Multimedia Tools and Applications (2019) 78:26597–26613
https://doi.org/10.1007/s11042-019-07788-7

A Hybrid CNN-LSTM Model for Improving Accuracy of Movie Reviews Sentiment Analysis

Anwar Ur Rehman[1] · Ahmad Kamran Malik[1] · Basit Raza[1] · Waqar Ali[1]

Received: 29 August 2018 / Revised: 3 April 2019 / Accepted: 15 May 2019 /
Published online: 11 June 2019

Abstract

Nowadays, social media has become a tremendous source of acquiring user's opinions. With the advancement of technology and sophistication of the internet, a huge amount of data is generated from various sources like social blogs, websites, etc. In recent times, the blogs and websites are the real-time means of gathering product reviews. However, excessive number of blogs on the cloud has enabled the generation of huge volume of information in different forms like attitudes, opinions, and reviews. Therefore, a dire need emerges to find a method to extract meaningful information from big data, classify it into different categories and predict end user's behaviors or sentiments. Long Short-Term Memory (LSTM) model and Convolutional Neural Network (CNN) model have been applied to different Natural Language Processing (NLP) tasks with remarkable and effective results. The CNN model efficiently extracts higher level features using convolutional layers and max-pooling layers. The LSTM model is capable to capture long-term dependencies between word sequences. In this study, we propose a hybrid model using LSTM and very deep CNN model named as Hybrid CNN-LSTM Model to overcome the sentiment analysis problem. First, we use Word to Vector (Word2Vc) approach to train initial word embeddings. The Word2Vc translates the text strings into a vector of numeric values, computes distance between words, and makes groups of similar words based on their meanings. Afterword embedding is performed in which the proposed model combines set of features that are extracted by convolution and global max-pooling layers with long term dependencies. The proposed model also uses dropout technology, normalization and a rectified linear unit for accuracy improvement. Our results show that the proposed Hybrid CNN-LSTM Model outperforms traditional deep learning and machine learning techniques in terms of precision, recall, f-measure, and accuracy. Our approach achieved competitive results using state-of-the-art techniques on the IMDB movie review dataset and Amazon movie reviews dataset.

Keywords Natural Language Processing (NLP) · Sentiment Analysis · CNN · LSTM

✉ Ahmad Kamran Malik
ahmad.kamran@comsats.edu.pk

Extended author information available on the last page of the article

1 Introduction

The conceptual and psychological nature of natural languages increases complexity of text processing. There are two types of NLP applications. In first type of applications, the major concern is computational task like spell checker, machine translator, and grammar checkers. In second type of applications, linguistics aspect is more important and the major concern is resemblance to human languages, however, it also manipulate and recognize the psychological and theoretical knowledge. Sentiment Analysis (SA), poetry, story generation and intelligent information retrieval lie in second category [28]. SA is an important research area in NLP and machine learning. SA is the process of extracting and identifying subjective information (opnion) in a piece of text. Specifically, SA determines whether the writer's attitude towards an entity is negative, positive or neutral. Opinion mining is also called SA and it includes analysis of users' opinions, evaluations, sentiments, attitudes, appraisals, and emotions towards entities like organizations, products, individuals, services, topic, event, issues and their attributes. With the rapid and common use of social media on the internet, enormous data of users' review, brands, emotions, politics, and opinions is available on the web. Web provides useful information in the form of sentiments to readers, politicians, vendors etc. SA has recognized significant attention because it transforms unstructured reviews of users to useful information. SA is a text organization technique that is used to express feelings in different manners like negative, positive, dislike, like, thumb up, thumb down etc. There is a need to get useful information from huge amount of data using different machine learning techniques [1]. Participating in social media, defining the boundaries and channels of an individual's information flows, following, making friends, sharing, subscribing and forwarding tweets are a few types of user interaction practices and content that regulates how information flows through social media platforms. Through the use of these resources, anyone can contribute to a variety of traditional and other sources of information. Alternatively, social media users may also use these social platforms to recreate and strengthen traditional hierarchies by continuing to rely on few sources of information. The hierarchy of the network is a symbol of its unique flow of information [14]. In business, SA is a process of cataloging and identifying a slice of text according to the business [18]. This text can be in the form of feedback, tweets, comments etc. The organizations can promote their business using SA and can get the idea that how many users are satisfied with their products using the ratio of negative and positive tweets. The challenges in SA are parsing, labeling and named entity recognition (NER) which can be solved using machine learning and stat-of-the-art deep learning techniques [1].

We propose a hybrid model that combines and exploits recurrent, convolutional and global max-pooling layers on pre-trained Word2Vc. We use a very deep architecture of convolutional layers to extract local features of text. We also utilize long term memory concept of LSTM model and capture long term dependencies between sequence of words. For experiments, we used standard datasets of movies from amazon and IMDB and divided the dataset into a training set and testing set.

Our main contributions in this study are as follow.

1. Word embedding is created using Word2Vec model, which is an unsupervised model and is trained on a large collection of words. This model is able to capture semantics of words.
2. To capture sentiment polarity from texts, we used the LSTM model to detect deeper semantics of words. This model efficiently learne long-term dependencies between word sequences in long texts.

3. For further refinements of embeddings, we use a very deep CNN model on a supervised dataset. To generate a number of features, we also used many weight matrices with windows of different length.
4. We take advantages of the CNN model in extracting local features and long distance dependencies are captured by the LSTM model and combine these features into one single proposed hybrid CNN-LSTM model. Experimental results showed that our model achieves efficient results.

The rest of the paper is organized as follows. Section 2 describes the related work. Section 3 presents Word to Vector Model, the architecture of CNN and LSTM model. The methodology of our research work is described in Section 4. Section 5 describes experimental setup. Section 6 shows the results and discussion. Section 7 concludes the study.

2 Related work

A huge amount of data exists on social websites like Facebook, twitter etc. This makes SA a pretty challenging task and many issues arise during the processing of social media content [17]. Large amount of data is generated through Web on a daily basis which needs to be processed to obtain meanings from data. Machine learning and deep learning techniques have been used for SA [11, 29, 30]. In NLP field, researchers have been developing many different techniques to solve SA issues and these techniques use a bag of words representation [16]. Due to the self-adaptive, self-configurative, and self-aware nature of deep learning techniques, the research community is devoted to find out solutions to extract useful information by discarding the irrelevant and unnecessary data on the social media sites. We review the related work of SA in two categories i.e. SA with deep learning and machine learning. First, we describe the traditional approaches used for SA. The study [10] focused on the basic problem in SA which is sentiment polarity categorization. It uses products reviews dataset from amazon. Support Vector Machine (SVM), Random Forest (RF) and Naïve Bayesian (NB) techniques are used and produced better results. SA for the Thai language is used in [25]. Online community's reviews from pantip.com were used for SA with four sentiments which are negative, positive, neural and need. An unsupervised deep learning *paragraph2Vec* approach for feature extraction was proposed and applied that outperformed TF, TF-IDF, SVM and NB in terms of accuracy. In [24], author performed sentiment analysis for Thailand tweets using Logistic Regression. Experiment results showed that Logistic Regression performs with Paragraph2Vec well in terms of accuracy and time. In [9], authors compared SVM and NB techniques for Arabic tweets and text classification using WEKA tool. TF-IDF and cosine measure approaches were used for weighting scheme and similarity calculation among documents respectively. Experimental results show that NB performed well in terms of accuracy and time.

To predict the sentiments of visual content in visual SA, CNN framework approach has been proposed in [15]. For the experiment, back propagation is applied on the dataset of 1269 images which are collected from twitter. From results, the authors show that the proposed system acquired high performance in terms of recall, accuracy, and precision on twitter dataset and proposed GoogLeNet improved results by 9% over the AlexNet. The study [31] applied CNN on the micro-blogs comments to acquire the altitude, opinions of online users about special events. CNN technique was used as it overcomes the feature extraction and learns the data through training implicitly. A corpus of 1000 comments of micro-blogs was developed and divided into three different labels. Deep Belief Network with Feature Selection (DBNFS) has been proposed by the authors to overcome the vocabulary problems in [23]. Chi-Squared technique is used to enhance the learning phase of DBN to DBNFS. In experimental work, four different datasets were used for estimation in sentiment classification. Feature selection and reduction comparison have been done before and after the experiment to evaluate the accuracy of a proposed model. The experiment proved that DBNFS works better than DBN as training time of DBNFS is lower than DBN. In [22], a combination of CNN + Word2Vc framework has been proposed. Authors proposed seven layers model to improve generalizability and accuracy of the model to analyze the text in movie reviews dataset using Word2vec, Parametric Rectified Linear Unit (PReLU), Dropout technology and CNN model. Proposed model achieves 45.4% accuracy which is improved as compared to another neural network.

A combination of CNN and LSTM (ConvLstm) technique proposed by authors in [12]. Stanford Sentiment Tree Bank (SSTb) and IMDB datasets were used for experiments and achieved efficient results with less convolutional layers. The experimental study also proved that unsupervised pre-trained word vectors are an important feature for NLP in deep learning. Authors presented a new architecture for NLP in [2] to operate at character level directly and used small convolution and pooling operations to learn a high-level representation of sentences. Freely available eight datasets were used in the implementation of proposed VDCNN. Experimental results showed that increasing the depth of VDCNN up to 29 layers gradually increases the performance. Authors in [8] proposed LSTM based approach for product-based sentiment analysis. They implemented conditional random field classifier with bidirectional LSTM (Bi-LSTM-CRF) and aspect-based LSTM for polarity identification on Hotel's review dataset. The proposed approach achieved 39% improvements for aspect opinion target expression. A Two-Parse algorithm is proposed for product review analysis with approximate 7000 keywords training dataset in [19]. Proposed algorithm is an efficient solution to polarity problem in a dataset. Authors also proposed K nearest neighbor weighted (weighted K-NN) classifier which achieved higher accuracy as compared to existing K nearest neighbor classifier. Weighted K-NN classifier successfully classify the weekly and lightly polar reviews with high polar ones from online reviews of amazon.com, ebay.com, flipcart.com, etc. Proposed classifier provides an option to modify the parameters according to system requirements. Gini Index features selection approach with SVM is proposed in [27] to classify the sentiments of movies reviews data. Experimental results showed that proposed approach achieved better classification in terms of accuracy and reduced error rate. Authors proposed statistical technique also improves the accuracy of sentiment polarity in a big movies reviews dataset. In the study [4–6], the authors proposed the rating prediction recommended system using deep learning.

3 Background of deep learning model

3.1 Word to vector model

Word2Vc is deep learning model and it was proposed by Google in 2013. Word2Vc model creates vector numeric values using sentence of words. Based on word meaning Word2Vc compute the distance between words. Given the huge amount of data, usage, and content, Word2Vc can create exceedingly accurate estimates about a word's meaning. Therefore, Word2Vc runs fast even for a huge dataset. It uses google news dataset for training. The google news dataset contains pre-trained vectors. Dropout technique is used in our model to prevent from overfitting and to drop the irrelevant information from network to enhance the performance. Using dropout technique it selects top most related words from google news dataset are associated with “good”, “bad” and “terrible”. Negative words like “terrible”, “bad” and “horrible” disappear on one side of the graph, while positive words like “good” and “fantastic” appear in the second group. The DN demonstrate that Word2Vc can perfectly find the similar words in vector space. Input data of CNN cannot change in next layer and we use the same size of input data in next it means the input sentence contains same number of words.

3.2 Convolutional neural network model

The CNN is a special type of neural network and is employed from the field of image processing. However, CNN model has been effectively used in text classification. In CNN model, a subset of input to its preceding layers is connected using a convolutional layer that is why CNN layers are called feature map. The CNN model uses polling layer to reduce the computational complexity. The polling techniques in CNN reduce the output size of one stack layers to next in such a way that important information is preserved. There are many polling techniques available, however, max-polling is mostly used in which pooling window contains max value element. The flattened layer is used to feed the output of polling layer and maps it to next layers. The final layer in CNN typically is fully connected. Figure 1 shows the basic architecture of CNN.

3.3 Recurrent network model

Our proposed model uses LSTM which is a special type of Recurrent Neural Network (RNN). In RNN, neurons are connected with each other in the form of directed cycle. The RNN model processes the information in a sequential manner because it uses internal memory to process a sequence of words or inputs. RNN performs the same task for each element because output is dependent on all previous nodes inputs and remember information.

for further processing. The Eq. 1 represent general RNN model where *ht* is the new state at time *t, fw* is a function with *w* parameter, *ht-1* is an old state (previous state) and *xt* is input vector at time *t*.

$$h_t = f_w(h_t{-}1, x_t) \tag{1}$$

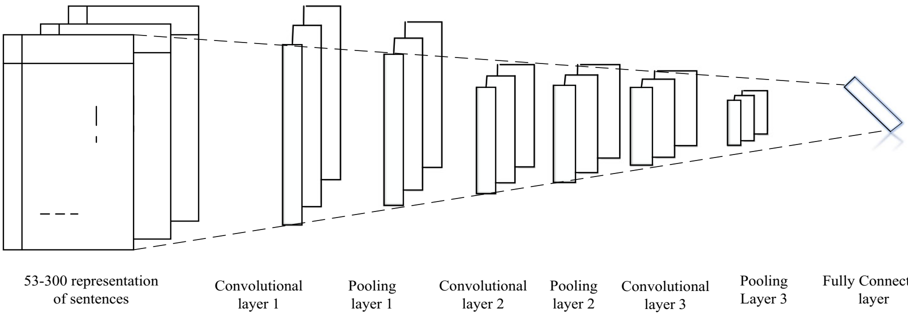

Fig. 1 The architecture of the CNN

We change the Eq. 1 to Eq. 2 that is used for assigning weights.

$$h_t = \tanh(W_{hh}h_1 + W_{xh}x_t) \tag{2}$$

In Eq. 2, *tanh* is the activation function, w_h is the weight of hidden state and x_t is the input vector. The exploding gradient or vanishing problem is created when learning of gradient algorithm is back propagated by the network. A special type of RNN model which is called Long Short-Term Memory (LSTM) is used to handle the vanishing gradient problem. The LSTM model saves long-term dependencies using three different gates in an effective way. The architecture of LSTM model is shown in Fig. 2. The structure of LSTM is chain like and it is similar to RNN, however, LSTM uses three gates to regulate and preserve information into every node state. The explanation of LSTM gates and cells is provided in Eqs. 3-6.

$$\text{Input Gate}\ \ In_t = \sigma(W_{in}.[hs_t{-}1], x_t + b_{in}) \tag{3}$$

$$\text{Memory Cell}\quad C_t = \tanh(W_c.[hs_t{-}1], x_t + b_e) \tag{4}$$

$$\text{Forget Gate}\quad f_t = \sigma\left(W_f.[h_{st}{-}1], x_t + b_f\right) \tag{5}$$

$$\text{Output Gate}\quad f_\circ = \sigma(W_o.[h_{st}{-}1], x_t + b_\circ) \tag{6}$$

In above equations, *b* represents the bias vector, *W* is used for weight and *xt* is the input vector at time *t*, where as *in*, *f*, *ct*, and *o* represent input, forget, cell memory and output gates.

4 Methodology

In this section, we show the details of our proposed model, which contains recurrent and convolutional neural network. Our model takes input as word embeddings and feeds

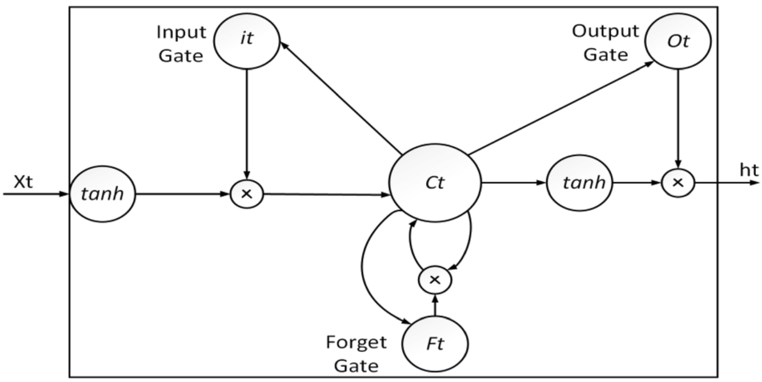

Fig. 2 The Architecture of LSTM Model

them into convolutional layers to extract local features. After that, output of convolutional model is given to an LSTM model to learn long-term dependences between the sequence of words and in the end a classifier layer is applied.

In this paper, we use a word to vector (Word2Vc) technique with CNN and LSTM model for SA. Deep learning methods cannot understand human text directly, therefore, firstly we use Word2Vc technique that translates the text into Word2Vc that takes string of sentence as input, transforms it into vector values and compares these values with other vector values to compute the distance of words that group the similar words in one cluster based on word's meaning. These vector values become the input of CNN model. Here, we describe the models that we used in our experiments. Our proposed hybrid CNN-LSTM model is described in Subsection 4.1 whereas Subsections 4.2 and 4.3 describe the details of our modeling used for CNN and LSTM respectively. We compare the results of three models with each other and with the traditional machine learning techniques.

4.1 The proposed Hybrid CNN-LSTM model

In Fig. 3, we show the main architecture of the proposed hybrid CNN-LSTM model. It takes a corpus as input and in pre-processing phase, it performs sentence segmentation, tokenization, stop word removal and stemming tasks. After this, it applies word embedding layer using Word2Vec. Convolutional layer extracts the high level features and LSTM layer detects long term dependencies between words. In the end, we apply classification layer using sigmoid function.

4.2 The proposed model using CNN

In this section, we describe the proposed CNN model that uses Word2Vc technique for word embedding. Firstly, Word2Vc translates the text into vector numeric values and then we apply CNN model to train vector numeric values. We use 3 pooling layers, 3 convolutional layers and one fully connected layer. We show the systematic diagram of CNN with 7 configuration layers. For experiments, we use tensor flow open source python library for numerical computation. In CNN, we use pooling layers, convolutional layers, dropout out layers and RLU for accuracy improvement. Dropout technique is proposed by Hinton in 2012 [23]. Dropout is an important trick in deep learning because it prevents machine learning algorithms from over-fitting. In backpropagation, dropout algorithms skip the neurons that do not contribute. The dropout technique drops the neurons during training to prevent neurons from co-adaptation. Each hidden neuron gives output with 0.5 probability. In following subsections, we describe the proposed CNN model with different layers which are used in the model.

4.2.1 Initializing CNN

Pre-Processing: Pre-processing is a process which helps to organize the dataset by performing basic operations on dataset before passing it to a model such as removal of spaces and meaningless word, converting different forms of a word into theirs roots words, and removal of duplicate words, etc. It converts the raw dataset into a useful and organized dataset for further use.

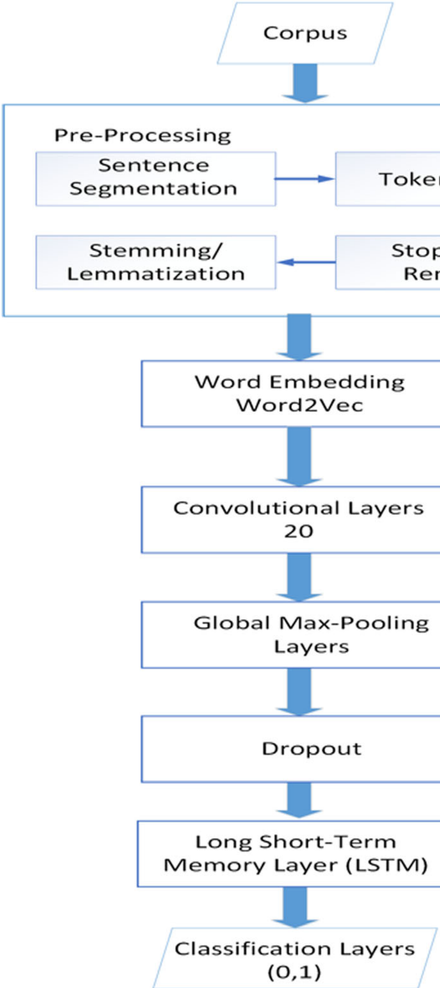

Fig. 3 Methodology of Proposed Hybrid CNN-LSTM Model

Embedding layer Pre-processed dataset provides a unique and meaningful sequence of words and every word has unique ID. Embedding layer initializes the words to assign random weights and it learns the embedding to embed all words in the training dataset. This layer is used in different ways and is mostly used to learn embedding of words that can be saved to use in another model. In this paper, we used pre-trained Word2Vc model for words embedding.

Convolution layer Our CNN model consist of seven layers; three convolutional layers, three pooling layers, and last one is the fully connected layer. Embedding layer passes the word in the form of sentences to convolutional layers. Convolution layer convolve the input using pooling layers, pooling layer helps reduce the representation of input sentences, input parameters, computation in the network and control the overfitting in the network.

Global max-pooling We applied global max-pooling at the end of network layers, it provides the global best results from the whole network after applying different convolution layers.

Activation Function We use RELU activation function in our model. RELU gives zero at negative values and it increases with positive values.

Dense layer Dense layer, also called fully connected layer, is used to perform classification on the extracted features of the convolutional layers. Using dense layer, every current input (neuron) in the layer of the network is connected to every input (neuron) in the proceeding layer of the network.

SoftMax SoftMax is a function that is mostly used in the final layer of the neural network. It takes the average of the random results into 1 and 0 form. Figure 4 shows the proposed model using CNN.

4.3 The proposed model using LSTM

In this section, we describe the techniques and layers used in the LSTM model.

Embedding layer We use Word2Vc model for words embedding. Embedding layer initializes the words to assign random weights and learns to embed all words in the training dataset.

LSTM layer After embedding, we use LSTM model for layers of RNN. LSTM uses three types of gates and cells for handling information flow in the network.

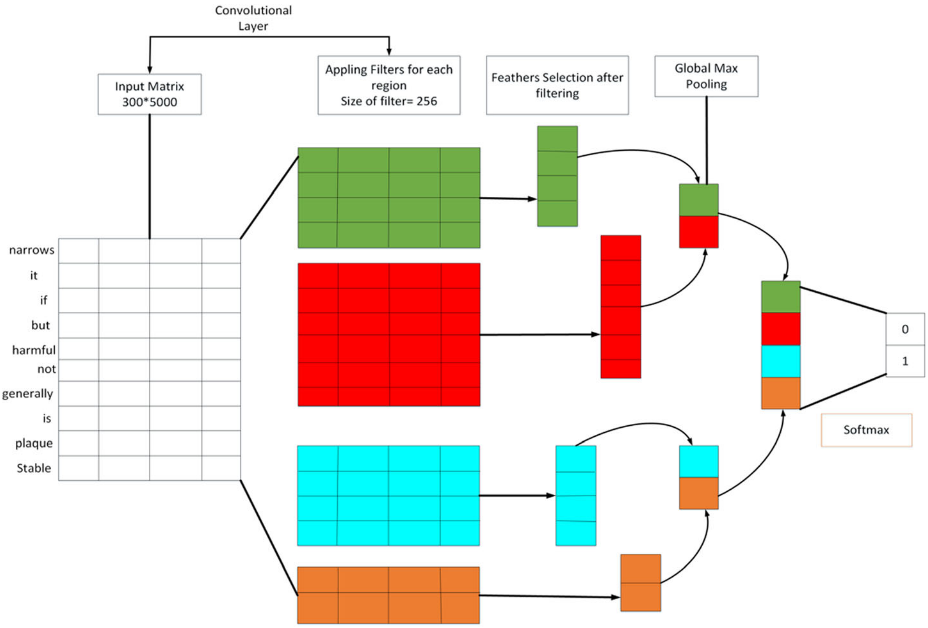

Fig. 4 Proposed model using CNN

Dropout Techniques We use dropout technique because it prevents our model from overfitting. It drops the irrelevant information from the network which do not contribute in further processing to enhance the performance of our model.

Dense Layer We use a dense layer in the proposed model. It connects each input with every output using weights.

SoftMax It is a function that is mostly used in the final layer of the neural network. It takes the average of the random results into 0,1 form. Figure 5 shows proposed model using LSTM.

5 Experimental Setup

We used two standard datasets for evaluation of our proposed Hybrid CNN-LSTM Model. One is the IMDB movie reviews dataset available on *http//rottentomatoes.com* and second is amazon movie reviews dataset available on *https://www.kaggle.com/bittlingmayer/amazonreviews*. Several experiments are performed using the proposed model on both datasets. Our approach obtained better results with high precision, recall, f-measure, and accuracy as compared to traditional machine learning algorithms i.e. Naïve Base, Support Vector Machine etc.

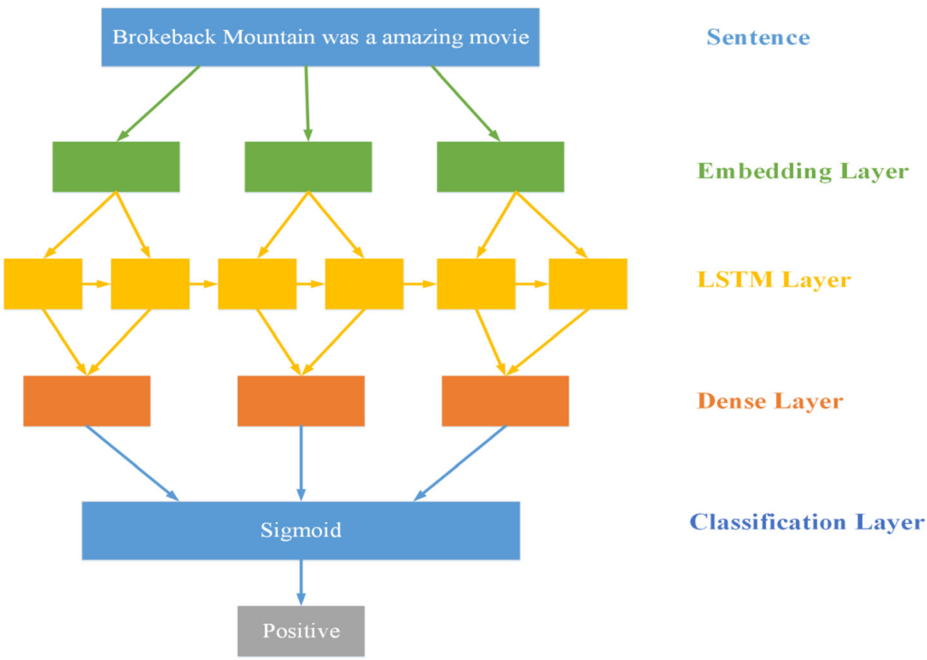

Fig. 5 Proposed model using LSTM

5.1 IMDB Movie Reviews Dataset

The benchmark IMDB dataset of movie reviews for sentiment analysis was first time published in [26]. This dataset contains 40,000 binary labeled reviews. We divide the dataset into 80:20 training and testing cases. The label distribution is balanced with each subset of data. For the validation set, we used 10% labeled from training documents.

5.2 Amazon Movie Review Dataset

In the start, we remove irrelevant HTML tags from the dataset and normalize the dataset. We perform pre-processing on the dataset that include tokenization, space removal, punctuation removal and irrelevant words as stop word. There are 2000 examples of movie reviews, half of which is negative and the other half positive in the original dataset. We used 1600 examples to train the model and it is tested by 400 examples. In the dataset, 1 represents positive comment and 0 represents negative comment about the movie. Deep learning model takes input in the vector forms as mentioned above and changes the text into vector using word2vec. The Table 1 shows parameter settings of proposed Hybrid CNN-LSTM model.

5.3 Evaluation Method

The performance of classification model is verified using standard evaluation metrics i.e. f-measure, recall and precision as shown in Eqs. 7-9. We used Adam optimizer to calculate the accuracy of proposed hybrid model.

$$Precision\ (P) = \frac{\text{True Positive}}{(\text{True Positive} + \text{False Positive})} \tag{7}$$

$$Recall\ (R) = \frac{\text{True Positive}}{(\text{True Positive} + \text{False Negative})} \tag{8}$$

$$F\text{−}measure = \frac{2*(Precision*Recall)}{(Precision + Recall)} \tag{9}$$

Where TP is the true positive, FP is the false positive and FN is the false negative. F-measure is the harmonic mean of precision and recall.

Table 1 Model parameters

Tuning Parameters	CNN Model	LSTM Model
Learning Rate	0.01	0.01
Dropout	0.2	0.2
Embed size	300	300
Step Size	20	20
No filters	256	256
Batch size	64	64

We used SGD approach to train our model. The Equations10 and 11 are used for weight updating.

$$V_j + 1 = 0.9 * v_j – 0.0005 * € * w_j – € * \left(\frac{d_l}{d_w} | w \right) D_j \tag{10}$$

$$W_j + 1 = w_j + v_j + 1 \tag{11}$$

€ is the learning rate, v is the momentum variable, and j is the iteration index.

6 Results and Discussion

We implemented two common deep learning models (CNN, LSTM) and proposed Hybrid CNN-LSTM Model on two datasets IMDB and Amazon. We performed many experiments on IMDB sentiment analysis dataset to attempt a candid comparison with competitive techniques. In our experiments, we pursued the experimental protocols as presented in [13]. In IMDB dataset, one movie review contains many sentences. We applied proposed hybrid CNN-LSTM model on IMDB dataset. We used word2vc technique to initialize the words as vector space and word2vec use skip gram and bag-of-words technique to convert the words in vector representation. We show the f-measure, recall and precision of our proposed hybrid CNN-LSTM model and different other techniques on two datasets in Fig. 6. The initial highlights of our results on IMDB dataset are that the proposed hybrid model improves the f-measure score upto 4-8% when compared with CNN and LSTM individually. Our hybrid model used 10 convolutional layers to extracts local information in an efficient way as compared to the networks proposed in [13, 21]. Figure 7 show the accuracy of the proposed hybrid CNN-LSTM model and traditional approaches (NB, SVM, GA). A machine learning hybrid approach NB-SVM performed better in term of accuracy, however, it was applied on a small dataset with more parameters.

We also performed many experiments on amazon movie reviews dataset and compared the results with traditional models. The Fig. 6 also shows the f-measure, precision and recall of many different models using Amazon movie review dataset. The results show that performance of our proposed deep learning models is better than traditional machine learning techniques. In Our model, we used dropout technique which improved the execution time. We observed that our proposed Hybrid CNN-LSTM model improved accuracy with respect to baseline algorithms [3, 21, 26]. Figure 7 shows the performance of the proposed hybrid model that outperformed traditional machine learning techniques in term of accuracy on the IMDB movie reviews dataset.

6.1 Overview

To develop an algorithm that can understand the hierarchal representation of the sentence in a text is the main challenge of NLP. Classification and feature extraction are considered as the combined task of CNNs. Recently, CNNs has been further improved [12, 15, 22, 23, 31] using multiple convolution and pooling layers to extract the sequential information from hierarchal input. Reducing the size of the network is the focus of several research studies. Authors in [20]

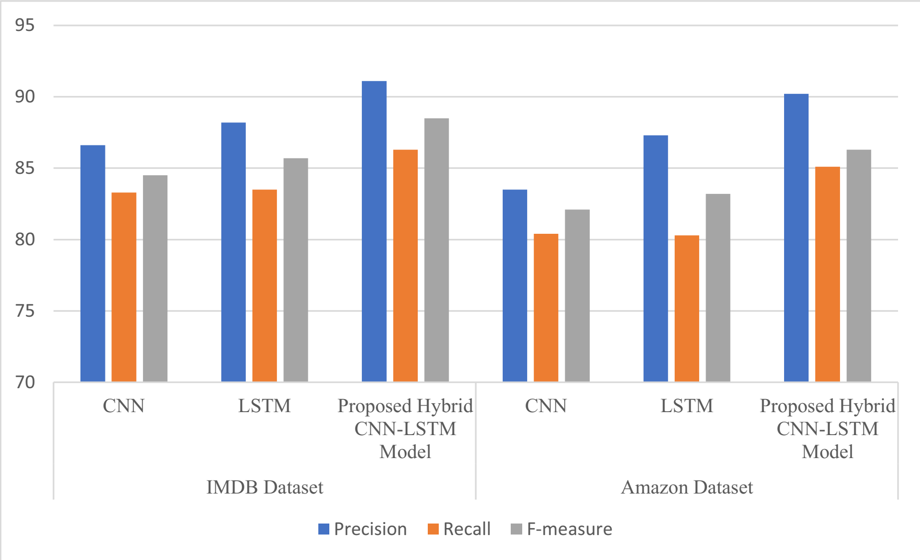

Fig. 6 Comparsion of Proposed Hybrid CNN-LSTM model with CNN and LSTM w.r.t Precision, Recall and F-measure on IMDB and Amozon Movie Reviews Dataset

replaced the layers with fully connecting layers using average pooping layer and removed redundant connections to allow weight sharing in simple network. In our study, we implemented both traditional and deep learning methods for the fair comparison of the performance on SA benchmark datasets. Weights are considered binary which reduce memory consumption [7]. Our approach efficiently performed as compared to model in [21]. We tried to select best architectures that deliver comparable results among existing techniques. In proposed Hybrid CNN-LSTM model, we used global max-pooling beside of simple max-pooling layer which produced better results in term of accauracy from basline appaorches. Our proposed approaches used less parameters which consume less memory and are efficient in term of convolution layers.

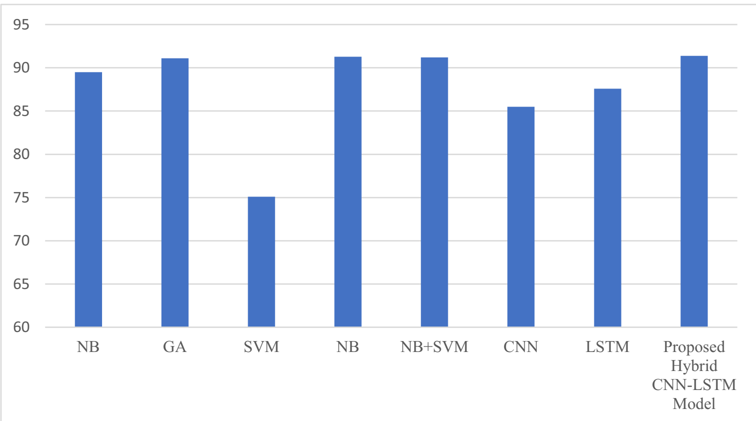

Fig. 7 Accuracy Comparsion of Proposed Hybrid CNN-LSTM model with traditional approaches on IMDB dataset

7 Conclustion

CNN helps to learn how to extract features from the data. However, it also requires many convolution layers to captures the long-term dependencies, capturing dependencies becomes worse with the increase of input sequence of length in a neural network. Basically, it leads towards a very deep layer of convolution neural networks. The LSTM model is capable to capture long-term dependencies between word sequences. In this study, we proposed a Hybrid CNN-LSTM model for sentiment analysis. The Proposed Hybrid CNN-LSTM model performed very well on two benchmark movie reviews datasets as compared to single CNN and LSTM models in terms of accuracy. The Proposed Hybrid CNN-LSTM model achieved 91% accuracy as compared to traditional machine learning and deep learning models.

Acknowledgements This work is funded by the COMSATS University Islamabad (CUI), Islamabad, Pakistan, CUI/ORICPD/19.

References

1. Ain QT, Ali M, Riaz A, Noureen A, Kamran M, Hayat B, Rehman A (2017) SA using deep learning techniques: a review. Int J Adv Comput Sci Appl, 8(6)
2. Al-Smadi M, Talafha B, Al-Ayyoub M, Jararweh Y (2018) Using long short-term memory deep neural networks for aspect-based sentiment analysis of Arabic reviews. International Journal of Machine Learning and Cybernetics, 1-13
3. Amolik A, Jivane N, Bhandari M, Venkatesan M (2016) Twitter sentiment analysis of movie reviews using machine learning techniques. Int J Eng Technol 7(6):1–7
4. Cheng Z, Ying D, Lei Z, Mohan K (2018) Aspect-aware latent factor model: Rating prediction with ratings and reviews. In: Proceedings of theWorld Wide Web Conference on World Wide Web, pp. 639-648
5. Cheng Z, Ying D, Xiangnan H, Lei Z, Xuemeng S, Mohan K (2018) A^ 3NCF: An Adaptive Aspect Attention Model for Rating Prediction. In IJCAI, pp. 3748-3754
6. Cheng Z, Xiaojun C, Lei Z, Rose C, Mohan K (2019) MMALFM: Explainable recommendation by leveraging reviews and images. ACM Trans Inf Syst 37(2):16
7. Collobert R, Weston J (2008) A united architecture for natural language processing: Deep neural networks with multitask learning, in Proc. 25th Int. Conf. Mach. Learn., pp. 160-167
8. Conneau A, Schwenk H, Barrault L, Lecun Y (2017) Very deep convolutional networks for text classification. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, pp. 1107-1116
9. Elghazaly T, Mahmoud A, Hefny HA (2016) Political sentiment analysis using twitter data. In Proceedings of the ACM International Conference on Internet of things and Cloud Computing, pp.11
10. Fang X, Zhan J (2015) Sentiment analysis using product review data. J Big Data 2(1):5
11. Govindarajan M (2013) Sentiment analysis of movie reviews using hybrid method of naive bayes and genetic algorithm. Inte J Adv Comput Res 3(4):139
12. Hao H (2014) Recursive deep learning for sentiment analysis over social data. In Proceedings of the 2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT), pp. 180-185
13. Hassan A, Mahmood A (2018) Convolutional Recurrent Deep Learning Model for Sentence Classification. IEEE Access 6:13949–13957
14. Himelboim I, Smith MA, Rainie L, Shneiderman B, Espina C (2017) Classifying twitter topic-networks using social network analysis. Social Media+ Society, 1-13
15. Islam, J, Zhang Y (2016) Visual sentiment analysis for social images using transfer learning approach. In IEEE International Conferences on Big Data and Cloud Computing (BDCloud) pp. 124-130
16. Kaur A, Gupta V (2013) A survey on SA and opinion mining techniques. J Emerg Technol Web Intell 5(4): 367–371
17. Li G, Hoi SC, Chang K, Jain R (2010) Micro-blogging sentiment detection by collaborative online learning. In IEEE 10th International Conference on Data Mining (ICDM), pp. 893-898

18. Liao S, Wang J, Yu R, Sato K, & Cheng Z (2017) CNN for situations understanding based on SA of twitter data. Procedia Computer Science, 376-381
19. Manek AS, Shenoy PD, Mohan MC, Venugopal KR (2017) Aspect term extraction for sentiment analysis in large movie reviews using Gini Index feature selection method and SVM. World wide web 20(2):135–154
20. McCallum A, Nigam K (1998) A comparison of event models for naïve Bayes text classi-cation, in Proc. AAAI Workshop Learn. Text Catego-rization, pp. 41-48
21. Ouyang X, Pan Z, Cheng H, Lijun L (2015) Sentiment analysis using convolutional neural network. In IEEE 2015 International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing, , pp. 2359-2364
22. Pang B, Lee L (2005) Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In Proceedings of the 43rd annual meeting on association for computational linguistics, pp. 115-124
23. Ruangkanokmas P, Achalakul T, Akkarajitsakul K (2016) Deep belief networks with feature selection for sentiment classification. In IEEE 7th International Conference on Intelligent Systems, Modelling and Simulation (ISMS), pp. 9-14
24. Sanguansat P (2016) Paragraph2vec-based sentiment analysis on social media for business in thailand. In IEEE 8th International Conference on Knowledge and Smart Technology (KST), pp. 175-178
25. Severyn A, & Moschitti A (2015) Twitter sentiment analysis with deep convolutional neural networks. In Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval pp. 959-962. ACM
26. Singh J, Singh G, Singh R (2017) Optimization of sentiment analysis using machine learning classifiers. Human-centric Comput Inf Sci 7(1):32
27. Srivastava A, Singh MP, Kumar P (2014) Supervised semantic analysis of product reviews using weighted k-NN classifier. In 11th IEEE International Conference on Information Technology: New Generations (ITNG), pp. 502-507
28. Syed AZ, Aslam M, Martinez-Enriquez AM (2010) Lexicon based SA of Urdu text using SentiUnits. In Mexican International Conference on Artificial Intelligence Springer, Berlin, Heidelberg pp. 32-43
29. Tripathy A, Agrawal A, Rath SK (2015) Classification of Sentimental Reviews Using Machine Learning Techniques. Procedia Comput Sci 57:821–829
30. Wang S, Manning CD (2012) Baselines and bigrams: Simple, good sentiment and topic classification. In Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics
31. Yanmei L, Yuda C (2015) Research on Chinese Micro-Blog Sentiment Analysis Based on Deep Learning. In IEEE 8th International Symposium on Computational Intelligence and Design (ISCID), pp. 358-361

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Anwar-ur-Rehman is a Masters students in Computer Science at COMSATS University Islamabad (CUI), Islamabad, Pakistan. His research interests include Machine learning, Deep learning and Social Networks.

Ahmad Kamran Malik received his Ph.D. from the Vienna University of Technology (TU-Wien), Austria. He is working as an Assistant Professor at COMSATS University Islamabad (CUI), Islamabad, Pakistan. Currently, his research interest is focused on Data Science, Social Network Analysis, and Information Security.

Basit Raza is working as Assistant Professor in the department of Computer Science, COMSATS University Islamabad (CUI), Islamabad, Pakistan. He received his Ph.D. (Computer Science) degree in 2014. He has published a number of conference and journal papers of internal repute. His research interests are Database management system, Security and Privacy, Data Mining, Data Warehousing, Machine Learning and Artificial Intelligence.

Waqar Ali is a MS students in Computer Science at COMSATS University Islamabad (CUI), Islamabad, Pakistan. His research interests include Machine learning, Deep learning and Social Networks.

Affiliations

Anwar Ur Rehman[1] • **Ahmad Kamran Malik**[1] • **Basit Raza**[1] • **Waqar Ali**[1]

Anwar Ur Rehman
it.anwaar@gmail.com

Basit Raza
basit.raza@comsats.edu.pk

Waqar Ali
waqarali199@ymail.com

[1] Department of Computer Science, COMSATS University Islamabad (CUI), Islamabad, Pakistan