JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 1, 2022, p. 473-482
https://publishoa.com
ISSN: 1309-3452

A Pragmatic Approach to Emoji based Multimodal Sentiment Analysis using Deep Neural Networks

T Praveen Kumar

Department of CSE
Methodist College of Engineering and Technology
Hyderabad, Telangana, India
pravinthumukunta@gmail.com

B.Vishnu Vardhan

Department of CSE
JNTUH College of Manthani
Manthani, Telangana, India.
mailvishnuvardhan@gmail.com

Abstract

The Opinions of the customers regarding products have also become an important parameter for sales. The manufacturing companies are also continuously monitoring the feedback given on social media sites about their products, especially mobile reviews. The Sentiment Analysis (SA) is playing a vital role. The analysis cannot be limited to only text categorization as positive, negative, or neutral. The Emojis are also capturing emotions. So, in our proposed work the multi-modal sentiment analysis is done using text and Emojis. And the malleability of Deep learning models on the text has also increased. The combination of Word embedding models CBOW and SG are combined with the deep learning classifiers like LSTM, CNN, Bi-LSTM, and CNN-LSTM. The novelty of this work is to develop an Emoji-Based sentiment lexicon and cosine similarity usage for finding similarity. These were all modeled to predict the emotions in new mobile product reviews collected from various social media sites. The evaluation parameters proved that our proposed work had better results. The CNN-LSTM model topped in the accuracy of 94.94%.

Keywords: Word2Vec-CBOW, Word2Vec-SkipGram, LSTM, CNN, Bi-LSTM

1. Introduction

The success of the company and product depends on the customers. The customers' review of a particular product helps in increasing or decreasing the sales. Sentiments are one of the best features for analyzing customers' responses [13]. SA can be defined as a process of computationally identifying and categorizing opinions from a piece of review or voice message and determining whether the writer's attitude towards a particular product is favorable or adversarial or nonpartisan. The input for SA is either collected as text or emojis together from comments on Mobile apps, LinkedIn posts, social media posts, or Facebook shares. These millions of reviews/posts are mostly in the form of texts and emojis or only emojis to express emotion. These things on a single product will help the companies identify its pros and cons in a simpler manner using SA.

The SA combines the two languages namely Natural Language Processing (NLP) and Deep Learning. The NLP is a language that transforms human language into something which machines can understand. The syntactic techniques and semantic techniques are used for processing the text. Now the processed text is ready for classification by deep learning algorithms. These algorithms help in making predictions based on the patterns. These algorithms are not dependent on the explicit instructions but the sample data i.e., trained data. A model is built that will analyze sentiments based on emotions. These emotions are either only text, text, and Emojis or only Emojis. This will reduce the manual overhead and error on the bias [14]. Conventional sentiment analysis involves using reference dictionaries to determine how positive specific keywords are, then combining these scores to get the text's emotion.

Recent studies suggest that the SA is also done using Deep Learning algorithms that achieve remarkable results. In this work, we propose a SA technique which is analyzing emotions based on Text and Emojis. The Emojis were 76 in the year 1995 and have increased to 3663 by 2022. The 😭 (loud crying face) is considered to highest used in the year 2021[1]. The accuracy of SA increases by including Emojis. They represent unique sentiments and emotions embedded in them. So, our proposed work includes SA with both text and Emojis using Deep learning algorithms. We also propose to generate embedding for Emojis with similar words around them. The DL algorithms accept the numeric data. So, in this paper, we used word embeddings like Word2Vec to convert product reviews into a vector form and four Deep Learning classifiers are used [16] [20].

The proposed research work is organized in the following manner. Related Literature Survey is in section-2, the Solution for the proposed problem in Section 3 and in Section 4 the Sentiment Analysis of Product Reviews. Finally, the Results of our Experiments in Section-5.

2. Literature Survey

The proposed model in our research draws inspiration from the existing works of SA, Word Embeddings using Emojis, and Deep Learning models. There is a lot of literature on machine learning algorithms especially SVM classifier and Random Forest etc[14][15]. The feature extraction like TF-IDF, Count Vectorizer, etc. Some of the recent works are discussed here.

Jagadeesh Panthati et.al. [5] proposed an approach to carry for SA of product reviews. They have used the deep learning classifier CNN. The novelty of this model was to use two-word embeddings and the classifier CNN with frameworks of TensorFlow and Keras for training and classification [7]. This model is used to capture the emotions of new product reviews. The experiments have improved the accuracy and also concluded that the accuracy increases proportionally with the size of the dataset. But it was confined to a static dataset.

Ranjan Bhowmik et. al. [6] proposed Deep Learning models for the Sentiment Analysis of Bangla text. They have used an extended lexicon data dictionary (LDD) in this work. In Bangala text to identify the sentiment score (SS) the rule-based methods are used and the DL models. The cricket reviews were considered as a dataset. The polarity was defined by the BTSC algorithm and implemented on various models of CNN and LSTM [8]. The performance was good when compared with evaluators like F1-score, sensitivity, and recall. The hybrid model was proposed which combines LSTM and BERT which is having high accuracy compared with the individual models. This work was concluded on a benchmark dataset of Bangala reviews.

Zhenpeng Chen et. al. [9] proposed a novel method to predict sentiments using emojis. They have used Emojis as an instrument. The ELSA algorithm proposed helps in learning both cross-language[10] and different languages sentiments. Experiments were carried out on a variety of benchmark datasets. Even as the size of labeled and unlabeled data diminishes, it beats state-of-the-art cross-border approaches. The positive results suggest that emojis could be utilized as a generic tool for text mining tasks that require a large number of labelled samples, particularly in situations where language inequality exists.

GUIXIAN XU et. al. [11] The contribution of sentiment information to the classic TF-IDF technique is integrated into this paper's suggested improved word representation approach, which generates weighted word vectors. To efficiently collect context information and better represent the comment vectors, the weighted word vectors are transmitted into a bidirectional long short-term memory (Bi-LSTM). A feedforward neural network classifier is used to determine the sentiment trend of the comment. The suggested sentiment analysis approach is compared to RNN, CNN, LSTM, and NB sentiment analysis methods under identical conditions [12]. According to the testing results, the proposed sentiment analysis approach has greater sensitivity, specificity, and F1 score. The strategy has been proven to be beneficial in terms of comment accuracy. However, in the training model, the sentiment analysis approach of comments based on Bi-LSTM takes a lengthy time.

3. Methodology

The methodology followed for our suggested work is represented in fig 1. The summary of the approach of the data augmentation, data pre-processing, word

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 1, 2022, p. 473-482
https://publishoa.com
ISSN: 1309-3452

embeddings, and deep learning models are used in our model.

3.1 Dataset Description

The mobile reviews of Samsung M21 were collected from various social media sites. These are collected as a dataset of 9003 reviews. The unstructured dataset is consisting of text and emojis which has to be classified as positive, negative, and neutral. A miniature of the emojis that are considered in the dataset as a population from various social media posts is shown in Table 1. The emojis in each social media platform for sentiments are represented with the same code and slight changes in their representation.

S.no	Code	Browser	Apple	Google	Facebook	Windows	Twitter	Samsung	Short Name
1	U+263A	☺	☺	☺	☺	☺	☺	☺	smiling face
2	U+1F917	🤗	🤗	🤗	🤗	🤗	🤗	🤗	smiling face with open hands
3	U+1F610	😐	😐	😐	😐	😐	😐	😐	neutral face
4	U+1F44C	👌	👌	👌	👌	👌	👌	👌	OK hand
5	U+1F4F1	📱	📱	📱	📱	📱	📱	📱	mobile phone
6	U+1F50B	🔋	🔋	🔋	🔋	🔋	🔋	🔋	Battery

Table 1: A population of Emojis from various social media platforms

The corpus class considered is consisting of Positive reviews of 74% i.e., 6519, and negative 26% i.e. 2326 from a total of 9003 reviews.

3.2 Preprocessing

The data collected from Samsung M21 mobile reviews is consisting of Text and Emojis. These are compiled in a 'CSV' file. This extracted file is arranged in a format that is free from noise data or different units or missing values etc. The collected corpus is preprocessed so that outcome of the research is not affected. It has six steps involved.

1)DROP NUMBERS: The Mobile reviews corpus may contain a lot of numbers that when some operations on are performed give incorrect outputs, and such unrealistic entries are removed.

2)LOWER CASE CONVERSION: The whole corpus which has uppercase characters are converted into the lower case for uniformity.

3) WORD TOKENISATION: This process splits all the mobile reviews into words. These words are useful in classifying and counting them for identifying a particular sentiment.

4) PUNCTUATION: The tokens identified will have some special characters which need to be removed from our data corpus. The punctuation will be removed and replaced by an empty string.

5) LEMMATIZATION: The word tokens are lemmatized where the context is used to convert the token into meaning full form or root words using the dictionary. The lexical database called WordNet Lemmatize is used in our work.

6) STOPWORDS: The most common words in the language which are frequently used are removed from our corpus. These words add no meaning to our sentiment model. The word 'not' has been retained in our corpus as our proposed work is on sentiment analysis and its meaning helps in identifying the context.

The corpus used in this research has passed through the above all pre-processing steps and the data is represented as shown in Table-2.

3.3 Data Labelling

The data annotation is done on the raw data of mobile reviews. The data is labelled using the Sentiment Intensity Analyser which has a Vader Sentiment, polarity scores, Compound Scores, and Label Encoding. The Valence Score is calculated based on the means of observation and experience. It is measured on a score of -4 to 4 where '-4' is for more negative valence and '4' is for more positive valence. This method relies on a dictionary that maps word and lexical features to sentiment. The polarity scores are calculated in the range of 0 to 1. The compound score is calculated which is in the range of -4 to 4. The researchers used the below normalization for compound score calculation.

$$x = \frac{x}{\sqrt{x^2+\alpha}}$$

where x is the total of the constituent words' valence sc ores, and α is the normalization constant.

3.4 Feature Extraction:

In this proposed work we have two used feature extraction methods for word embeddings. The reviews of each word are represented using a vector in n-dimensional space. The Word2Vec method with CBOW and Skip-gram. The Word2Vec model represents every token in the review as a vector. In Skip-gram, the tokens in the review are considered and the context words are searched in the vicinity. The CBOW is exactly opposite to the previous method where the input is a context word and it tries to predict the output.

3.5 Classification Algorithms:

A convolutional neural network model is presented to categorize the sentiment of reviews as positive or negative. CNN is a deep learning feed-forward artificial neural network. The system employs a variety of multilayer perceptron’s which is meant to use a minimal amount of pre-processing. CNN has fewer connections and is composed of neurons that have Weights and biases that can be learned and trained. CNN is a sophisticated deep learning model that recognizes patterns, Image net categorization, and picture content are done with Exceptional outcomes [2]. CNN became well-known as a result of these findings as a tool for classifying difficulties A CNN is made up of an input layer, many output layers, and a processing layer. There are two hidden layers and one output layer. Layers of convolution, completely connected layers, pooling layers, activation functions other levels are hidden layers, as well.

The LSTM is a type of RNN that can learn long-term dependencies [3]. LSTMs are specifically designed to prevent the long-term dependency issue that plagued ordinary RNNs. A bi-directional LSTM is employed in our LSTM architecture, which is comparable to the one used in. The context of the previous words is kept by one LSTM, while the context of the future words is kept by the other.

As a result, the tweets are first sent to each of the LSTMs, each of which has a concealed size of h. Each LSTM's final output is concatenated to produce a vector of length 2h. This vector is then sent to a fully linked layer that uses the ReLU activation function to activate it.

3.6 Performance Evaluation:

The sentiment analysis is improving and solving various real-world problems. One of the important features of opinion mining is performance evaluation. It is one of the challenging tasks to evaluate the operation of a machine learning model. Appropriate evaluation parameters are very much needed in analyzing the model perfectly [4]. The proposed model's advantages and disadvantages aid researchers in better understanding the work. This proposed work compared the performance parameters of Accuracy, Precision, Recall, and F1-Score.

4. Proposed Framework

The proposed framework is to develop a system that provides a simple and reliable process to identify the sentiment analysis using emojis. The phases of the process are depicted in Figure 1, and our model is presented in more detail in Figure 2. The product reviews are collected from social media websites. These are all stored in a data corpus. For this purpose, 9000 reviews were collected. Pre-processing is done which involves major six steps. It includes the dropping of numbers which eliminates all the numbers in the reviews. Then the data is completely converted into lowercase for the data uniformity. Next, the data tokenization is performed for the identification of individual words. Then, the lemmatization of the data is done for the meaningful semantic word retention in the data corpus. One more important step in pre-processing is the removal of stop words. As our research is on the sentiment analysis so we retained the ‘not’ word which will help in understanding the key aspects of sentiments.

The data labelling is done based on the Valence score calculation, compound score calculation, polarity labelling, and Label encoding. For the data labelling process, the Emoji-based Sentiment lexicon is developed which helps in assigning a polarity score based on the emojis considered. The feature extraction methods considered for word embedding are Word2EC CBOW and Skip-gram.

Next, the data corpus is split into two parts. The training data corpus is of 70% and the remaining 30% is for testing purposes. Table -1 shows the details of machine learning models and their Hyperparameters. The classification algorithms used are LSTM, CNN, Bi-LSTM, and CNN-LSTM. The model is trained with the data that contains emojis also in our work. The training data is given to all four classifiers. Then the test dataset is used as input and the predictions are made. The evaluation of the models is done in the last phase. This is used to check the effectiveness of the proposed model. The indicators in this research are Accuracy, Sensitivity, Specificity, and F1-Score. The hypothesized model's

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 1, 2022, p. 473-482
https://publishoa.com
ISSN: 1309-3452

performance is compared to the performance of each classifier employed in this model separately just for comparison

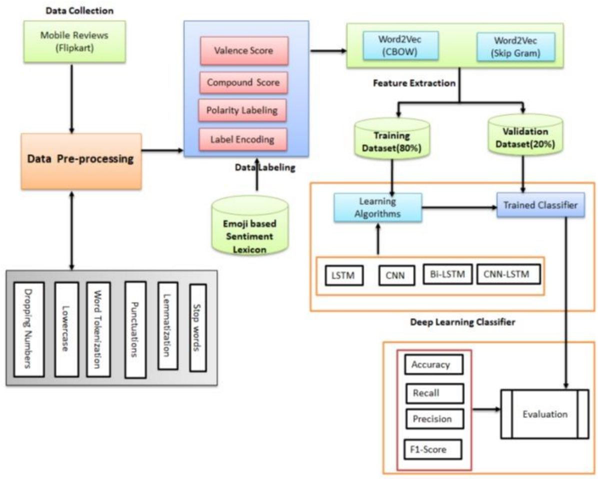

Fig-1: Architecture of the proposed framework

5. Experimentation

In this work the hardware used is Intel(R) i-7, 16GB internal memory,x64 processor, Vostro 5402. The operating system is Windows11 and the program development environment is the Python3.5 programming language. The Pandas NumPy, NLTK, genism, and Keras libraries of python are used to build the proposed sentiment analysis method and perform comparative experiments.

The mobile reviews of 9003 are considered for the corpus. This data is pre-processed with the six steps as explained clearly in the section---. Table 2 shows a sample mobile review of the Samsung M21 phone. Table 3 represents the pre-processed data after each step.

Review: "Excellent 1..... phone with Great battery backup and camera 📷 quality 👌 ❤

Table-2: A sample mobile review by a customer

Stage	pre-processing Technique	Sample output
1	Drop numbers	Excellent phone with Great battery backup and camera 📷 quality 👌 ❤
2	Remove punctuation	Excellent phone with Great battery backup and camera 📷 quality 👌 ❤
3	Convert to lower case	excellent phone with great battery backup and camera 📷 quality 👌 ❤
4	Tokenization	[excellent, phone, with, great, battery, backup, and, camera, 📷, quality, 👌, ❤
5	Lemmatization	[excellent, phone, with, great, battery, backup, and, camera, 📷, quality, 👌, ❤
6	Stop word removal	[excellent, phone, with, great, battery, backup, and, camera, 📷, quality, 👌 ,❤]
7	Final Text	[excellent, phone, great, battery, backup, and, camera, 📷, quality, 👌, ❤]

Table 3: The Sample output after each pre-processing step

From the above fig 3, the stage 1 the number '1' is dropped, in stage 2 the punctuation comma and dots are removed, in stage 3 the review is completely converted to lower case, In stage 4 the tokenization is done for every word where the emojis 📷 👌 ♡ are retained, unlike other works which ignore them. In Stage 5, the Lemmatization is done for retaining the semantics of all the words considered, and in Stage 6 the stop words like 'with' are removed. Finally, the stage 7 data is stored in the data corpus for further processing. The above procedure is repeated for all the 9000 reviews. This data corpus is now labeled using the four techniques mentioned in Table 5.

S.No	Input Emoji	Polarity score
1	😀	2.3
2	☹	-2.1
3	Super	2.9
4	📱	0.2
5	👌	1.5
6	📷	1.0
7	Excellent	3.1
…..		
7514	Worst	-3.1

Table 4: Emoji-based sentiment lexicon

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 1, 2022, p. 473-482
https://publishoa.com
ISSN: 1309-3452

S. No	Data labelling Techniques	Sample output
1	Valence score	neg 0.0 , neu 0.463 , pos 0.537
2	Compound score	compound 0.9317
3	Sentiment labeling	Positive
4	Label Encoding	1

Table 5: The process of data Labeling

In this work, the Emoji based Sentiment lexicon is created. It is a list of lexical features that are created by adding emojis to the existing lexicon. The valency scores are assigned to these emojis based on their equivalent text polarity values as shown in Table 4. These scores are ranging from -4(extremely negative) to 4 (most positive). This will be useful in labeling a review as positive, neutral or negative which is a combination of emojis and text. In the above example the sentiment score, neg — negative is 0.0, pos — positive is 0.537, and neutral is 0.463. The 'compound' or threshold score is 0.9317. The third key is the sentiment labeling of the review as positive or negative and the fourth key value is '1' in the Label encoding which indicates positive.

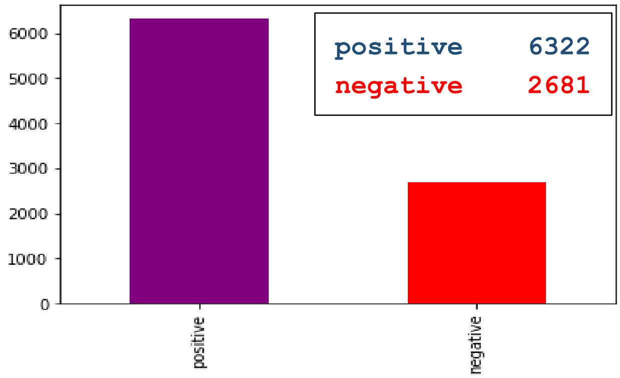

Table 6: Status of Product Reviews

Table 6 is representing the summary of the data corpus after data labeling. All the words in the review are converted into feature vectors of size 3434 *100 using Word2Vec(CBOW and Skip-gram). The parameters considered for these are a) vector size equal to 100 as shown in Table 7 for emoji '☺', b) window =2, c) min_count value is 1 which ignores all the values less than this, workers=2 which indicates the number of threads to train the model. In the trained model, the number of Epochs i.e. no of iterations used is 30. The similarity index for the features are calculated using cosine similarity. For the understanding purpose, a feature 'phone' is shown as in Table 8.

w2v_model['🙂']): emoji2vec

```
[-0.10559238 -0.01508978  0.3918457  -0.20601632  0.07849847 -0.21261047
 -0.04483866 -0.28549522  0.22632946 -0.13188744  0.19426458  0.24906415
 -0.15595575 -0.11857392  0.21335414  0.12431891  0.09985711  0.10875168
 -0.10537813  0.26412192  0.08023263  0.16051461 -0.00407149 -0.09833059
 -0.1599173  -0.42756212 -0.06421069 -0.18818253 -0.08211535 -0.01422769
 -0.1705794  -0.05605508 -0.02097391  0.19061926  0.5729737  -0.17399545
 -0.16118369  0.14628941 -0.31942406 -0.02424535  0.04452643 -0.324061
 -0.0186437   0.17305967 -0.20467825  0.17255409 -0.46979892 -0.24556047
  0.05748998  0.24570595 -0.00206046  0.2306269   0.14530261  0.18696293
 -0.06651347  0.15357956  0.22741319 -0.3035989   0.06318473  0.3330087
  0.05613586 -0.11230835 -0.21567935 -0.3741529  -0.31272906 -0.2194124
  0.07726093  0.330252   -0.4432649  -0.09516697 -0.05664514  0.07897006
  0.2329704   0.16760674  0.01990952 -0.18518354 -0.05053899 -0.12839451
 -0.02871113  0.279383   -0.17689353 -0.18024912 -0.09698466  0.09600269
 -0.48509398 -0.3772633   0.23513803 -0.3617358   0.21257982 -0.31599486
 -0.03885762 -0.01713937  0.47828284  0.16666646  0.10523699 -0.13831821
 -0.06731892  0.06180862  0.05649524  0.05870271]
```

Table 7: Word2Vec: Vector size of 100

Word	Similar words	Similarity score
phone	mobile	0.9454361796379089
	device	0.8916792869567871
	slim	0.8882595300674438
	overall	0.8779714107513428
	product	0.8674222230911255
	specification	0.8621055483818054
	smartphone	0.8619677424430847
	limited	0.8583779335021973
	sheet	0.8578208088874817
	reasonable	0.8568084836006165

Table 8: word2vec model for Similarity index of the feature 'phone'

Every token is converted into a vector of dimension 100 called as 'Embedding_size'. As seen in fig-2, all of these 100-dimensional vectors are stored in an array with a vocabulary size of 3434*100. To standardize the size of the input vectors, zero padding is provided to the shorter reviews because the duration of the various reviews varied. Because the maximum number of terms in a review in the collected data is 3434, a review matrix of size 3434 x 100 is created.

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 1, 2022, p. 473-482
https://publishoa.com
ISSN: 1309-3452

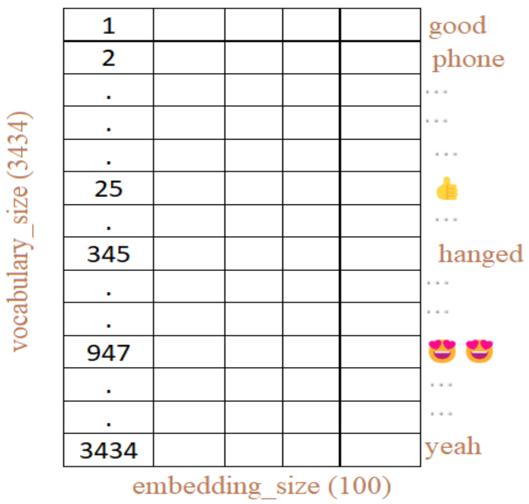

Fig 2: Dimensional Vector of size 3434*100

The resulted review matrix is then fed as input to the classifiers. The model is compiled with a word embedding matrix of size 100D. The optimizer is Adam and the parameter loss function is binary_crossentropy. Fig 3 is representing the hyperparameters on each of the models.

After compilation the training and validation are done for a batch size of 100 with Epochs 5 for all models. The training accuracy (TA) and Training loss (TL) is helpful in determining whether it is overfitting or not. The Validation Accuracy and Validation Loss helps in the same. The X-axis is Epochs and Y-axis is Accuracy for TA and X-Axis is Epochs and Y-axis is Loss as shown in fig 4.

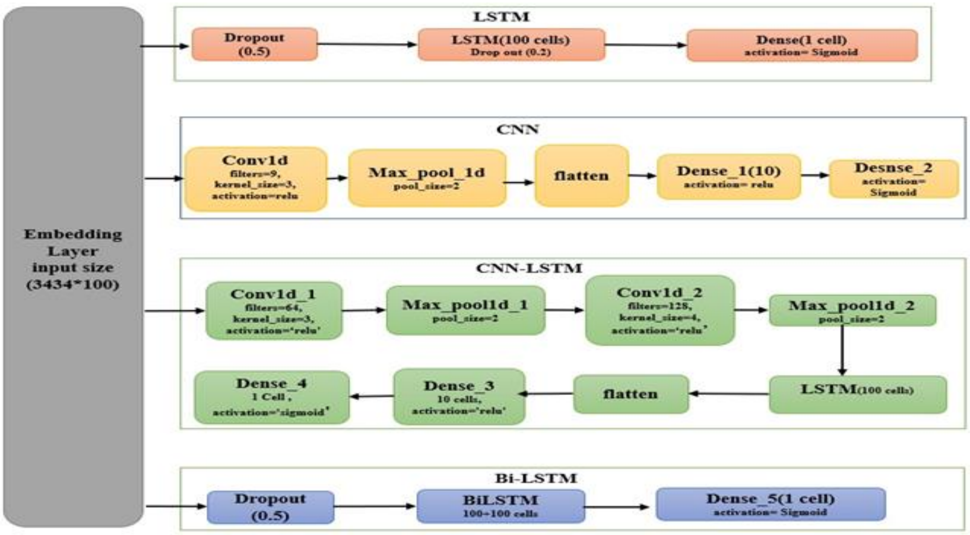

Fig 3: Hyperparameters of the LSTM, CNN, Bi-LSTM,CNN_LSTM models

6. Results and Analysis

The experiment is performed with 9005 mobile reviews which were downloaded from a social media website. The mobile reviews collected had both text and emojis. This data corpus was pre-processed and stored. This data is now labeled using the emoji-based sentiment lexicon as the data was considered along with emojis. For this, the Valence score is calculated for the three labels as negative, positive, and neutral. The Compound scores are calculated and based on these values they are labeled as Sentiment positive/negative. Finally, the label encoding is done. The word embedding is done using the Word2Vec model. The methods CBOW and Skip-gram are compared with all the deep learning classifiers.

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 1, 2022, p. 473-482
https://publishoa.com
ISSN: 1309-3452

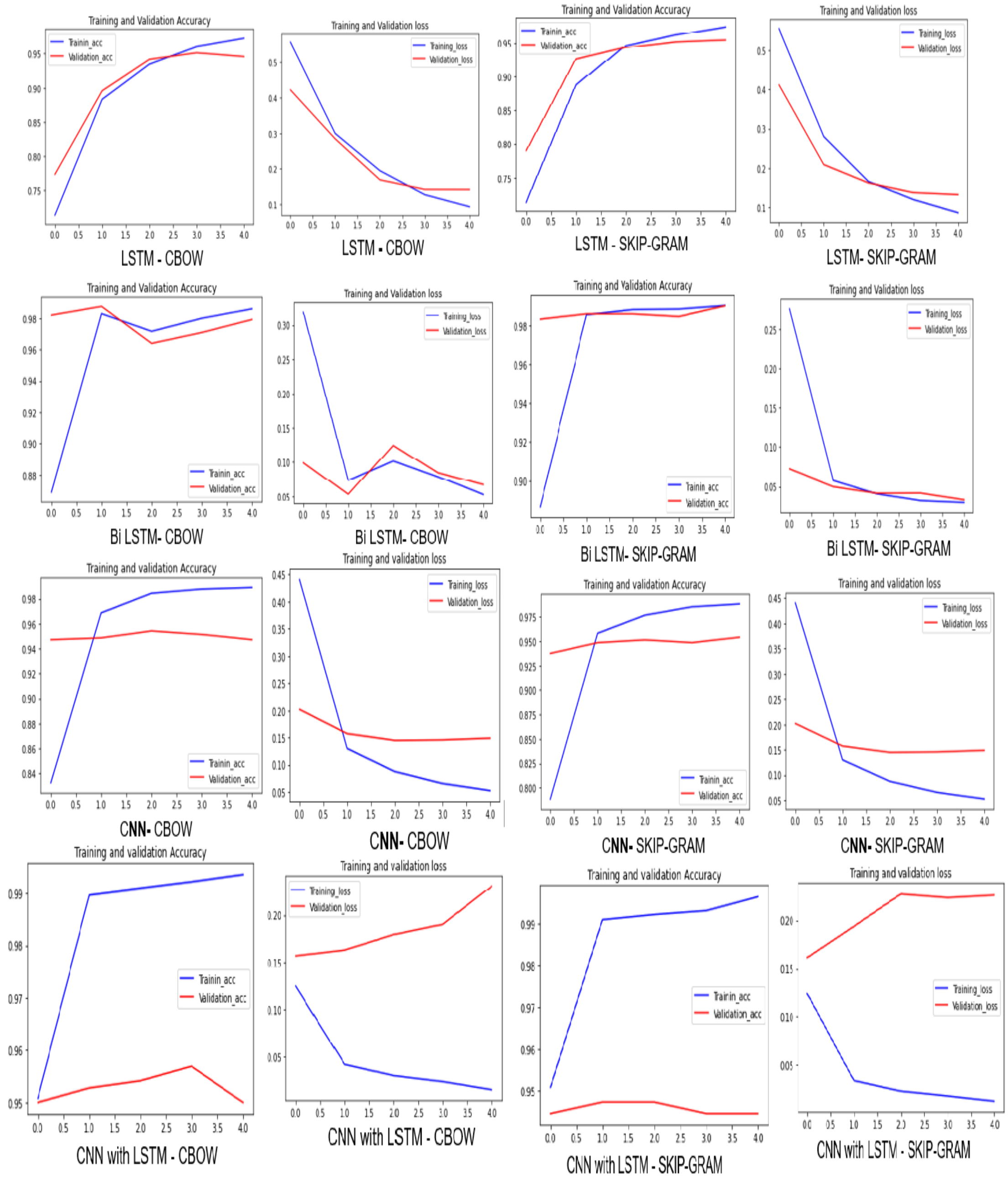

Fig 4: The Epochs with respect to Training Accuracy, Validation Accuracy, Training Loss and Validation Loss

The results of our experiments are evaluated using accuracy, precision, recall and F1-score. The accuracy is calculated as follows:

$$\text{Accuracy} = \frac{\text{Identified Correct mobile reviwes}}{\text{Total Mobile reviews}}$$

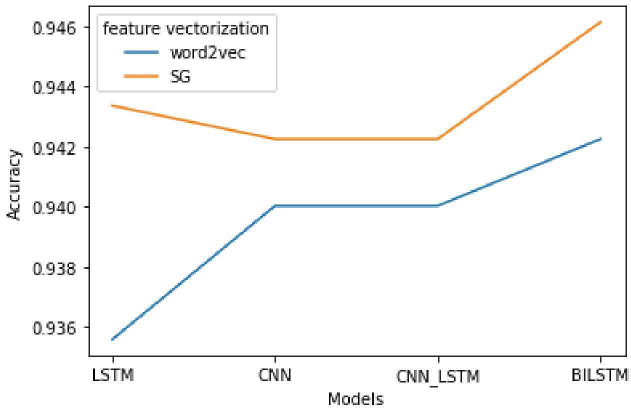

Fig 5: Accuracies of Deep Learning models

Table 9 represents the comparison of the CBOW and SG concerning the evaluation parameters. Fig 5 represents the accuracies of the various classification algorithms. The Deep learning classifiers used are LSTM, CNN, BiLSTM, and CNN-LSTM. The proposed model of using Emoji2Vec for the SA has given satisfactory results. The accuracy of all these models is listed below. Moreover, it is observed that the SG method results are slightly better than the CBOW methods.

Features / Classifier		word2Vec (CBOW)				word2Vec (SG)			
	Class	Precision	Recall	F1Score	Accuracy	Precision	Recall	F1Score	Accuracy
LSTM	Negative	0.88	0.88	0.88	0.93	0.88	0.91	0.89	0.94
	Positive	0.95	0.96	0.96		0.97	0.96	0.96	
CNN	Negative	0.89	0.92	0.91	0.95	0.91	0.92	0.91	0.95
	Positive	0.97	0.96	0.96		0.97	0.97	0.97	
BiLSTM	Negative	0.88	0.91	0.89	0.94	0.9	0.9	0.9	0.95
	Positive	0.96	0.96	0.96		0.96	0.96	0.96	
CNN-LSTM	Positive	0.87	0.91	0.89	0.94	0.89	0.91	0.9	0.95
	Negative	0.97	0.95	0.96		0.97	0.96	0.96	

Table 9: The evaluation parameters of the model for the mobile review dataset

The accuracy of the CNN-LSTM classifier with both word embeddings resulted in 94.94% consistently. The second best is the Bi-LSTM using the CBOW model with an accuracy of 94.83%. All the other models are resulting nearly the same concerning accuracy.

The dataset used for these experiments has around 9003 reviews only. The proposed model was successful in SA of all the reviews without filtering emojis. Moreover, emojis were also used as a part of the source in finding the analysis. The word embedding feature extraction methods with different classifiers also resulted from mostly nearby results as we have used a small dataset. In the future, we have a plan of evaluating the proposed Emoji2Vec model with deep learning algorithms like BERT also.

7. Conclusion

In this proposed work, the SA is performed on mobile product reviews using various deep learning algorithms. The novelty of this work was to develop an Emoji-based sentiment lexicon based on the text polarity scores. The performance evaluators considered are sensitivity, specificity, F1-score, and accuracy. The mobile reviews considered had both text and emojis. This data was preprocessed and data labeling was done. During data labeling, we used the Emoji2Vec model for calculating the polarity scores. The feature extraction is done using the word embedding models CBOW and SG. Then the LSTM, CNN, Bi-LSTM, and CNN-LSTM models were developed along with their hyperparameters. The experimented results prove that all the models perform well. The CNN-LSTM proved to have high accuracy. The limitation was about the computational requirements for processing large data and few neural network models. Moreover, the SA cannot be limited to only identifying positive and negative labels of text and Emoji data. The aspects of the products also have to verify. So, in the future, this work is extended to complex Deep learning architectures and aspect-based.

References

[1] https://www.unicode.org/emoji/charts-14.0/emoji-versions.html

[2] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems, pages 1097–1105, 2012.

[3] Hochreiter, S., Schmidhuber, J., (1997). “Long short-term memory”. Neural computation 9, 1735–1780

[4] A. Tharwat, ‘‘Classification assessment methods,’’ Appl. Comput. Information., Aug. 2020

[5] Jagadeesh Panthati, Sentiment Analysis of Product Reviews using Deep Learning

[6] Sentiment analysis on Bangla text using extended lexicon dictionary and deep learning algorithms Nitish Ranjan Bhowmik a, Mohammad Arifuzzaman b,∗ , M. Rubaiyat Hossain Mondal a,∗

[7]TensorFlow API Developer. Keras preprocessing library, module:tf.keras.preprocessing.text.tokenizer. 2021, https://www.tensorflow.org/

api_docs/python/tf/keras/preprocessing/text/Tokenizer. (Accessed 31 July 2021).

[8] Hossain, Naimul, et al. "Sentiment analysis of restaurant reviews using combined CNN-LSTM." 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT). IEEE, 2020.

[9] Emoji-Powered Representation Learning for Cross-Lingual Sentiment Classification Zhenpeng Chen1, Sheng Shen1,2, Ziniu Hu1,3, Xuan Lu1, Qiaozhu Mei4, Xuanzhe Liu1∗

[10] Xinjie Zhou, Xiaojun Wan, and Jianguo Xiao. 2016. Cross-lingual sentiment classification with bilingual document representation learning. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016. 1403–1412

[11] Sentiment Analysis of Comment Texts Based on BiLSTM GUIXIAN XU 1, YUEQING MENG1, XIAOYU QIU2, ZHENG YU 1, AND XU WU1.

[12] W. A. Shu-heng, T. U. Ibrahim, and K. Abiderexiti, ‘‘Sentiment classification of Uyghur text based on BLSTM,’’ Comput. Eng. Des., vol. 38, no. 10, pp. 2879–2886, 2017.

[13] S. Krishnamoorthy, ‘‘Sentiment analysis of financial news articles using performance indicators,’’ Knowl. Inf. Syst., vol. 56, no. 2, pp. 373–394, Aug. 2018.

[14] Fauzi, M.A., “Random Forest Approach for Sentiment Analysis in Indonesian Language,” Indonesian Journal of Electrical Engineering and Computer Science, 12(1), 2018.

[15] Lilleberg, Joseph, Yun Zhu, and Yanqing Zhang. "Support Vector Machines and Word2vec for Text Classification with Semantic Features." In Cognitive Informatics & Cognitive Computing (ICCI* CC), 2015 IEEE 14th International Conference on, pp. 136-140. IEEE, 2015.

[16] Lizhen, Liu, Song Wei, Wang Hanshi, Li Chuchu, and Lu Jingli. "A Novel Feature-Based Method for Sentiment Analysis of Chinese Product Reviews." China communications 11, no. 3: 154-164, 2014.

[17] Yang, Xiao, Craig Macdonald, and Iadh Ounis. "Using Word Embeddings In Twitter Election Classification." arXiv preprint arXiv:1606.07006, 2016.

[18] Harinder Kaur, Sentiment Analysis of User Review Text through CNN and LSTM Methods- Palarch’s Journal Of Archaeology Of Egypt/Egyptology 17(12), ISSN 1567- 214x

[19] Shah, A. M., Yan, X., Shah, S. A. A., & Mamirkulova, G. (2019). Mining patient opinion to evaluate the service quality in healthcare: a deep learning approach. Journal of Ambient Intelligence and Humanized Computing, 11(7), 2925–2942.

[20] Mulkalwar, Anurag, and Kavita Kelkar. "Sentence Level Sentiment Classification Using HMM with the help of Part of Speech Tagging." International Journal of Computer Science Engineering and Information Technology Research (IJCSEITR) (2014)