INTERNATIONAL CONFERENCE ON ADVANCES IN TECHNOLOGY AND COMPUTING (ICATC-2022)
Faculty of Computing and Technology (FCT), University of Kelaniya, Sri Lanka
3rd December 2022

Index No:CT-07-78

LSTM Based Emotion Analysis of Text in Tamil Language

M.R. Faiyaz Ahamed
Faculty of Computing and Technology
University of Kelaniya
Kelaniya, Sri Lanka
faiyazrafeek@gmail.com

S. P. Kasthuri Arachchi
Faculty of Computing and Technology
University of Kelaniya
Kelaniya, Sri Lanka
sandelik@kln.ac.lk

***Abstract*—The sentiments and emotions expressed by users on the internet greatly influence the decision-making process of business firms. Recent studies show that emotion analysis yields more precise information than sentiment analysis. Text emotion analysis has become popular for higher-demand languages like English, Chinese, French, and Arabic. However, no prior studies have been conducted on locally speaking languages, including Tamil, Malayalam, and Sinhala. Therefore, this paper presents a deep learning based novel model to identify the emotions expressed in Tamil texts using a Long Short-Term Memory (LSTM) network. Besides, to enhance the robustness of our proposed model, we conducted experiments with machine learning classifiers, including Support Vector Machine (SVM), Naïve Bayes (NB), Logistic Regression (LR), and Random Forest Classifier (RFC). The experimental results prove that our Tamil text emotion analysis model significantly outperforms other machine learning models, achieving an accuracy of 80%.**

Keywords— Sentiment Analysis, Emotion Analysis, Machine Learning, Recurrent Neural Network, LSTM

I. INTRODUCTION

Millions of individuals use microblogging platforms to express their thoughts, feelings, emotions, and experiences and online purchasing platforms to express their reviews on products or services. Individuals and business organizations are always keen to know what others think of them and their feelings about their acts, performance, products, or services. These public opinions can help them change their behavior, improve performance, modify their services, create new products or services, and so forth.

Natural Language Processing (NLP) is used in various applications, including voice recognition, text categorization, knowledge discovery, and computational linguistics [1]. Sentiment Analysis (SA) and Emotion Analysis (EA) are the key aspects of NLP. Although these terms are commonly used synonymously, they differ from each other. The process of detecting whether the given input is positive, negative, or neutral is called "sentiment analysis." Whereas "emotion analysis" recognizes unique human emotions such as anger, fear, joy, surprise, or depression. Studies stated that people's reactions to events or situations that are important to them are defined by their emotions [2]. *Emotions that can be expressed include happiness, sadness, fear, and anger*. Emotion models serve as the cornerstone for emotion detection systems, defining how emotions are expressed.

Sentiment Analysis and Emotion Analysis can help product companies increase their revenues and client retention by studying the public perceptions of their products and services. It also aids in the development of more engaging and effective marketing campaigns by predicting customer trends. SA is a widely utilized indicator for stock market forecasting [3]. The views about a firm on social media are closely associated with the company's stock price rises and decreases. Consequently, based on social media comments, one may forecast whether the company will earn a profit or a loss in the future and if it is worthwhile to invest in its shares.

According to [4], Existing research has given a variety of approaches for Sentiment Analysis since data were abundant. As demonstrated in [5], some emotion analysis studies were only carried out for globally speaking languages like English, Chinese, and Spanish. Tamil has more than 92 million speakers globally, making it the 17th most-spoken language in the world [6]. Thus, analyzing online Tamil material will yield information about significant commercial, political, and social topics.

Prior research has yet to be done to develop a proper model for text emotion analysis in Tamil. Therefore, there is an unavailability of a proper pre-trained model to analyze the emotion of text in the Tamil language to understand the different emotions. One of the main reasons for the model's unavailability is the need for an adequately annotated corpus. However, developing a state-of-the-art deep learning model for the emotional analysis of text in the Tamil language will be possible using the recently created annotated corpus by Jenarthanan et al. [7]. A deep-learning-based LSTM model was used to build the model to detect the emotions expressed in Tamil. Traditional machine learning models such as NB, SVM, and RFC are also used for comparison purposes. The objectives of the study are as follows,

- We comprehensively review numerous text classification methods used in Emotion Analysis and Sentiment Analysis.
- We propose a deep learning-based state-of-the-art model that can effectively identify the emotions expressed in Tamil text.

The rest of the manuscript is structured as follows. Chapter 2 mentions the existing studies and related studies reviewed related to study and background. A detailed description of the proposed methodology for detecting and analyzing emotion in the text is illustrated in Chapter 3. Chapter 4 depicts the complete empirical results of the study using different approaches and the discussion. Finally, the paper concludes in Chapter 5. This study's recommendations and future directions are also included in Chapter 5.

II. Related Work

A. Sentiment Analysis

There are several ways a Sentiment analysis can be performed, such as document, sentence, and aspect levels [8]. The document-level classifier classifies an interpretive document as a positive or negative viewpoint. Several supervised machine learning algorithms were used to classify the documents for news comments. Girish et al. [9] combined the NB and Neural Network algorithm to classify viewers' movie reviews. They demonstrated that combining these two strategies boosted sentiment analysis accuracy by up to 80.65%. The most fine-grained analysis of the document is sentiment analysis at the sentence level. Each sentence is viewed as a separate unit with its collection of opinions. According to [10], feature extraction is a fundamental and necessary process for Sentiment Analysis. Numerous studies have been conducted utilizing classifiers such as SVM, NB, and Decision trees (DT) and classified them by combining word embedding to produce accurate results. Da Silva et al. [11] presented hybrid models that achieved a novel accuracy of 81.06%, including a blend of SVM, NB, DT, etc. Araque et al. [12] merge standard aspect models based on guided feature detection with deep neural network methods for sentiment classification utilizing word2vec.

B. Emotion Analysis

Detecting "emotion" from text is more complicated than detecting "sentiment." Although these two phrases are commonly used interchangeably, they have different definitions in terms of computing. Sentiment analysis refers to deriving personal textual data to determine a person's polarity of attitude toward another person, object, situation, or action. On the other hand, emotion detection determines how a person feels about a particular event, person, or object using predefined emotion models based on psychological emotion theories.

Seol et al. [13] used neural networks to achieve 45% to 65% per-class accuracy for an eight-class categorization. Hasan et al. [14] proposed traditional ML approaches like NB, SVM, and DT to recognize emotions from 135,000 processed tagged texts from Twitter. Recently advanced neural networks, such as deep learning algorithms, have been employed to solve the emotion analysis task. Convolution Neural Networks (CNN), LSTM models surpass classic machine learning algorithms on text classifying tasks, including sentiment analysis, emotion detection, and attitude recognition. Mansur et al. [15] have developed three deep neural network architectures: CNN, ANN, and LSTM. Experimental results prove that the Deep learning models outperform classical machine learning algorithms for Turkish text emotion detection. Another method aims at understanding implemented based on emotional contexts by pre-training neural models. A Bidirectional Encoder Representation from Transformers (BERT) based model was recently proposed [16]. This pre-trained BERT model delivered surface quality throughout various of NLP applications, particularly emotion analysis, despite incurring additional work structure changes.

C. Traditional Emotion Classification Approaches

1) Keyword-based Approach

The most straightforward method is to detect emotions using a keyword-based approach. The goal is to match patterns that are comparable to emotion keywords. The first challenge is finding the phrase that most accurately conveys the emotion. This is usually performed by tagging the words in a phrase with a Part of Speech (POS) tagger and retrieving the Verb, Noun, Adverb, and Adjective terms. Researchers usually build keyword dictionaries based on emotions and phrases. Internet tools and programs, such as WordNet [17], can help you identify synonyms and antonyms for terms to use in your lexicon.

2) Lexical-based Approach

This approach, also known as the keyword-based analysis, looks at emotional terms associated with specific emotional factors. WordNet-Affect and the NRC word–emotion lexicon [18] are two standard lexicons for emotion recognition. Balahur et al. [19] detect emotion using EmotiNet, a resource for detecting emotion from text based on commonsense knowledge of ideas, interactions, and emotional consequences. Choudhury et al. [20] discovered a vocabulary of over 200 emotions often seen in Tweets.

3) Machine Learning-based approach

Both unsupervised and supervised machine learning techniques were utilized for textual emotion identification. Different types of machine learning algorithms, including NB, SVM, DT, and others, may be required for this approach. Pang et al. [21] examined the effectiveness of NB, maximum entropy, and SVM classification in online movie reviews utilizing online text emotion recognition. The study found that SVMs outperformed other approaches in terms of accuracy. Machine learning-based emotion analysis has the advantage of modeling many features.

D. Deep Learning-based Emotion Classification Approaches

In recent years, deep learning algorithms are made tremendous breakthroughs in machine vision and voice recognition. Now DL has also been used in NLP applications such as word embedding and text training. Multiple layers of neurons are used in deep learning models. Thousands of neurons are connected, allowing for parallel processing to occur. Bengio et al. [22] suggested a neural network-based language model estimation architecture that merged the learning of word embedding modeling and analytical language models using a feedforward neural network with different hidden layers.

Chatterjee et al. [23] built a sentiment and linguistic emotion recognition model by injecting sentiment and linguistic interpretations into two LSTM layers. After that, the models are merged and sent to a complex system for categorization purposes. Skip-Gram was introduced by Mikolov et al. [24], and it is an excellent method of training elevated distributed word vectors. Skip-Gram could begin by detecting nearby keywords in a text Glorot et al. [25] suggested a neural network strategy for solving the transfer learning constraint in emotion classifications and used it to increase the accuracy of emotion detection in bulk online reviews. Most of the available continuous word representation algorithms describe the grammatical surroundings of terms while ignoring the emotional content of the text. Tang et al. [26] solved this question by training sentiment-specific word representations, which capture semantic meaning in continual word representations.

CNN uses convolution filters to learn local characteristics. CNNs were first employed in computer vision

and have since been used in NLP, yielding positive outcomes in text summarization, relevant search recovery, sentence structuring, and other traditional NLP applications. using two-hybrid models, Xu et al. [27] retrieved emotional features from video and text. Social media sites have recently been overwhelmed with posts about covid 19. Singh et al. [28] achieved an average accuracy of 94% using emotion recognition analysis on covid-related tweets collected globally and in India using only the BERT model with the Twitter dataset.

E. Text Classification in the South Indian Language

While most text classification research has been conducted for English, Chinese, French, Arabic, Indian languages, including Telugu, Tamil, Hindi, Bengali, and others, contributed significantly less. The unavailability of labeled data, grammatical and morphological resources, and techniques for Indian languages is the primary cause. The study of[29] investigates with the sentiment analysis techniques and other resources available for Indian languages to apply sentiment analysis. The usage of Indian languages is increasing rapidly on different social media platforms. Therefore, it is essential to build resources for low-resource Indian languages. Singapore, Tamil Nadu, an Indian city, and Sri Lanka have Tamil as the official language. In their daily lives, about 100 million individuals use these languages. This work is beneficial because they are widely spoken languages in South Asian countries.

Arunselvan et al. [30] utilized machine learning classifiers like NB, LR, and SVM for SA on movie reviews. Using bigrams, LR achieved 61.88% and 57.14% accuracy, respectively. For unigram features, MNB and BNB achieved an accuracy of 61.01 percent and 60.19%, respectively. Shriya et al. [31] aimed to use SVM, Maximum Entropy classifier (Maxent), Decision tree, and NB to create a model that could predict sentiments from Tamil movie reviews. The SVM had a 75.96% accuracy, whereas the Decision Tree had a 66.29%. When we consider the text classification studies in deep learning for the Tamil language, plenty of researchers only created the corpus for the studies. However, there were some studies carried out. Padmamla et al. [32] proposed a technique utilizing Neural Network models by increasing the precision of a sentiment gauger for the Tamil texts. The accuracy yield from the NN models is 73.2%.

Similarly, Balouchzahi et al. [33] presented machine learning and deep learning models for Sentiments Analysis of code-mixed Malayalam and Tamil texts. Anbukkarasi et al. [34] proposed Bi-LSTM and LSTM models to analyze Tamil tweets. In this paper [35] by using the Waikato environment for knowledge analysis (WEKA), authors have classified the roman Urdu reviews associated with automobiles. Further, in [36] this paper, they applied different machine learning techniques which are used to analyze the sentiment of the product reviews on Twitter. They have employed the Naive Bayes technique to obtain better results. Increment in the training data to improve the identification process and get the quality summary of the reviews.

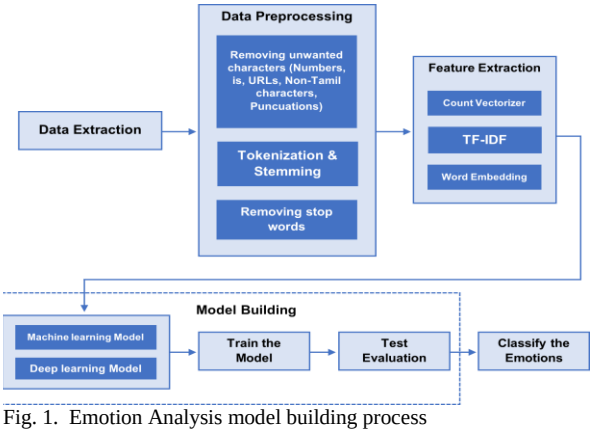


Fig. 1. Emotion Analysis model building process

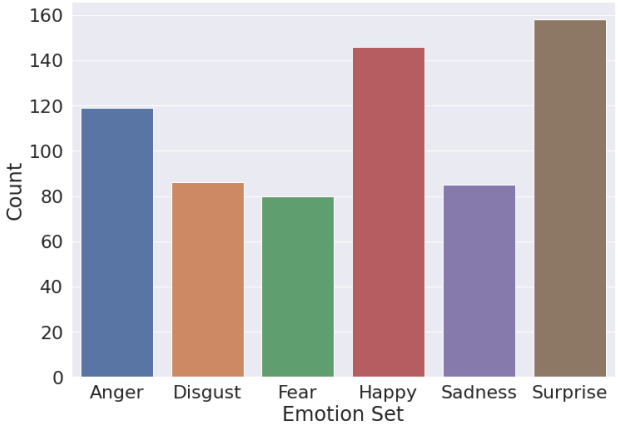


Fig. 2. Data distribution against six basic emotions

III. METHODOLOGY

A. Data Collection

Researchers have utilized several sentiment and emotion analysis datasets to evaluate the effectiveness of their algorithms. Twitter has now become a trendy social media or microblogging platform. The posts are referred to as tweets, and they are public. For this research, Annotated Corpus for Tamil & Sinhala Emotion Analysis (ACTSEA) was utilized as the primary dataset since it is the most extensive data collection currently existing for the Tamil language [7]. It categorized core emotional terms into six class categories: disgust, anger, fear, surprise, joy, and sadness. ACTSEA was collected from the Twitter network and properly annotated after cleaning. It consists of 600280 (Tamil) and 318308 (Sinhala) tweets. Finally, their corpus contains 1962 tweets that were correctly and incorrectly labeled. The study [7] used 625 properly annotated Tamil tweets from the ACTSEA corpus. Figure 1 represents the following steps to build the models, and Figure 2 shows the data distribution against six basic emotions.

B. Data Pre-processing

Cleaning and preparing data serves an essential part in accurately classifying any text. Preprocessing is the initial stage in NLP before analyzing or classifying it. During this study, removing punctuations, removing non-Tamil characters and digits, tokenization, stemming and

lemmatization, and removing stop words are utilized as preprocessing steps.

1) Removing Punctuations

After preprocessing, any punctuations (!,",#,$,%, &,',(,),*,+,,,-,.,/,:,;,,=,>,?,@,[,,],,_,',,|,,) will be deleted from the text. Using the Python string module, punctuations will be easily removed.

2) Removing Non-Tamil characters and digits

Non-Tamil characters like English alphabetic letters and other alphanumerics will be removed after the preprocessing step since these characters may reduce the model's accuracy.

3) Tokenization

Tokenization is breaking each sentence or document into chunks of words that may then be categorized. Consider the statement "நான் நாளை மைதானத்திற்கு விளையாடப் போகிறேன் (I'm going to play on the playground tomorrow)" which will become "நான்," "நாளை," "மைதானத்திற்கு," "விளையாடப்," "போகிறேன்" after tokenization. Normalize the material to ensure data consistency by converting it to simplified form and correcting spelling mistakes.

4) Stemming and Lemmatization

Each word is transformed, lowering its inflectional forms to a common root word to decrease the number of linguistic terms. The key distinction between lemmatization and stemming is that stemming extract a word's root by truncating suffixes, whereas lemmatization removes inflectional ends from a token and converts them to the base word lemma through morphological analysis.

5) Removing stop words

Stop words, such as articles, prepositions, and conjunctions, are used in a language. These words are not necessary for the meaning of the sentence. The NLTK library in Python can be used to remove stop words. The NLTK package supports various languages, including English, French, German, Finnish, Italian, and others, but not Tamil. Commonly used stop words in the Tamil language such as, "," "," "," ","were downloaded from GitHub and manually included in the NLTK library.

C. Feature Extraction

Text is translated into numerical terms by the machine. The technique of translating or converting texts into numerical vectors is known as word embedding or word representation. Preprocessed data were converted so that machine learning algorithms could use them. As stated in previous sections C.1 and D, numerous feature extraction techniques exist in the NLP domain. The bag of word model could be used as a benchmark against which other complex procedures can be compared [37]. A fixed-length count vector is defined, with each entry corresponding to a word from a pre-specified lexicon of terms. The Ngram model [37] directly extends the bag of words model, which resolves word order in sentence vector representation. Another prominent method for feature extraction is the Term Frequency Inverse Document Frequency (TFIDF) [26]. This approach encodes text as a matrix, with each number representing the volume of information contained in every term in a text input. As the name implies, it evaluates the recurrence of terms across documents and applies a lowering factor to common words.

A deep learning network can produce effective vector representations due to the availability of large datasets. More meaningful Feature extraction can be achieved using neural networks-based word embedding. Similar numerical vectors in a neural network-based word embedding represent similar or connected words. Because it preserves the semantics of words, this is commonly used in word guessing. The most widely used word embedding methodology is the Word2Vec model [38], which is based on statistical methodologies and deep learning techniques. GloVe, created by Stanford University researchers [38], and FastText [39], developed by Facebook, are two more deep learning-based word embedding methods. Word2vec vectors take longer to train than GloVe vectors. By multiple other criteria, FastText vectors are more accurate than Word2Vec vectors [38].

D. Classification Algorithms

1) Machine Learning Algorithms

Naïve Bayes (NB) is the traditional starting point algorithm for various machine learning contexts [8]. The NB algorithm is based on the Bayes theorem and belongs to the family of probabilistic classifiers. It takes the training dataset's probability distribution and assumes that all words are mutually independent. We utilized a Multinomial NB model developed in Python's sklearn library for our experiments. SVMs are a set of classifiers that perform well in NLP and use the theory of a decision plane or hyperplane in their work. Besides, it attempts to locate a hyperplane that splits data from two classes further apart than possible. Thus, we have used SVM classifiers for the experiment and model building.

A logistic regression technique can predict the likelihood of response depending on predictor factors in which more independent variables control the results. We have used the LR classifier to compare the output model. A Random Forest (RF) is an advancement of Decision Trees. RF builds many decision trees and examines their grades when determining the final result. It can be used for multi-class classification and easily combined with other techniques. We also utilized the RFC technique to assess the model.

2) Deep Learning algorithms

In recent years deep learning has developed as a robust pattern recognition and language processing technique in recent years. It is gaining popularity due to its capacity for automatic feature engineering and significant accuracy. Most modern deep learning models are based on numerous processing layers that learn data representations at various degrees of abstraction. While feedforward networks may recognize the next term in a series, primary Recurrent Neural Networks (RNN) could also recognize all preceding terms.

LSTM, an extension of RNN, consists of both feedback and feedforward connections. LSTM cells, unlike other RNN models, have their memory to store information for long sequences of tokens. A basic LSTM architecture comprises a memory cell, an input gate, an output gate, and a forget gate as the components [4]. The cell maintains a value (or state) for either a long or short time period. It is accomplished by exploiting the memory cell's activation function. The input to the model for text classification is a sequence of embedding vectors using our custom word embedding. The RNN-LSTM model was implemented using the Python TensorFlow library.

IV. Results and Discussion

The experiments were carried out in stages. We preprocessed the text after obtaining the data. Then, we used machine learning and deep learning classifiers to classify the text according to the six emotion classes: *happiness, fear, disgust, anger, surprise, and sadness*. We evaluated the performance results of four machine learning algorithms, NB, SVM, LR, RFC, and a Neural Network architecture, LSTM, to assess and compare the developed model.

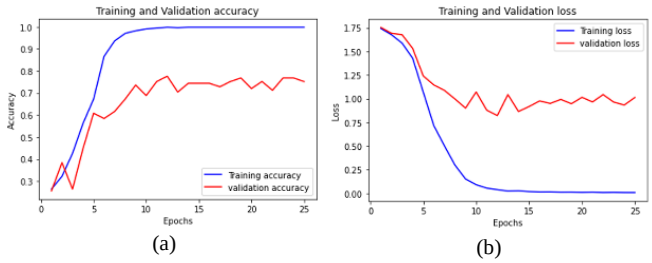
We experimented with two levels. First, using different feature extraction techniques, four traditional ML classifiers, including NB, SVM, LR, and RFC. The accuracy, precision, and recall F1 of each machine learning algorithm are measured, and this information is used to calculate the method's efficiency. We can deduce which algorithm works well in Tamil emotion analysis of texts based on these metrics by evaluating its overall accuracy.

Table 1 represents the accuracy results from the experiments using various feature extraction techniques, Count Vectorizer (CV), TFIDF, and Word Embedding. Second, we experimented with the LSTM model. We can see that the LSTM model achieved 0.75 (75%) accuracy. LSTM parameters such as dropout, loss function, optimizer, activation function, and a number of epochs were used in numerous trials. We used 'softmax' as the activation function and 'sparse categorical cross entropy' as the loss function with a dropout of 0.2 for 25 epochs. First, we achieved an accuracy of 75%, as seen in Figure 3. The training and testing splits are 80% and 20% of the total data, respectively. It can be seen that the model is 'overfitting, ' as the gap between two lines in both accuracy and the loss curve.

TABLE I. Accuracies of Experimented Models

Model	Accuracy (%)	
	CV	TFIDF
Multinomial Naïve Bayes	58	56
Linear Support Vector Machines	64	62
Logistic Regression	67	64
Random Forest Classifier	65	66
	Word Embedding	
LSTM (Before fine-tuning)	75	
LSTM (After fine-tuning)	80	

According to multiple experiments, the model's accuracy is influenced by the dropout value. Then, the dropout value changed from 0.2, 0.4 to 0.8, and achieved an accuracy of 80%, as seen in Figure 4. The experiments indicated that LSTM outperforms all other models in emotion analysis tasks, with a higher accuracy rate, indicating that this deep learning model can deal with text emotion analysis.

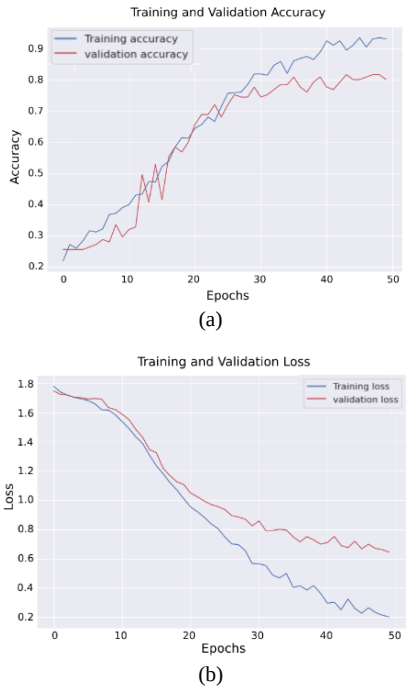


(a) (b)

Fig. 3. Performance of the proposed model with first experiment (a) training and validation accuracy (b) training and validation loss

V. Conclusion and Future Work

Analyzing the emotions expressed in the text significantly influences the provision of precise information, which can be helpful for the decision-making process. Hence, text emotion analysis for the Tamil language was a challenging task due to the unavailability of the corpus and a proper model. In this research, we present a deep learning-based state-of-the-art model which can effectively detect emotion from texts in the Tamil language. We utilized the ACTSEA corpus [7] as the dataset. After collecting the data, the preprocessing steps were carried out in different steps to clean the dataset for the feature extraction process. Besides, different vectorization techniques, such as Count Vectorizer (CV) and (Term Frequency Inverse Document Frequency) TF-IDF, were utilized for feature extraction. We trained and built the neural network model called LSTM. Then the proposed model was compared with traditional machine learning algorithms such as Naïve Bayes (NB), Support Vector Machines (SVM), Logistic Regression (LR), and Random Forest Classifiers (RFC). Experiment results show that the deep learning model outperformed all the traditional machine learning models. The highest accuracy of 80% has been obtained using the LSTM model. The accuracy of 58%, 64%, 67%, and 66% were obtained using appropriate feature extraction techniques by Naïve Bayes (NB), Support Vector Machines (SVM), Logistic Regression (LR), and Random Forest Classifiers (RFC)models, respectively. Finally, we conclude that a deep convolutional neural network called LSTM with word embedding vectors performs well in the emotional analysis of Tamil texts.



(a) (b)

Fig. 4. Performance of the proposed model with final experiment (a) training and validation accuracy (b) training and validation loss

This approach is the initial step toward the text-based emotion analysis of the Tamil language. The present study described in the literature suggests numerous possibilities for further analysis, which might be examined for future studies. Develop different word embedding methodologies and word embedding language feature combinations in the future. The data set should be increased to improve accuracy. In

conclusion, without requiring extensive task-specific architectural alterations, the pre-trained transformer-based BERT model performed at the cutting edge of numerous NLP applications, including emotion analysis. As for future works, we plan to work on this transformer-based BERT architecture to achieve more precise results with higher accuracy than the current proposed model.

REFERENCES

[1] M. Itani, C. Roast, and S. Al-Khayatt, "Developing Resources for Sentiment Analysis of Informal Arabic Text in Social Media," in Procedia Computer Science, 2017, vol. 117, pp. 129–136, doi: 10.1016/j.procs.2017.10.101.

[2] C. E. Izard, "Emotion theory and research: Highlights, unanswered questions, and emerging issues," Annual Review of Psychology, vol. 60. NIH Public Access, pp. 1–25, Jan. 2009, doi: 10.1146/annurev.psych.60.110707.163539.

[3] R. Gupta and M. Chen, "Sentiment Analysis for Stock Price Prediction," in Proceedings - 3rd International Conference on Multimedia Information Processing and Retrieval, MIPR 2020, 2020, pp. 213–218, doi: 10.1109/MIPR49039.2020.00051.

[4] R. Bhargava, S. Arora, and Y. Sharma, "Neural network-based architecture for sentiment analysis in Indian languages," J. Intell. Syst., vol. 28, no. 3, pp. 361–375, 2019, doi: 10.1515/jisys-2017-0398.

[5] G. Thripathi, "5 Ways the Internet of Things is Changing the Game for Education and Learning," IOT for All, 2020. https://www.iotforall.com/5-ways-iot-changes-education (accessed Sep. 30, 2021).

[6] A. P. D'Costa, "Top 30 Language Spoken in the World by Number of Speakers," Vistawide.Com, 2006. https://www.vistawide.com/languages/top_30_languages.htm (accessed Feb. 10, 2022).

[7] R. Jenarthanan, Y. Senarath, and U. Thayasivam, "ACTSEA: Annotated Corpus for Tamil Sinhala Emotion Analysis," in MERCon 2019 - Proceedings, 5th International Multidisciplinary Moratuwa Engineering Research Conference, Jul. 2019, pp. 49–53, doi: 10.1109/MERCon.2019.8818760.

[8] W. Medhat, A. Hassan, and H. Korashy, "Sentiment analysis algorithms and applications: A survey," Ain Shams Eng. J., vol. 5, no. 4, pp. 1093–1113, 2014, doi: 10.1016/j.asej.2014.04.011.

[9] L. L. Dhande and G. K. Patnaik, "Analyzing Sentiment of Movie Review Data using Naive Bayes Neural Classifier," Int. J. Emerg. Trends Technol. Comput. Sci., vol. 3, no. 4, pp. 313–320, 2014.

[10] F. Hemmatian and M. K. Sohrabi, "A survey on classification techniques for opinion mining and sentiment analysis," Artif. Intell. Rev., vol. 52, no. 3, pp. 1495–1545, 2019, doi: 10.1007/s10462-017-9599-6.

[11] N. F. F. Da Silva, E. R. Hruschka, and E. R. Hruschka, "Tweet sentiment analysis with classifier ensembles," Decis. Support Syst., vol. 66, 2014, doi: 10.1016/j.dss.2014.07.003.

[12] aque, I. Corcuera-Platas, J. F. Sánchez-Rada, and C. A. Iglesias, "Enhancing deep learning sentiment analysis with ensemble techniques in social applications," Expert Syst. Appl., vol. 77, pp. 236–246, 2017, doi: 10.1016/j.eswa.2017.02.002.

[13] Y.-S. Seol, D.-J. Kim, and H.-W. Kim, "Emotion Recognition from Text Using Knowledge-based ANN," 23rd Int. Tech. Circuits/Systems, Comput. Commun., 2008, Accessed: Feb. 14, 2022. [Online]. Available: http://www.ieice.org/proceedings/ITC-CSCC2008/pdf/p1569_P2-43.pdf.

[14] A. Abdi, S. M. Shamsuddin, S. Hasan, and J. Piran, "Deep learning-based sentiment classification of evaluative text based on Multi-feature fusion," Inf. Process. Manag., vol. 56, no. 4, pp. 1245–1259, 2019, doi: 10.1016/j.ipm.2019.02.018.

[15] M. A. Tocoglu, O. Ozturkmenoglu, and A. Alpkocak, "Emotion Analysis from Turkish Tweets Using Deep Neural Networks," IEEE Access, vol. 7, pp. 183061–183069, 2019, doi: 10.1109/ACCESS.2019.2960113.

[16] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference, 2019, vol. 1.

[17] T. Wei, Y. Lu, H. Chang, Q. Zhou, and X. Bao, "A semantic approach for text clustering using WordNet and lexical chains," Expert Syst. Appl., vol. 42, no. 4, pp. 2264–2275, 2015, doi: 10.1016/j.eswa.2014.10.023.

[18] S. M. Mohammad and P. D. Turney, "Crowdsourcing a word-emotion association lexicon," in Computational Intelligence, 2013, vol. 29, no. 3, pp. 436–465, doi: 10.1111/j.1467-8640.2012.00460.x.

[19] A. Balahur, J. M. Hermida, and A. Montoyo, "Detecting implicit expressions of emotion in text: A comparative analysis," in Decision Support Systems, 2012, vol. 53, no. 4, pp. 742–753, doi: 10.1016/j.dss.2012.05.024.

[20] M. De Choudhury, S. Counts, and M. Gamon, "Not all moods are created equal! Exploring human emotional states in social media," in ICWSM 2012 - Proceedings of the 6th International AAAI Conference on Weblogs and Social Media, 2012, pp. 66–73.

[21] B. Pang and L. Lee, "Opinion mining and sentiment analysis," Found. Trends Inf. Retr., vol. 2, no. 1–2, pp. 1–135, 2008, doi: 10.1561/1500000011.

[22] Y. Bengio, R. Ducharme, P. Vincent, and C. Jauvin, "A Neural Probabilistic Language Model," in Journal of Machine Learning Research, 2003, vol. 3, no. 6, pp. 1137–1155, doi: 10.1162/153244303322533223.

[23] A. Chatterjee, U. Gupta, M. K. Chinnakotla, R. Srikanth, M. Galley, and P. Agrawal, "Understanding Emotions in Text Using Deep Learning and Big Data," Comput. Human Behav., vol. 93, pp. 309–317, 2019, doi: 10.1016/j.chb.2018.12.029.

[24] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," 2013.

[25] X. Glorot, A. Bordes, and Y. Bengio, "Domain adaptation for large-scale sentiment classification: A deep learning approach," in Proceedings of the 28th International Conference on Machine Learning, ICML 2011, 2011, pp. 513–520.

[26] D. Tang, F. Wei, N. Yang, M. Zhou, T. Liu, and B. Qin, "Learning sentiment-specific word embedding for twitter sentiment classification," in 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014 - Proceedings of the Conference, 2014, vol. 1, pp. 1555–1565, doi: 10.3115/v1/p14-1146.

[27] G. Xu, W. Li, and J. Liu, "A social emotion classification approach using multi-model fusion," Futur. Gener. Comput. Syst., vol. 102, pp. 347–356, 2020, doi: 10.1016/j.future.2019.07.007.

[28] M. Singh, A. K. Jakhar, and S. Pandey, "Sentiment analysis on the impact of coronavirus in social life using the BERT model," Soc. Netw. Anal. Min., vol. 11, no. 1, 2021, doi: 10.1007/s13278-021-00737-z.

[29] M. Shelke Babasaheb Ambedkar, S. Deshmukh, M. B. Shelke, and S. N. Deshmukh, "Recent Advances in Sentiment Analysis of Indian Languages Article in International Journal of Future Generation Communication and Networking," International Journal of Future Generation Communication and Networking, vol. 13, no. 4, pp. 1656–1675, 2020, Accessed: Nov. 17, 2022. [Online]. Available: https://www.researchgate.net/publication/345240744

[30] S. J. Arunselvan, M. Anand Kumar, and K. P. Soman, "Sentiment analysis of tamil movie reviews via feature frequency count," Int. J. Appl. Eng. Res., vol. 10, no. 20, pp. 17934–17939, 2015.

[31] S. Shriya, R. Vinayakumar, M. Anand Kumar, and K. P. Soman, "Predicting the Sentimental Reviews in Tamil Movie using Machine Learning Algorithms," Indian J. Sci. Technol., vol. 9, no. 45, 2016, doi: 10.17485/ijst/2016/v9i45/106482.

[32] R. Padmamala and V. Prema, "Sentiment analysis of online Tamil contents using recursive neural network models approach for Tamil language," in 2017 IEEE International Conference on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials, ICSTM 2017 - Proceedings, 2017, pp. 28–31, doi: 10.1109/ICSTM.2017.8089122.

[33] F. Balouchzahi and H. L. Shashirekha, "LA-SACo: A Study of Learning Approaches for Sentiments Analysis inCode-Mixing Texts," 2021.

[34] S. Anbukkarasi and S. Varadhaganapathy, "SA-SVG@Dravidian-CodeMix-FIRE2020: Deep learning based sentiment analysis in code-mixed tamil-english text," in CEUR Workshop Proceedings, 2020, vol. 2826.

[35] M. Khan and K. Malik, "Sentiment classification of customer's reviews about automobiles in Roman Urdu," Advances in Intelligent Systems and Computing, vol. 887, pp. 630–640, 2019, doi: 10.1007/978-3-030-03405-4_44/COVER.

[36] N. Chintalapudi, G. Battineni, M. di Canio, G. G. Sagaro, and F. Amenta, "Text mining with sentiment analysis on seafarers' medical documents," International Journal of Information Management Data Insights, vol. 1, no. 1, p. 100005, Apr. 2021, doi: 10.1016/J.JJIMEI.2020.100005.

[37] A. K. Yadav and S. K. Borgohain, "Sentence generation from a bag of words using N-gram model," Proceedings of 2014 IEEE International Conference on Advanced Communication, Control and Computing Technologies, ICACCCT 2014, pp. 1771–1776, Jan. 2015, doi: 10.1109/ICACCCT.2014.7019414.

[38] J. Pennington, R. Socher, and C. D. Manning, "GloVe: Global vectors for word representation," in EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference, 2014, pp. 1532–1543, doi: 10.3115/v1/d14-1162.

[39] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching Word Vectors with Subword Information", doi: 10.1162/tacl_a_00051.