

# Sentiment Analysis on Tamil Code-Mixed Text using Bi-LSTM

Pradeep Kumar Roy<sup>1</sup>, Abhinav Kumar<sup>2</sup>

<sup>1</sup>Indian Institute of Information Technology, Surat, Gujarat, India

<sup>2</sup>Siksha 'O' Anushandhan, University, Bhubaneswar, Odisha, India

## Abstract

Sentiment analysis is one of the most researched topics in the computer science domain. Whenever the term opinion appears, sentiment analysis is required. Many business sectors are growing by analyzing users opinions about their products. E-commerce portals like Amazon and Flipkart offering users to express their opinion by posting the purchased product review. Further, the next buyer of the same product utilizes the user's review to make their decision-should purchase or not. Existing models of sentiment analysis mostly referred to English language textual comments. However, currently, users are posting the comments and reviews in mixed languages like Hindi-English, Malayalam-English and similar ones; it is called code-mixed languages. To identify the user sentiment from the code-mixed language, this research suggested a deep learning-based framework. The proposed framework automatically extracts the features from input sentences and predicts their sentiment with a 0.552 F1-score for the best case.

## Keywords

Sentiment Analysis, Code-Mixed, Tamil, Deep Learning, LSTM, Machine Learning

## 1. Introduction

People are expressing their opinion about things using natural language on different platforms, including YouTube, Facebook, Twitter and others [1, 2]. Analyzing the user's post and finding its opinion plays a vital role in the decision-making system and has the power to lift or down accordingly. For example, an E-commerce portal like Flipkart offering users to express their opinion about the product in the form of a review. This review helps the buyer to take their decision like whether the product is good or not [3]. Similarly, a newly released movie is good or bad can be predicted by the user's opinion available of online portal like IMDB. Currently, the Internet is reached to almost every individual, and hence user's comments are available in high volume.

To process the comments or user's review, many frameworks developed earlier using various machine learning and deep learning frameworks [4]. Most of the previous research work processed the comments or the user's review written in English text to develop sentiment

---


Forum for Information Retrieval Evaluation, December 13-17, 2021, India

✉ pkroynitp@gmail.com (P. K. Roy); abhinavkumar@soa.ac.in (A. Kumar)

ORCID 0000-0001-5513-2834 (P. K. Roy); 0000-0001-9367-7069 (A. Kumar)



© 2021 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

analysis frameworks. However, currently, a high volume of comments are posted by the users in mixed languages. For example- Kannada-English, Malayalam-English, Hindi-English and many more. Hence, the model developed so far may not be capable of handling the recent code-mixed comments [5].

The research community has recently been interested in sentiment analysis of code-mixed language. Kumar et al. [6] suggested a hybrid CNN-Bi-LSTM model for categorizing social media postings into distinct sentiment groups. To categorize Tamil-English and Malayalam-English code-mixed social media postings into distinct sentiment classes, Mahata et al. [7] suggested Bi-directional LSTM with language tagging. On the other side, Sharma and Mandalam [8] used sub-word level representation to capture text sentiment and an LSTM network to categorize Tamil-English and Malayalam-English social media postings into distinct polarity classes. Goswami et al. [9] proposed a morphological attention model for sentiment analysis on Hinglish data. Banerjee et al. [10] reported the finding of machine translation for Dravidian language such as English to Tamil, English to Malayalam, and similar ones.

In line with the works developed for sentiment analysis from code-mixed social media posts, we developed a deep neural model using Bi-directional LSTM [11, 12]. The data used for this research was developed by scraping the YouTube comments and labelled into five sentimental categories as: "positive, negative, neutral, mixed feelings or not in the intended languages" [13, 14]. Traditional machine learning classifiers and deep neural network-based LSTM are used to classify the Tamil code-mixed dataset. The experimental outcomes confirmed that the proposed model outperforms the traditional machine learning-based models by achieving higher prediction accuracy.

The rest of the paper is organized as follows: Section 2 discusses the proposed methodology. In Section 3, we discuss the experimental outcomes, and finally, Section 4 concludes the work.

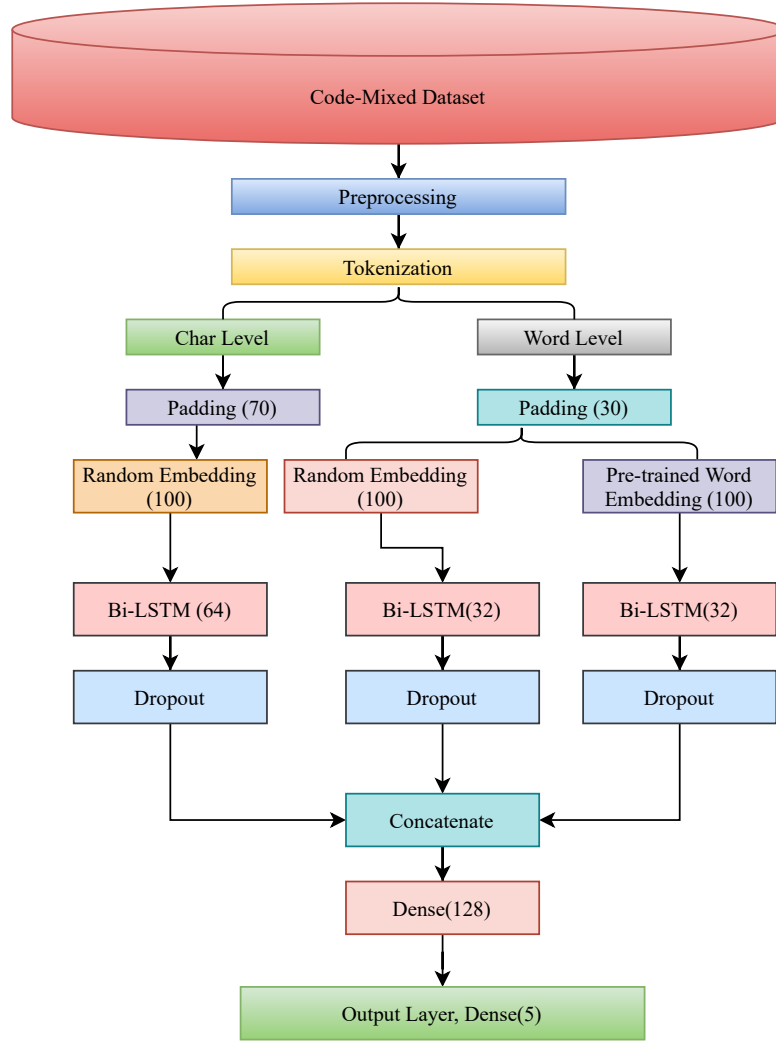
## 2. Methodology

This research suggested a framework to predict the sentiment analysis of code-mixed data using Bi-directional LSTM model Neural Network [11, 12]. The working steps of the proposed model are shown in Figure 1. The dataset used in this research is available on FIRE-2021<sup>1</sup> and was developed by Chakravarthi et al. [15, 16]. The statistics of the dataset used for model training and testing with the number of instances available in each category of the sentiment is shown in Table 1. The majority of the sample of the total dataset belongs to a positive sentiment class, whereas the remaining samples are distributed into four other categories.

The original dataset contains many unsupportive characters, which needs to be filtered out before passing it to the model for processing. The data cleaning step is performed to remove the emojis, special characters, non-ASCII characters. The number is removed and converted all information into lower cases. Further, the cleaned data passes are padded with zeros for making all messages of equal length. The maximum size of the message is fixed to 30 and 70, respectively, for the word and char level processing. The padded text passes to the embedding layer, where for each word, their corresponding vector is extracted from a pre-trained word embedding called GloVe[17]. The GloVe embedding, having the dimension of 100, means each

---

<sup>1</sup><https://dravidian-codemix.github.io/2021/datasets.html>



**Figure 1:** Proposed Framework for Sentiment Analysis with Code-Mixed Dataset

input word is mapped into 100-dimensional vectors. This way, for each message of size  $(n)$  and  $n \times 100$  sizes matrix created by the embedding layer. Random embedding technique is also used for the word and char level dataset as shown in Figure 1 with the same output dimension as 100.

Further, the embedded dataset passes to Bi-directional Long-Short Term Memory (Bi-LSTM) model for further processing. For Char embedding, 64 units of Bi-LSTM were used, whereas for processing the words, 32 units of Bi-LSTM. The dropout layer is added in all three cases, and then the outcomes are concatenated together. The concatenated outcomes of the Bi-LSTM models are passes to a fully connected dense layer with 128 neurons, followed by an output layer consisting of five neurons. The ReLU activation function is used in the internal layer of the network; however, at the output layer, Softmax is used.

**Table 1**

Data Statistics for Code-Mixed Tamil Dataset

Class	Training	Validation
Positive	20069	2257
Unknown State	5628	611
Negative	4271	480
Mixed feelings	4020	438
not-Tamil	1667	176

**Table 2**

Results with bi-gram features with ML classifiers

Class	NB			LR			RF		
	P	R	F1	P	R	F1	P	R	F1
Positive	0.58	0.99	0.73	0.61	0.94	0.74	0.62	0.93	0.74
Unknown State	0.00	0.00	0.00	0.04	0.00	0.00	0.10	0.01	0.02
Negative	1.00	0.00	0.00	0.44	0.05	0.08	0.46	0.12	0.19
Mixed feelings	1.00	0.00	0.00	0.38	0.06	0.10	0.27	0.03	0.05
not-Tamil	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Weighted Avg	0.56	0.57	0.42	0.45	0.55	0.44	0.45	0.55	0.46

### 3. Results

This research developed a model to classify the code-mixed input sentence in one of the pre-defined sentimental categories. To evaluate the model performance, the classification metrics called precision (P), recall (R), and F1-score (F1) are used. Precision is defined as the number of correctly predicted sentiment categories among the retrieved instances of the particular sentiment category. The recall is defined as the number of correctly predicted sentiment categories among the total number of instances of that particular sentiment category. The F1-score (F1) is the harmonic mean of the precision and recall [11, 18].

A number of the experiment was done with by extracting the various n-gram features from the text using tf-idf vectorization technique and passing it to traditional Machine Learning based classifiers like- Random Forest (RF), Logistic Regression (RF), and Naive Bayes (NB). The best outcomes of these classifiers are shown in Table 2. Most of the instances are miss-classified to another category of sentiment. The positive sentiment category is predicted with the highest prediction accuracy by all three classifiers, NB, LR, and RF. In contrast, the same classifiers are failed to detect the not-Tamil sentiment category. None of the classifier's performance was satisfactory for predicting code-mixed data of negative, mixed-feelings, unknown state and not-Tamil categories with bi-gram features.

To improve the model performance, we have used deep learning-based Bi- directional LSTM. The outcomes of the proposed B-LSTM model with validation dataset is shown in Table 4. The best performance is achieved for the Positive sentiment class. The precision, recall and F1-score values are 0.68, 0.80, and 0.74, respectively, whereas the lowest precision, recall and F1-score values are 0.20 0.11, 0.14, respectively, for mixed-feelings sentiment class. The performance of the proposed deep learning model outperforms the traditional machine learning models (Table

2) by achieving better performance for all classes. The non-Tamil classes are not recognized at all by any of the mentioned traditional ML models. However, the proposed deep learning model provides satisfactory prediction accuracy. The weighted average precision, recall and F1-score values are 0.54, 0.57, and 0.55, respectively on the validation dataset. However, on the test dataset, the weighted precision, recall and F1-score values are 0.544, 0.566, and 0.552, respectively.

**Table 3**

Hyper-parameters details for the proposed model

Hyper-parameters	Bi-LSTM model
Number of Bi-LSTM Units	64, 32, 32
Dropout rate	0.5
Activation functions	ReLU, Softmax
Epochs	50
Loss	Categorical Crossentropy
Optimizer	Adam

One of the possible reasons behind the biased performance of the model for different sentimental classes may include the inconsistent distribution of data samples in different classes of training and testing set (Table 1). The number of samples present in the Positive sentiment category is highest, whereas the lowest number of samples is present in not-Tamil category. The effect of this data distribution is seen on the model's outcomes (Table 4). Hence, to get better outcomes, data oversampling techniques such as SMOTE or ADASYN may help [19]. Another possible reason behind the low performance of the model may include the high number of code-mixed samples in training and testing dataset. By normalizing the dataset into English may help to achieve better prediction accuracy.

## 4. Conclusion

Sentiment analysis is one of the major research areas in computer science, where the opinion will be extracted from the input text. The opinion may be positive, negative or neutral. In the current time, users are popularly used mixed languages to post comments or reviews. Hence, getting the opinion from such a post is a challenging task. This research suggested a deep learning-based automated system to predict the sentiment of the user's post written in Tamil code-mixed. The proposed framework utilised the pre-trained word embedding technique and achieved a weighted F1-score of 0.552 for the best case on test sample.

## References

- [1] B. Liu, et al., Sentiment analysis and subjectivity., Handbook of natural language processing 2 (2010) 627–666.
- [2] R. Feldman, Techniques and applications for sentiment analysis, Communications of the ACM 56 (2013) 82–89.

**Table 4**

Results on Validation data with Bi-LSTM

Class	Precision	Recall	F1-score
Positive	0.68	0.80	0.74
Unknown State	0.34	0.31	0.32
Negative	0.41	0.30	0.35
Mixed feelings	0.20	0.11	0.14
not-Tamil	0.50	0.45	0.47
Weighted Avg	0.54	0.57	0.55

- [3] S. Saumya, J. P. Singh, Detection of spam reviews: a sentiment analysis approach, *Csi Transactions on ICT* 6 (2018) 137–148.
- [4] Q. T. Ain, M. Ali, A. Riaz, A. Noureen, M. Kamran, B. Hayat, A. Rehman, Sentiment analysis using deep learning techniques: a review, *Int J Adv Comput Sci Appl* 8 (2017) 424.
- [5] B. R. Chakravarthi, V. Muralidaran, R. Priyadharshini, J. P. McCrae, Corpus creation for sentiment analysis in code-mixed Tamil-English text, in: *Proceedings of the 1st Joint Workshop on Spoken Language Technologies for Under-resourced languages (SLTU) and Collaboration and Computing for Under-Resourced Languages (CCURL)*, European Language Resources association, Marseille, France, 2020, pp. 202–210. URL: <https://aclanthology.org/2020.sltu-1.28>.
- [6] A. Kumar, S. Saumya, J. P. Singh, Nitp-ai-nlp@ dravidian-codemix-fire2020: A hybrid cnn and bi-lstm network for sentiment analysis of dravidian code-mixed social media posts., in: *FIRE (Working Notes)*, 2020, pp. 582–590.
- [7] S. Mahata, D. Das, S. Bandyopadhyay, Sentiment classification of code-mixed tweets using bi-directional rnn and language tags, in: *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, 2021, pp. 28–35.
- [8] Y. Sharma, A. V. Mandalam, Bits2020@ dravidian-codemix-fire2020: Sub-word level sentiment analysis of dravidian code mixed data., in: *FIRE (Working Notes)*, 2020, pp. 503–509.
- [9] K. Goswami, P. Rani, B. R. Chakravarthi, T. Fransen, J. P. McCrae, Uld@ nuig at semeval-2020 task 9: Generative morphemes with an attention model for sentiment analysis in code-mixed text, *arXiv preprint arXiv:2008.01545* (2020).
- [10] S. Banerjee, A. Jayapal, S. Thavareesan, Nuig-shubhanker@dravidian-codemix- fire2020: Sentiment analysis of code-mixed dravidian text using xlnet, in: *FIRE*, 2020.
- [11] P. K. Roy, J. P. Singh, S. Banerjee, Deep learning to filter sms spam, *Future Generation Computer Systems* 102 (2020) 524–533.
- [12] S. Hochreiter, J. Schmidhuber, Long short-term memory, *Neural computation* 9 (1997) 1735–1780.
- [13] R. Priyadharshini, B. R. Chakravarthi, S. Thavareesan, D. Chinnappa, T. Durairaj, E. Sherly, Overview of the dravidiancodemix 2021 shared task on sentiment detection in tamil, malayalam, and kannada, in: *Forum for Information Retrieval Evaluation, FIRE 2021*, Association for Computing Machinery, 2021.
- [14] B. R. Chakravarthi, R. Priyadharshini, S. Thavareesan, D. Chinnappa, D. Thenmozhi, E. Sherly, J. P. McCrae, A. Hande, R. Ponnusamy, S. Banerjee, C. Vasantharajan, Findings

of the Sentiment Analysis of Dravidian Languages in Code-Mixed Text, in: Working Notes of FIRE 2021 - Forum for Information Retrieval Evaluation, CEUR, 2021.

- [15] B. R. Chakravarthi, V. Muralidaran, R. Priyadharshini, J. P. McCrae, Corpus creation for sentiment analysis in code-mixed tamil-english text, arXiv preprint arXiv:2006.00206 (2020).
- [16] R. Priyadharshini, B. R. Chakravarthi, S. Thavareesan, D. Chinnappa, T. Durairaj, E. Sherly, Overview of the dravidiancodemix 2021 shared task on sentiment detection in tamil, malayalam, and kannada, in: Forum for Information Retrieval Evaluation, FIRE 2021, Association for Computing Machinery, 2021.
- [17] J. Pennington, R. Socher, C. D. Manning, Glove: Global vectors for word representation, in: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), 2014, pp. 1532–1543.
- [18] P. K. Roy, Multilayer convolutional neural network to filter low quality content from quora, Neural Processing Letters 52 (2020) 805–821.
- [19] P. K. Roy, Z. Ahmad, J. P. Singh, M. A. A. Alryalat, N. P. Rana, Y. K. Dwivedi, Finding and ranking high-quality answers in community question answering sites, Global Journal of Flexible Systems Management 19 (2018) 53–68.