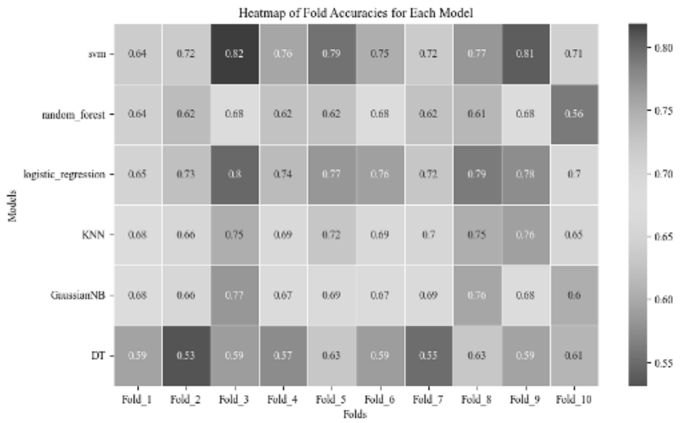

Fig.9. Heatmap of Fold Accuracies using IndicNLP

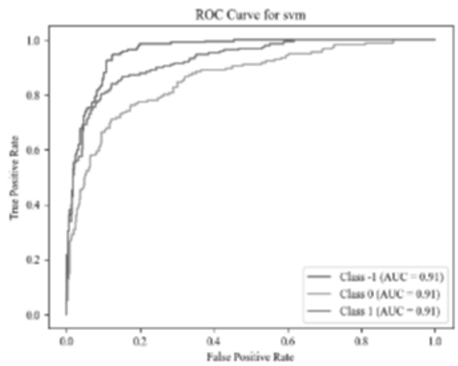

Fig.10. ROC Curve

The Table.7 shows SVM demonstrated strong performance with the highest fold accuracy of 82.00% while and a test score of 80.34%. SVM achieved the highest fold accuracy of 82.00%, while KNN and Gaussian NB also performed reasonably well, with test scores of 74.72% and 71.72%, respectively. In contrast, Decision Tree consistently showed lower performance across all metrics, with a test score of 62.17%. Fig.9 showing the Heatmap of Fold accuracies for all models during each fold.

As per Fig.10, AUC values indicate that SVM (0.914) and Logistic Regression (0.913) have the highest discriminative ability for sentiment classification, showing excellent performance. Random Forest (0.885) also performs well, closely followed by KNN (0.873) and Gaussian NB (0.865). Decision Tree (0.781) still exhibits lower AUC, suggesting it struggles to match the generalization ability of the other models

Table.8. Performance of models with TF-IDF Embedding

Word Embedding	Evaluation Metrics	Classifiers					
		SVM	**RF**	**LR**	**KNN**	**Gaussian NB**	**DT**
TF-IDF	Highest Fold Accuracy	69.38	65.00	**71.88**	66.88	61.25	62.50
	Mean Test Accuracy	64.85	60.22	**66.41**	57.28	53.41	56.03
	Test score	**69.85**	62.92	67.60	58.61	57.12	56.93
	Validation Score	65.35	61.42	**66.47**	56.74	52.99	54.30

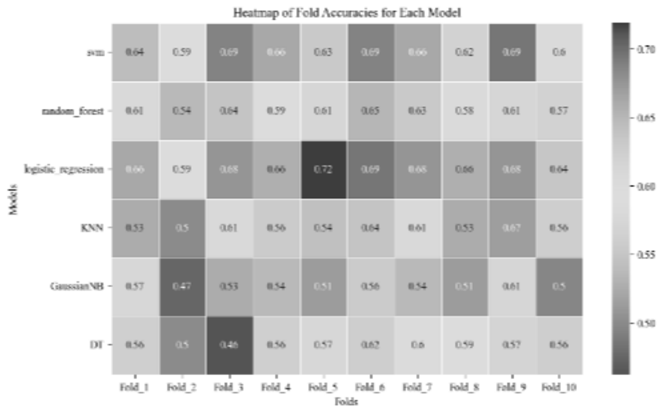

Fig.11. Heatmap of Fold Accuracies using TF-IDF

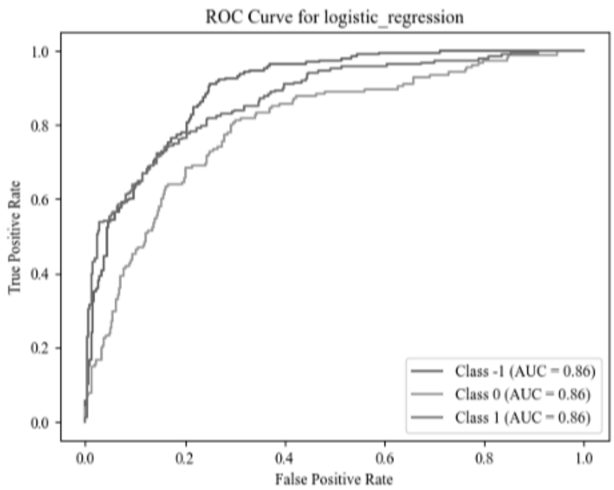

Fig.12. ROC Curve of LR

As per the Table.8 and Table.9 it clearly shows that Logistic Regression performed well than other classifiers in terms of Mean Test Accuracy and Validation Score. SVM and LR performed close in terms of unseen Test data using Bag of Words and TF-IDF. As the Bag of Words and TF-IDF used dense representation for features and it's not preserving any context about the sentence the accuracy is very low when compared against FastText and IndicNLP. Overall, the IndicNLP performs well than other feature representation technique.

Table.9. Performance of models with Bag of Words Embedding

Word Embedding	Evaluation Metrics	Classifiers					
		SVM	**RF**	**LR**	**KNN**	**Gaussian NB**	**DT**
Bag of Words	Highest Fold Accuracy	68.75	70.63	**71.88**	58.75	65.00	63.75
	Mean Test Accuracy	63.10	61.47	**66.16**	50.78	57.78	58.29
	Test score	**68.54**	67.04	**68.54**	58.05	64.23	58.05
	Validation Score	62.35	59.36	**63.48**	49.81	53.74	52.99

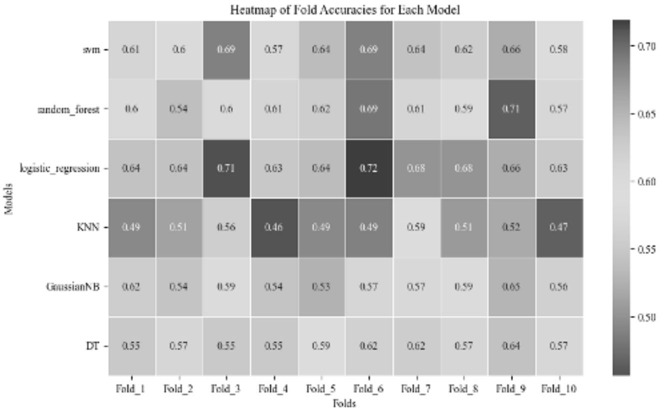

Fig.13. Heatmap of Fold Accuracies using Bag of Words

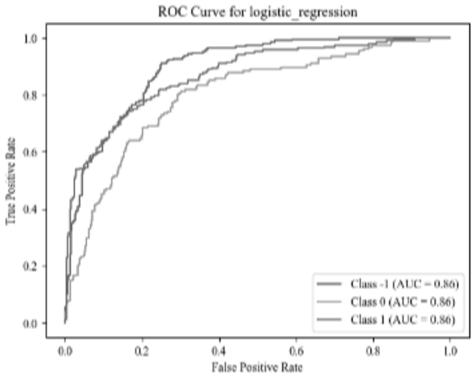

Fig.14. ROC Curve of LR

5. CONCLUSION

The results demonstrate that Logistic Regression consistently outperformed other classifiers for Marathi political sentiment analysis across various word embedding techniques. Specifically, using FastText embeddings, Logistic Regression achieved the highest mean test accuracy (74.00%) and AUC (0.906), closely followed by SVM with an AUC of 0.905. Both models demonstrated strong discriminative ability, while Decision Tree consistently underperformed, indicating its limited suitability for this task. The Bag of Words and TF-IDF embeddings yielded lower accuracy, primarily due to their inability to capture contextual information, in contrast to the more nuanced word representations from FastText and IndicNLP.

When evaluating IndicNLP embeddings, the results showed improved performance compared to traditional methods like Bag of Words and TF-IDF. IndicNLP, tailored for Indian languages, captured linguistic nuances and context, contributing to superior classifier performance, especially with Logistic Regression and SVM. This highlights the importance of using embeddings specifically designed for morphologically rich languages like Marathi. In particular, IndicNLP proved to be the most effective feature representation, enhancing both the accuracy and AUC of the models.

For future work, further exploration of deep learning approaches, such as transformers and contextual embeddings like BERT, fine-tuned for Marathi, Lexicons can be considered to further boost model performance. Additionally, expanding the dataset by incorporating more diverse sources of Marathi text beyond Twitter could improve the generalizability of the models. Finally, the use of multilingual embeddings or transfer learning techniques may also open new avenues for improving sentiment analysis in regional languages like Marathi, ensuring more comprehensive context understanding and better sentiment classification.

REFERENCES

[1] P. Rupali Patil, R.P. Bhavsar and B. V. Pawar, “Automatic Marathi Text Classification”, *International Journal of Innovative Technology and Exploring Engineering*, Vol. 9, pp. 2446-2454, 2019.

[2] R.P. Patil, R.P. Bhavsar and B.V. Pawar, “A Note on Indian Languages Text Classification Systems”, *Asian Journal of Mathematics and Computer Research*, Vol. 15, No. 1, pp. 41-55, 2017.

[3] E. Patsiouras, I. Koroni, I. Mademlis and I. Pitas, “GreekPolitics: Sentiment Analysis on Greek Politically Charged Tweets”, *Proceedings of European Conference on Signal Processing*, pp. 1320-1324, 2023.

[4] W. Ying and H. Yufan, “Research on Sentiment Analysis of Travel Reviews based on Word2Vec and RNN”, *Proceedings of International Conference on Electrical, Automation and Computer Engineering*, pp. 133-137, 2023.

[5] Liang-Chih Yu, Jin Wang, K. Robert Lai and Xuejie Zhang, “Refining Word Embeddings for Sentiment Analysis”, *Proceedings of International Conference on Empirical Methods in Natural Language Processing*, pp. 534-539, 2017.

[6] J. Li, Y. Huang, Y. Lu, L. Wang, Y. Ren and R. Chen, “Sentiment Analysis using E-Commerce Review Keyword-Generated Image with a Hybrid Machine Learning-based Model”, *Computers Materials and Continua*, Vol. 80, No. 1, pp. 1581-1599, 2024.

[7] M. Alfreihat, O.S. Almousa, Y. Tashtoush, A. AlSobeh, K. Mansour and H. Migdady, “Emo-SL Framework: Emoji Sentiment Lexicon using Text-Based Features and Machine Learning for Sentiment Analysis”, *IEEE Access*, Vol. 12, pp. 81793-81812, 2024.

[8] A.K. Sinha, S.S. Kumar Singh and S. Sai, “Detecting the Political Bias in News Articles and Similarity using Word Embeddings in American Journalism”, *Proceedings of International Conference on Computational Intelligence for Information, Security and Communication Applications*, 2023, pp. 121-125, 2023.

[9] K. Rauniyar, “Multi-Aspect Annotation and Analysis of Nepali Tweets on Anti-Establishment Election Discourse”, *IEEE Access*, Vol. 11, pp. 143092-143115, 2023.

[10] A. Joshi, S.V. Akram and A.K. Dixit, “Multilingual Sentiment Analysis for Android Mobile Application (Bandhu) for Aged People”, *Proceedings of International Interdisciplinary Humanitarian Conference for Sustainability*, pp. 980-987, 2022.

[11] A. Rath, B. Hridaya, D. Vimala and J. George, “Multilingual Sentiment Analysis of YouTube Live Stream using Machine Translation and Transformer in NLP”, *Proceedings of International Conference on Trends in Quantum Computing and Emerging Business Technologies*, pp. 1-5, 2022.

[12] M. Janotheepan, S.D.H.S. Wickramarathna, M.J.A. Amas, T. Rajeetha, A.K.L.M. Farhath and H.A.F. Sanas, "Sentiment Analysis for YouTube Cooking Recipes Videos using User Comments", *Proceedings of International Conference on Advanced Research in Computing*, pp. 235-240, 2024

[13] Md. Nesarul Hoque, Umme Salma, Md. Jamal Uddin, Md. Martuza Ahamad, Sakifa Aktar, "Exploring Transformer Models in the Sentiment Analysis Task for the Under-Resource Bengali language", *Natural Language Processing Journal*, Vol. 8, pp. 1-7, 2024.

[14] Junsen Fu, Xianyong Li, Yihong Zhu, Yajun Du, Yongquan Fan, Xiaoliang Chen, Dong Huang, Shumin Wang, "An Implicit Aspect-based Sentiment analysis Method using Supervised Contrastive Learning and Knowledge Embedding", *Applied Soft Computing*, Vol. 167, pp. 1-6, 2024.

[15] F. Huang, X. Li, C. Yuan, S. Zhang, J. Zhang and S. Qiao, "Attention-Emotion-Enhanced Convolutional LSTM for Sentiment Analysis", *IEEE Transactions on Neural Networks and Learning Systems*, Vol. 33, No. 9, pp. 4332-4345, 2022.

[16] V. Jothi Prakash, S. Arul Antran Vijay, "A Multi-Aspect Framework for Explainable Sentiment Analysis", *Pattern Recognition Letters*, Vol. 178, pp. 122-129, 2024.

[17] V. Jothi Prakash, S. Arul Antran Vijay, "A Novel Socio-Pragmatic Framework for Sentiment Analysis in Dravidian-English Code-Switched Texts", *Knowledge-Based Systems*, Vol. 300, pp. 1-7, 2024.

[18] Kumar Saurav, Kumar Saunack, Diptesh Kanojia and Pushpak Bhattacharyya, "A Passage to India": Pre-trained Word Embeddings for Indian Languages", *Proceedings of International Conference on Spoken Language Technologies for Under-Resourced Languages*, pp. 352-357, 2020.

[19] F.A. Ramadhan and P.H. Gunawan, "Sentiment Analysis of 2024 Presidential Candidates in Indonesia: Statistical Descriptive and Logistic Regression Approach", *Proceedings of International Conference on Data Science and Its Applications*, pp. 327-332, 2023.

[20] T. Samin, N.A. Mouri, F. Haque and M.S. Islam, "Sentiment Analysis of Bangladeshi Digital Newspaper by using Machine Learning and Natural Language Processing", *Proceedings of International Conference on Electrical Engineering and Information and Communication Technology*, pp. 149-153, 2024.

[21] S. Kumar and M.T. Uddin Haider, "Sentiment Analysis of Political Party Twitter Data using Ensemble Learning Classifier", *Proceedings of International Conference on Computing Communication and Networking Technologies*, pp. 1-5, 2023.

[22] S. Bhadauria, M. Shukla, P. Kumar and N. Dwivedi, "Forecasting Election Sentiments: Deep Learning vs Traditional Models", *Proceedings of International Students Conference on Electrical, Electronics and Computer Science*, pp. 1-6, 2024.

[23] R. Gaikwad, S. Nagare, M. Baheti, A. Syed, S. Maher and P. Dapke, "A Machine Learning and DL Approach to Marathi Sentiment Analysis using Senticnet", *Proceedings of International Conference on Contemporary Computing and Communications*, pp. 1-6, 2024.

[24] V. Patwardhan, G. Takawane, N. Kelkar, O. Gaikwad, R. Saraf and S. Sonawane, "Analysing the Sentiments of Marathi-English Code-Mixed Social Media Data using Machine Learning Techniques", *Proceedings of International Conference on Emerging Smart Computing and Informatics*, pp. 1-5, 2023.

[25] N. Rathod, N. Mistry, D. Talati, M. Parikh, A. Kore and P. Kanani, "Marathi Social Media Opinion Mining using XLM-R", *Proceedings of International Conference on Applied Artificial Intelligence and Computing*, pp. 730-736, 2022.

[26] P. Deshmukh, N. Kulkarni, S. Kulkarni, K. Manghani and R. Joshi, "Leveraging Parameter Efficient Training Methods for Low Resource Text Classification: A case study in Marathi", *Proceedings of International Conference for Convergence in Technology*, pp. 1-6, 2024.

[27] D. Goswami, S. Malviya, R. Mishra and U.S. Tiwary, "Analysis of Word-level Embeddings for Indic Languages on AI4Bharat-IndicNLP Corpora", *Proceedings of International Conference on Electrical, Electronics and Computer Engineering*, pp. 1-4, 2021.

[28] Digital-2024, Available at https://datareportal.com/reports/digital-2024-india, Accessed in 2024.

[29] P. Swapnil Goje and H. Rupali Patil, "Sentiment Analysis of Political Parties on Social Media: A Machine Learning and Lexicon-based Approach", *International Journal of Scientific Research in Computer Science Engineering and Information Technology*, Vol. 10, No. 5, pp. 233-246, 2024.