International Journal of Advanced Engineering Research and Science (IJAERS)
Peer-Reviewed Journal
ISSN: 2349-6495(P) | 2456-1908(O)
Vol-9, Issue-12; Dec, 2022
Journal Home Page Available: https://ijaers.com/
Article DOI: https://dx.doi.org/10.22161/ijaers.912.12

Survey of Deep Learning Approaches for Twitter Text Classification

Mr. Lukesh Kadu, Dr.Manoj Deshpande, Dr.Vijaykumar Pawar

[1]Research Scholar, A.C.Patil College of Engineering, Kharghar, Navi Mumbai, India
[2]Computer Departement, A.C.Patil College of Enginering, Khargahr, Navi Mumbai, India
[3]Principal, A.C.Patil College of Engineering, Khargahr,Navi Mumbai, India

Received: 15 Nov 2022,

Receive in revised form: 03 Dec 2022,

Accepted: 10 Dec 2022,

Available online: 17 Dec 2022

©2022 The Author(s). Published by AI Publication. This is an open access article under the CC BY license (https://creativecommons.org/licenses/by/4.0/).

***Keywords*— Convolution Neural Network (CNN), Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM), Deep Learning, Bidirectional Long Short-Term Memory (BiLSTM), Bidirectional Encoder Representations from Transformers (BERT), Robustly Optimized BERT Pre-training Approach (RoBERTa).**

***Abstract*— Sentiment analysis (also known as opinion mining or emotion AI) is the use of natural language processing, text analysis, computational linguistics, and biometrics to systematically identify, extract, quantify, and study affective states and subjective information. Sentiment analysis is widely applied to voice of the customer materials such as reviews and survey responses, online and social media, and healthcare materials for applications that range from marketing to customer service to clinical medicine. With the rise of deep language models, such as RoBERTa, also more difficult data domains can be analyzed, e.g., news texts where authors typically express their opinion/sentiment less explicitly. Sentiment analysis aims to extract opinion automatically from data and classify them as positive and negative. Twitter widely used social media tools, been seen as an important source of information for acquiring people's attitudes, emotions, views, and feedbacks. Within this context, Twitter sentiment analysis techniques were developed to decide whether textual tweets express a positive or negative opinion. In contrast to lower classification performance of traditional algorithms, deep learning models, including Convolution Neural Network (CNN) and Bidirectional Long Short-Term Memory (Bi-LSTM), have achieved a significant result in sentiment analysis. Keras is a Deep Learning (DL) framework that provides an embedding layer to produce the vector representation of words present in the document. The objective of this work is to analyze the performance of deep learning models namely Convolutional Neural Network (CNN), Simple Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM), bidirectional Long Short-Term Memory (Bi-LSTM), BERT and RoBERTa for classifying the twitter reviews. From the experiments conducted, it is found that RoBERTa model performs better than CNN and simple RNN for sentiment classification.*

I. INTRODUCTION

Social media is a platform for people to express their feelings, feedback, and opinions. To understand the sentiment context of the text, sentiment analysis plays the role to determine whether the sentiment of the text is positive, negative, neutral or any other personal feeling. Sentiment analysis is important from the perspective of business or politics where it highly impacts the strategic decision making, Therefore, sentiment analysis is recognized as a significant technique to generate useful information from unstructured data sources such as tweets or reviews. Social media platforms, including Twitter, Facebook, Instagram, blogs, reviews and news websites allow people to share widely their opinions and reviews. Generally, sentiment analysis categories into three levels namely document-level, sentence-level, and feature-level. Document-level sentiment analysis classifies the whole review document as either positive or negative. Semantic orientation approaches and machine learning approaches are the two methods used for sentiment classification. Semantic orientation approaches determine the word's polarity using a corpus or dictionary. They do not perform well in terms of classification accuracy because there is no single knowledge base which provides polarity for every domain. Machine Learning (ML) approaches initially build a model from the labelled data and then use the built model to classify the test data. They require large amount of labelled training data to build an efficient model[1].

Machine Learning involves algorithms which extract knowledge from data for creating predictions, rather than involving humans to manually develop rules and build models for resolving enormous amount of knowledge. There are three types of algorithms used for machine learning such as supervised, unsupervised and reinforcement learning. In supervised learning, the data with class labels also called training data is used by the machine learning algorithms to construct a model. The trained model is then used to identify the class label of new unseen test data. In unsupervised learning, the model automatically finds patterns and relationships in the dataset by creating clusters in it [2]. Reinforcement learning aims to develop a system or an agent that learns from the rewards and punishments received from the environment In document-level sentiment classification, lexical, syntactic, and semantic features in a document are first extracted. Then, weights are assigned to these features using binary, Term Frequency (TF) and Term Frequency- Inverse Document Frequency (TF-IDF) weighting schemes and given as input to the machine learning algorithms. The performance of ML based sentiment classification depends on the feature extraction techniques, feature selection methods and feature weighting schemes used. It is not always possible to get labelled data for all the domains to train the model. Also, machine learning approaches require manual effort to extract the features. To address the above issues, this work introduces deep learning models for sentiment classification.

Twitter tweets contain hidden valued information that can be used to determine an author's attitude for a contextual polarity in the text [2]. Even though statistical machine learning algorithms per- form well for simpler sentiment analysis applications, these algorithms cannot be generalized to more complex text classification problems. Deep learning is a technique which is nowadays used in a wide range of applications, The advantages of deep learning models include automatic feature extraction, easy computation due to the use of accelerated hardware, provides best performance even with huge amount of data. Deep learning models achieve significant results in sentiment analysis speech recognition and computer visions. There are some deep learning algorithms that are widely used in sentiment analysis are Convolution Neural Network (CNN) and Recurrent Neural Network (RNN) Simple RNN and RNN with LSTM tries to analyses for sentiment classification. Stochastic Gradient Descent, RMSprop are used as optimizers and their performance is evaluated. Word2Vec and Glove models were used as word embedding technique to present the tweets in the form of numeric values or vectors. These models are pre-train unsupervised word vectors that are trained with a large collection of words and can capture word semantics. The study applied these different word vector models to verify effectiveness of the model.

Sentiment analysis and emotion analysis are performed. Text Blob is used for annotating the sentiments data while emotions are annotated using the Text2Emotion model. Positive, negative, and neutral sentiments are used while emotions are classified into happy, sad, surprise, angry, and fear. The suitability and performance of three feature engineering approaches are studied including term frequency-inverse document frequency (TF-IDF), bag of words (BoW), and Word2Vec. Experiments are performed using several well-known machine learning models such as support vector machine (SVM), logistic regression (LR), Gaussian Naive Bayes (GNB), extra tree classifier (ETC), decision tree (DT), and k nearest neighbour (KNN).

II. LITERATURE REVIEW

K. S. Kalaivani and S. Uma suggested approaches for deep learning. Keras is a Deep Learning (DL) framework that provides an embedding layer to produce the vector representation of words present in the document. analyzed the performance of three deep learning models namely Convolutional Neural Network (CNN), Simple Recurrent Neural Network (RNN) and Long Short-Term Memory (LS TM) for classifying the book reviews. From the experiments conducted, it is found that LSTM model performs better than CNN and simple RNN for sentiment classification**.**

Sakirin Tan and Rachid Ben Said implemented ConvBiLSTM;a word embedding model which converts tweets into numerical values, CNN layer receives feature embedding as input and produces smaller dimension of features, and the Bi-LSTM model takes the input from the CNN layer and produces classification result [4]. Word2Vec and GloVe were distinctly applied to observe the impact of the word embedding result on the proposed model. ConvBiLSTM was applied with retrieved Tweets and SST-2 datasets. ConvBiLSTM model with Word2Vec on retrieved Tweets dataset outperformed the other models with 91.13% accuracy.

Sungheetha and Sharma [5] introduced a new Capsule model known as Trans Cap to address the issue of labelling the aspect-level data. Aspect and dynamic routing algorithms are used to transfer the knowledge from the document-level task to aspect-level task. The authors proved that the proposed model performs better than the state-of- the art models for aspect-level sentiment analysis.

Kalaivani and Kuppuswami improved the performance of syntactic features for document-level sentiment classification by backing off the head word or modifier word to the corresponding POS cluster [6]. The authors proved that the use of WFO based feature selection technique to select prominent generalized syntactic features outperforms other existing features for classifying product reviews.

Soujanya Poria and Devamanyu Hazarika discussed this perception by pointing out the shortcomings and under-explored, yet key aspects of this field necessary to attain true sentiment understanding. We analysed the significant leaps responsible for its current relevance. Further, we attempt to chart a possible course for this field that covers many overlooked and unanswered questions [7].

Ambreen nazir, Yuan Rao, Ling Sun explore the Issues and challenges that are related to extraction of different aspects and their relevant sentiments, relational mapping between aspects, interactions, dependencies, and contextual-semantic relationships between different data objects for improved sentiment accuracy, and prediction of sentiment evolution dynamicity [8].

Kian Long Tan, Chin Poo Lee, Kian Ming Lim proposed The Robustly optimized BERT approach maps the words into a compact meaningful word embedding space while the Long Short-Term Memory model captures the long-distance contextual semantics effectively. hybrid model outshines the state-of-the-art methods by achieving F1-scores of 93%, 91%, and 90% on IMDb dataset, Twitter US Airline Sentiment dataset, and Sentiment140 dataset, respectively [8].

A densely connected convolutional neural network with multi-scale feature attention was developed by Wang et al., for text classification [9]. Dense connections are used to easily generate large N-gram features from various smaller N-gram features. Feature attention mechanism is used to select effective features with varying N-grams such as unigrams, bigrams and trigrams from multi-scale features.

To overcome the problem of capturing sentiments present in the text from long-time steps, Huang et al., developed a novel model called Sentence Representation-Long Short- Term Memory (SR-LSTM) [10]. The variants of LSTM such as peephole connection LSTM, coupled input output forget LSTM, gated recurrent unit (GRU) and bidirectional LSTM were implemented. Finally, the authors concluded that the newly introduced models SR-LSTM and SSR-LSTM build more accurate model compared to other models for IMDB, Yelp 2014 and Yelp 2015.

Peng et al., introduced a novel deep graph CNN model to capture non-consecutive relations and long range semantic relations for large scale text classification [11]. In few applications like sentiment analysis, capturing long range semantics is more important than sequential information. Initially, the text was converted into graph-of-words and graph convolution operation was performed to capture the text semantics. From the results, it is clear that the proposed model performs better than the existing classification models.

III. PROPOSED WORK

A. Deep Learning

Recently, deep learning algorithms have achieved remarkable results in natural language processing area. They represent data in multiple and successive layers. They can capture the syntactic features from sentences automatically without extra feature extracting techniques, which consume more resource and time. This is the reason why deep learning models have attracted attention from NLP researchers to explore sentiment classification. By making use of a multi-layer perceptron structure in deep learning, CNN can learn high-dimensional, non-linear, and complex classification. As a result, CNN is used in many applications such as computer vision, image processing, and speech recognition.

B. Convolutional Neural Network

Figure 1 shows the architecture of CNN which consists of a convolutional layer, pooling layer. Flatten layer and a dense layer. Generally, CNN is used for image, audio and video applications like image classification, semantic segmentation, object detection etc., In recent times, it has been applied to text classification and has shown good performance, So, in this work it is used for sentiment classification as convolutional filters present in this model is able to automatically learn the prominent features for this task.

1) Embedding Layer: Neither the machine learning algorithms nor deep learning algorithms can directly process the raw text. It should be converted into a numerical form for further analysis. Two most used embeddings are frequency-based embeddings and prediction-based embeddings. Frequency based embeddings use count vector, TFIDF vector or co-occurrence vector to represent the documents. Since these methods are limited in representing.

2) Convolutional Layer: The purpose of this layer is to select the high-level features for sentiment classification. As the name implies, convolution operation is performed in this layer. A filter is move over the input matrix to construct the feature map. The feature map size is managed by three criterions such as depth, stride, and padding. Depth depends on number of filters used for convolution operation.

3) Pooling Layer: This layer is introduced to reduce the dimensions of the feature that was produced as output from the convolutional layer. This layer reduces the computations needed to reduce the dimensionality of the data. There are three types of pooling namely max pooling, average pooling, and sum pooling. In this work, max pooling is used. Max pooling identifies the maximum value from the portion of the data covered by the kernel or filter. From the literature, it is found that max pooling outperforms average and sum pooling in various applications.

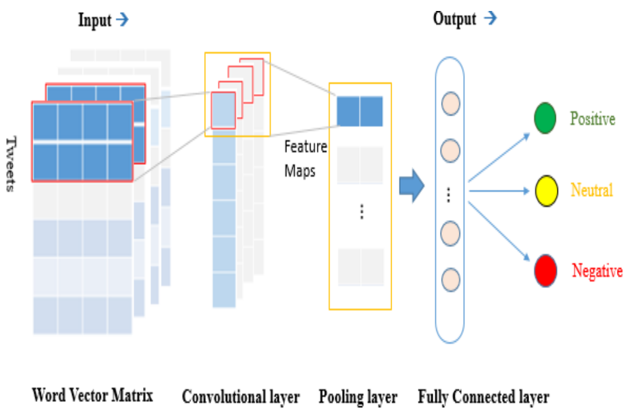

Fig 1. Architecture of CNN

4) Flatten Layer: Flattening layer is used to convert the feature matrix into a vector of feature values. So, the unified pooled feature matrix is converted into a single column vector.

5) Dense Layer: Dense layer is used to identify the class label depending on the activation function used. Activation functions used may be SoftMax or sigmoid based on the type of classification task. SoftMax is used for multiclass classification and sigmoid is used for binary classification. Since, the reviews are either positive or negative, sigmoid activation function is used.

C. Recurrent Neural Network

Recurrent neural network (RNN) is a subdivision of networks which are applicable for studying representation of subsequent data such as Natural language processing. It yields an objective function that depends not only on the current input but also along with earlier state output or hidden state. Here, earlier state output is a function of earlier state input. The current state output is a function of previous input and output.

$$h_t = \tanh\left(b + W_{ht-1} + U_{xt}\right)$$

where b is the bias value, W represents the weights for the previous output and U is the weight for the current input. t is used to denote the position in the sequence.

Figure 2 shows the architecture of simple RNN. The raw data is pre-processed and a vocabulary is constructed which contains unique words present in the document. This is passed to an embedding layer which provides the embedding value for each and every word present in the vocabulary. The embedding values are passed to a simple recurrent neural network. It predicts the output for current text depending on previous output and input. The output of SRNN layer is passed to dropout layer which avoids overfitting by dropping some of the features that are not prominent. Finally, dense layer along with the activation function provides the polarity of the review.

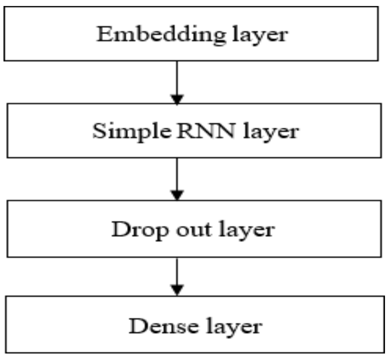

Fig 2. Architecture of SRNN

D. Long Short-Term Memory

The main component of LSTM is the cell state and various gates. The cell state is responsible for transferring the information along the sequence chain. The cell state acts like a memory by carrying the information for the complete processing of the entire sequence. Here, the short-term memory issue of RNN is overcome such that even the relevant information from the initial time steps can have its impact till the later time steps. So, the relevant information gets added and irrelevant information gets removed via gates in the cell state during the training process.

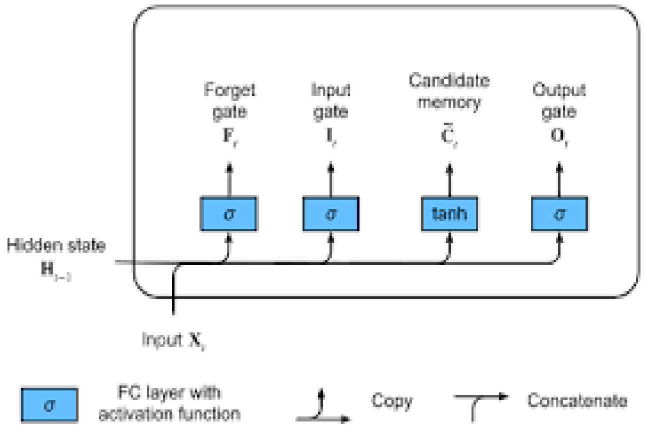

Fig 3. Architecture of LSTM

E. Bidirectional LSTM

Bi-LSTM is one of RNN algorithms to improve LSTM which has shortcomings of text sequence features. It solves the task of sequential modelling better than LSTM [32], [33]. In LSTM, information is flowed from backward to forward, whereas the information in Bi-LSTM flows in both directions backward to forward and from forward to backward by using two hidden states. The structure of Bi-LSTM makes it a pioneer in sentiment classification because it can learn the context more effectively. Figure 4 shows the architecture of Bi-LSTM [34]. By utilising two ways of direction, input data of both preceding and succeeding sequence in Bi-LSTM are retained, unlike the standard RNN model that needs decay to include future data.

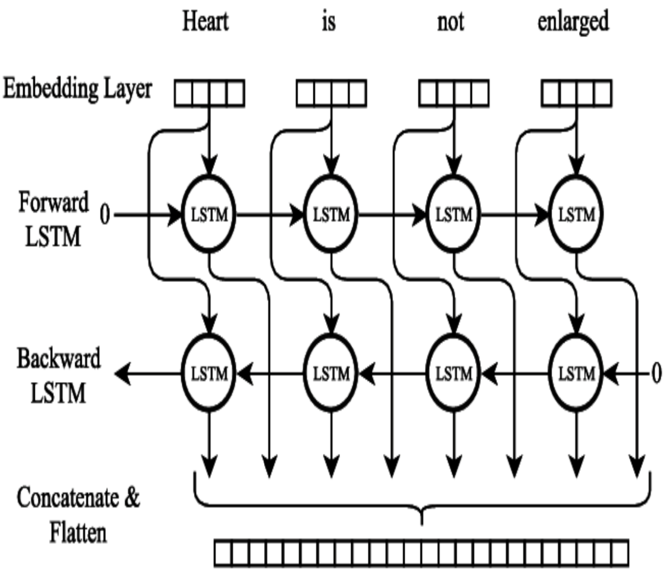

Fig 4. Architecture of BiLSTM

F. BERT (Bidirectional Encoder Representations from Transformers)

It is a Natural Language Processing Model which achieve state-of-the-art accuracy on many NLP and NLU tasks such as: BERT is basically an Encoder stack of transformer architecture. A transformer architecture is an encoder-decoder network that uses self-attention on the encoder side and attention on the decoder side. BERT makes use of Transformer, an attention mechanism that learns contextual relations between words (or sub-words) in a text. In its vanilla form, Transformer includes two separate mechanisms — an encoder that reads the text input and a decoder that produces a prediction for the task. Since BERT's goal is to generate a language model, only the encoder mechanism is necessary.

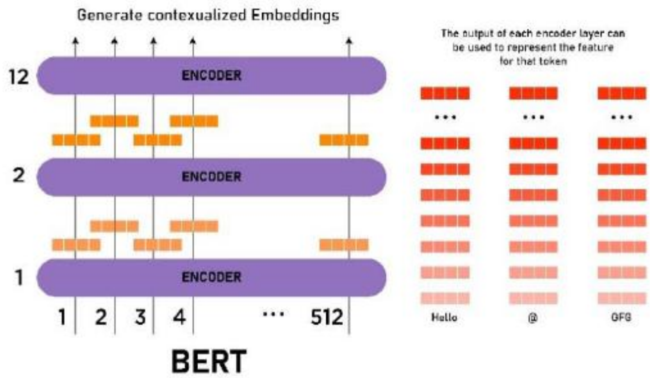

Fig 5. Architecture of BERT

F. RoBERTa (Robustly Optimized Bidirectional Encoder Representations from Transformers)

RoBERTa The RoBERTa model is an extension of Bidirectional Encoder Representation from Transformers (BERT). The BERT and RoBERTa fall under the Transformers [2] family that was developed for sequence-to-sequence modeling to address the long-range dependencies problem.

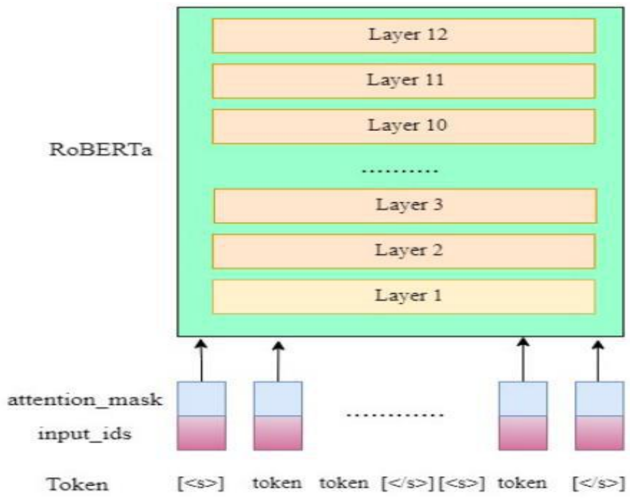

Fig 6. Architecture of RoBERTa

Transformer models comprise three components, namely tokenizer, transformers, and heads. The tokenizer converts the raw text into the sparse index encodings. Then, the transformers reform the sparse content into contextual embedding for deeper training. The heads are implemented to wrap the transformers model so that the contextual embedding can be used for the downstream tasks. The components of the Transformers are depicted in Figure 6.

IV. RESULT AND DISCUSSION

A. Dataset Used

1) Huge crash in stock market 2022

Gathered Tweets related to Stock Market Crash in 2022 from

Twitter which performs various task NLP task on this data source. The sentiment of the tweet's column consists of three categories: Positive 12542 tweets Neutral 11498 tweets Negative 9906 tweets.

2) Stock Market TWEETS Data-NL2021

Twitter is one of the most popular social networks for sentiment analysis. This data set of tweets are related to the stock market.
We collected 943,672 tweets between April 9 and July 16, 2020, using the S&P 500 tag (#SPX500), the references to the top 25

3) Stock Market Tweet | Sentiment Analysis lexicon

Tweets were collected between April 9 and July 16, 2020 using not only the SPX500 tag but also the top 25 companies in the index and "#stocks". 1300 tweets were manually classified and reviewed. All the source code used to download tweets, check the top words, and evaluate the sentiment are present.

Three deep learning architectures CNN, simple RNN and LSTM are compared for document-level sentiment classification. Below figures shows the training and testing accuracy obtained for all the three networks. From the figures, it is clear that LSTM shows superior performance when compared to other two networks in terms of accuracy. The reason behind this is that both CNN and simple RNN models are not able to remember the sequence of words like LSTM network.

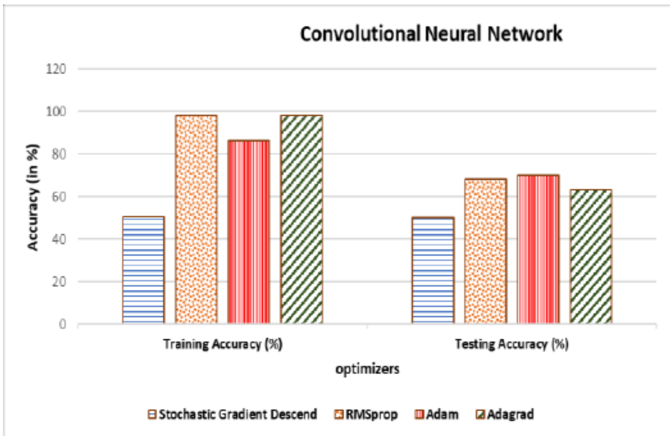

Fig 7: Performance comparison of CNN

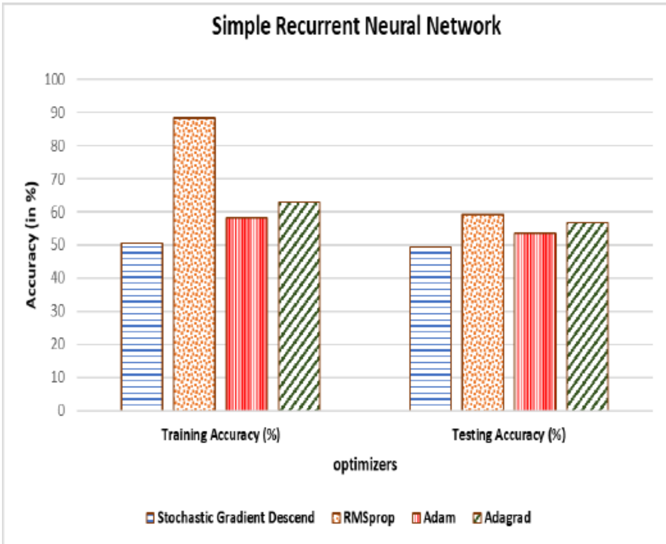

Fig 8: Performance comparison of SRNN

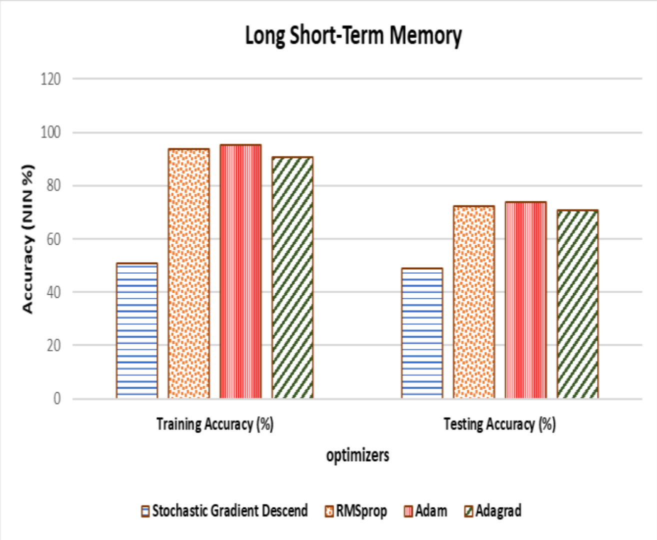

Fig 9: Performance comparison of LSTM

V. CONCLUSIONS

The performance of three deep learning models is analysed for document-level sentiment classification. For sentiment classification, the local and non-local relationship between the words in the sentence should be considered for improved classification performance. The proposed approach helps the model to classify text sentiment effectively by capturing both local and global dependencies in the contextual of sentences. The model is trained and evaluated on tweets dataset like Stock Market Tweet, Sentiment Analysis lexicon, Stock Market TWEETS Data-NL2021 and Huge crash in stock market 2022 dataset. Finally, the model could classify text sentiment effectively on both datasets. The experiment result verified the feasibility and effectiveness of model. In the future, the performance of other deep learning models may be analyzed for sentiment classification.

REFERENCES

[1] K. S. Kalaivani, S. Uma & Dr.C.S. Kanimozhiselvi "Comparison of Deep Learning Approaches for Sentiment Classification" IEEE Xplore, ICICT 2021.

[2] Borja Arroyo Galendei, Silvia Uribes. "Conspiracy or not? A Deep Learning Approach to Spot It on Twitter" IEEE Access, February 15, 2022,

[3] Naila Aslam, Furqan Rustam, Ernesto Lee & Patrick Bernard Washington "Sentiment Analysis and Emotion Detection on Cryptocurrency Related Tweets Using Ensemble LSTM-GRU Model" IEEE Access March 19, 2022.

[4] Ambreen Nazir &Yuan Rao, Lianwei Wu, "Issues and Challenges of Aspect-based Sentiment Analysis: A Comprehensive Survey", IEEE Transaction on Affective Computing, Vol. 13, No. 2, April-June 2022.

[5] Sakiran Tam, Rachid Ben Said & O. Ozgur Tanriover, "A ConvBiLSTM Deep Learning Model-Based Approach for Twitter Sentiment Classification: from IEEE Access March 19, 2021.

[6] A. Sungheetha & R. Sharma, "Trans capsule model for sentiment classification", Journal of Artificial Intelligence, vol. 2, no. 03, pp. 163-169, 2020.

[7] K. S. Kalaivani & S. Kuppuswami, "Exploring the use of syntactic dependency features for document -level sentiment classification", Bulletin of the Polish Academy of Sciences. Technical Sciences, vol. 67, no. 2, 2019.

[8] Soujanya Poria, Devamanyu Hazarika, Navonil Majumder, Rada Mihalcea: Current Challenges and New Directions in Sentiment Analysis Research", IEEE Transactions on Affective Computing 2020 IEEE

[9] Kian Long Tan, Chin Poo Lee, Kian Ming Lim, "RoBERTa-LSTM: A Hybrid Model for Sentiment Analysis with Transformer and Recurrent Neural Network", IEEE Access March 2,2022.

[10] S. Wang, M. Huang & Z. Deng, Z, "Densely Connected CNN with Multi-scale Feature Attention for Text Classification", In IJCAI, pp. 4468-4474, 2018.

[11] A. Yenter & A. Verma, A, "Deep CNN-LST M with combined kernels from multiple branches for IMDb review sentiment analysis", In 2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference, pp. 540-546, 2017.

[12] A. Yenter & A. Verma, A, "Deep CNN-LST M with combined kernels from multiple branches for IMDb review sentiment analysis "In 2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference, pp. 540-546, 2017

[13] S. Wang, M. Huang & Z. Deng, Z, "Densely Connected CNN with Multi-scale Feature Attention for Text Classification", In IJCAI, pp. 4468-4474, 2018.

[14] G. Rao, W. Huang, Z. Feng & Q. Cong, "LSTM with sentence representations for document-level sentiment classification", Neurocomputing, vol. 308, pp. 49-57, 2018.

[15] H. Peng, J. Li, Y. He, Y. Liu, M. Bao, L. Wang & Yang, Q, "Large- scale hierarchical text classification with recursively regularized deep graphcnn", In Proceedings of the 2018 World Wide Web Conference pp. 1063-1072, 2018.

[16] Adyan Marendra Ramadhani & Hong Soon Goo, "Twitter Sentiment Analysis using Deep Learning Methods", 2017 7th International Annual Engineering Seminar (InAES), Yogyakarta, Indonesia