

RESEARCH

Open Access



ConvLSTMConv network: a deep learning approach for sentiment analysis in cloud computing

Mohsen Ghorbani^{1†}, Mahdi Bahaghighat^{1*†} , Qin Xin² and Figen Özen³

Abstract

The rapid development of social media, and special websites with critical reviews of products have created a huge collection of resources for customers all over the world. These data may contain a lot of information including product reviews, predicting market changes, and the polarity of opinions. Machine learning and deep learning algorithms provide the necessary tools for intelligence analysis in these challenges. In current competitive markets, it is essential to understand opinions, and sentiments of reviewers by extracting and analyzing their features. Besides, processing and analyzing this volume of data in the cloud can increase the cost of the system, strongly. Fewer dependencies on expensive hardware, storage space, and related software can be provided through cloud computing and Natural Language Processing (NLP). In our work, we propose an integrated architecture of Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) network to identify the polarity of words on the Google cloud and performing computations on Google Colaboratory. Our proposed model based on deep learning algorithms with word embedding technique learns features through a CNN layer, and these features are fed directly into a bidirectional LSTM layer to capture long-term feature dependencies. Then, they can be reused from a CNN layer to provide abstract features before final dense layers. The main goal for this work is to provide an appropriate solution for analyzing sentiments and classification of the opinions into positive and negative classes. Our implementations show that found on the proposed model, the accuracy of more than 89.02% is achievable.

Keywords: Natural language processing, Deep learning, Opinion mining, Sentiment analysis, Cloud computing, Convolutional neural network, Long short-term memory network

Introduction

Nowadays, communication technologies [1–3] and computer networks have been deployed worldwide more than ever ([4]). Developing interesting technologies such as Software Defined Networks (SDN) [5], Cognitive Radio [6–8], and LiFi [2] along with emerging infrastructure such as Big Data, Cloud, and the Internet of Things (IoT) draw a very broad expansion of advanced data networks in the time ahead. In this condition, Sentiment Analysis (SA), also called Opinion Mining (OM) and Cloud computing

are the most useful steps to handle working with this giant volume of data that available.

Opinion Mining is a subset of Natural Language Processing (NLP) to build an intelligent system which can be used to review the information collected from different opinions and is a computational technique for studying people's opinions for automatic extraction and classification of emotions, attitudes towards an entity and sentiments from reviews. This is an ongoing field in research that can use text mining [9, 10]. In practical multimedia and machine learning applications [11, 12], it is more user-friendly to use speech/speaker recognition before text mining and NLP. In [12], a method for automatic recognition of the speaker was presented. This method

*Correspondence: m.bahaghighat@raja.ac.ir

[†]Mohsen Ghorbani and Mahdi Bahaghighat contributed equally to this work.

¹Department of Engineering, Raja University, Qazvin, Iran

Full list of author information is available at the end of the article

focused on its dependence on the text. They used the Mel Frequency Cepstrum Coefficients (MFCC) to extract feature vectors. These feature vectors were then applied by LBG Vector Quantization to obtain the codewords on the dataset and utilized the Dynamic Time Warping (DTW) technique for recognizing the speaker. In [13], the authors proposed the main goal for analyzing opinion and sentiment is to collect and analyze the reviews and also examine the sentiment scores obtained. That was divided into four critical levels: Document level, Sentence level, Word level, and Aspect level.

The rapid growth of the Internet and websites containing user reviews require expensive hardware to save, manage, and perform the computations. The big data on cloud computing is a fast-growing technology that has prepared itself for the computer industry by providing the space required for storage, software, hardware, and services [14]. In [13], they focused on the important challenges which have an effect on scores and polarity in sentiment at the sentiment evaluation phase. SA is one of the most active researches in NLP, and it is studied in many fields such as data mining, text mining and social sciences such as political science, communications, and finance. This is because opinions are so important in all human activities, and we often look for others' opinions whenever we need to make a decision [15].

In this paper, the main idea for Sentiment Analysis (SA) and Opinion Mining (OM) is based on deep learning algorithms with word embedding. At first, the features were extracted via a CNN layer and then these extracted features were sent to a bidirectional LSTM layer to learn long-term dependency. After completing these steps, we used another layer of CNN to begin learning the extracted features again. This makes the features easier to learn and the machine achieve better understanding about the classes. Obtained result showed the improved accuracy in our study. In fact, we used the CNN algorithm and the LSTM to improve the accuracy in the sentiment analysis. In our work, the word embedding was deployed for word representation. Subsequently, we considered this word representation with the polarity score feature as the set of sentiment features. This set of features were combined and fed into a CNN layer and an LSTM network layer. The second CNN layer is targeted to fulfill learning polarity in the text with the higher accuracy. Our proposed model, called *ConvLSTMConv*, is provided for a binary classification between negative or positive sentiment categories. The experimental results obtained shows that our method offers a classification method with acceptable performance. There are several challenges in this area, which can be pointed out as sarcasm. For example, people in positive words express their negative emotions. Word ambiguity is another challenge that makes polarization impossible because some words depend on the text. Some

words also have a multipolarity, and indicate a plurality. Fake opinion, also called fake review, refer to fake and negative comments about an object to undermine its credibility, which has become a major challenge. The rest of this paper is organized as follows:

In “[Related works](#)” section, we discuss the related work on this topic. Then, we elaborate our ConvLSTM-Conv model for sentiment classification in “[Methodology](#)” section. The process of experiment and simulation results are presented in “[Experiment and results](#)” section, and finally, in “[Conclusion and future works](#)” section, the conclusions and future works are presented.

Related works

Nowadays, many applications are deploying sentiment analysis (SA). A new model to carry out group decision making processes using free text and alternative pairwise comparisons was presented in [16]. It was designed to perform the SA via social networks, and it was one of the main advantages of the model. In [17], the authors applied sentiment analysis on the topic of tourism. The tourists usually are eager to share their experiences on a journey through social media. Sentiment classification with high accuracy is a major challenge, in the massive and irregular data.

In [18], a maximum entropy-PLSA (Probabilistic latent semantic analysis) model was introduced to extract emotion words. They used Wikipedia and corpus. In addition, the study of the impact of different text preprocessing steps on the accuracy of three machine learning algorithms, including Naive Bayes (NB), maximum entropy (MaxE), and support vector machines (SVM) for sentiment analysis was proposed in [19]. A computational model was presented in millions of movie reviews using machine learning and Naive Bayes classifiers on high-volume data and cloud computing execution in [20]. Furthermore, a deep CNN-based sentiment classification approach that can be used in Android applications was provided in [21]. It could classify reviews from various streaming services like Netflix and Amazon without needing server-side APIs.

Supervised learning is a subset of machine learning where the system attempts to learn a function from input to output. Supervised learning requires some input data in order to train the system. A supervised learning approach in human-annotated hotel reviews was deployed for ABSA (Aspect-Based Sentiment Analysis) tasks analysis in [22]. A set of lexical, morphological, syntactic, and semantic features was extracted to train classifiers as part of the targeted ABSA tasks. In [23], a new model called GloVe-DCNN with a sentiment feature set was proposed. It was a combination of word embedding and n-grams features and also polarity score features of sentiment words that combined and integrated into a

deep CNN. In [24], the authors presented a fuzzy-based strategy approach to building a general model to compute the polarity in texts into arbitrary domains, which took advantage of the possible conceptual domains' overlaps. The fuzzy logic [25] was used for representing the polarity learned from either training sets or a training set.

In [26], a novel multi-layer architecture for representing customer reviews techniques (including word embedding and compositional vector models) was presented and then integrated into a neural network and used a backpropagation algorithm for training a model for aspect rating prediction as well as generating aspect weights.

In [27], a new method of machine learning based on Minimum Cuts was proposed, which linked the text classification techniques to the subjective parts of the document to determine the polarity of sentiments. Their method first subdivided the subjective and objective words of the documents and dispensed the rest of the words for the next step. Then, a classification algorithm was applied to extract the result [27].

In the paper [28], presented by Kennedy and Inkpen, they have investigated datasets based on two methods of determining sentiments. In the first method, they examined the effect of valence shifters in the classification and used three types of shifters: negations, intensifiers, and diminishers. The second method of classification used was a machine learning algorithm, SVM. They first started with the unigram features and later applied the bigram features, which included valence shifters and the rest of the vocabulary, and also showed that combining the two methods produced better results. In [29], a weigh word-based features technique in binary classification tasks is used. The authors used words and phrases as features, and the values were assigns equal to their frequency or TFIDF score.

In paper [30], they presented a combination of unsupervised and supervised techniques for learning word vectors that provide information about capturing semantic term–document information as well as rich sentiment content.

In [31], the research focused on the effect of syntactic information in document-level sentiment. In their model, the classification using a convolutional kernel and reducing the complexity of kernels by extracting the minimum infrastructure with a high impact by a polarity dictionary was created. Studying and evaluating diverse linguistic structures encoded as a convolutional kernel for the document-level sentiment classification problem were done in order to use syntactic structures without defining explicit linguistic rules.

In [32], a study to create a two-stage sentiment polarity classification system using a reject option was presented. Their research was a combination of both Naive Bayes

(NB) classifier and Support Vector Machine (SVM)[33] models. They used a two-stages sentiment polarity classification using the rejection option to perform the sentiment classification in documents. In the first stage, an NB classifier, which was trained based on a feature representing the difference between numbers of positive and negative sentiment orientation phrases in a document review that deals with easy-to-classify documents. Remaining documents, that are detected by the NB classifier in use of rejection decision as “hard to be correctly classified”, and then secondly, are forwarded to process in an SVM classifier, where the hard documents are represented by additional bag-of-words and topic-based features [32]. Nguyen et al. [34] in 2014, also proposed a new rating-based feature in document level sentiment analysis by combining the features with unigram, bigram, trigram, and n-gram, then presented the results on the benchmark dataset published by Pang and Lee (2004). For first, they described a rating based feature which was based on regression model, and then learned from an external independent dataset with 233600 movie reviews, and then applied the learned model on the dataset from different domain and achieved state-of-the-art result with 91.6% for the sentiment polarity classification task. They used a supervised machine learning method to classify the polarity of sentiment at the document level. In addition to the N-Gram features, they used new rating-based features for training models. They rated the score rating of each document as a feature called RbF to learn the classification model and used the SVM model in LIBSVM3 to learn classification in dataset. At the first, the accuracy results of their method with SVM-based performances on the dataset was 87.6%. The accuracy based on RbF feature was 88.2%. By a combination of unigram and RbF features, the accuracy was 89.8%. The accuracy based on N-grams was 89.25% and finally by combining N-gram and RbF features, they reached a new state-of-the-art performance with 91.6%.

Methodology

The sentiment analysis is a subset of NLP and data mining. Whenever users visit a website to buy a product, they initially look at previous reviews for the same product category. A summary of this set of comments determines the buyer's opinion on that product. However, we need effective methods for categorizing sentiments in the documents. This is because the classification of the text involves the automatic sorting of a set of documents into specific categories from a predefined set. The sentiment analysis sometimes goes beyond the categorization of texts to find opinions and categorizes them as positive or negative, desirable or undesirable. There is a need for a classification tool or a system that can classify the sentiments of the text accurately because simple

text classification techniques are not sufficient to identify hidden parameters. The need for sentiment analysis increases due to the use of sentiment analysis in a variety of areas, such as market research, business intelligence, e-government, web search, and email filtering. Machine learning and deep learning algorithms are popular tools to solve business challenges in the current competitive markets.

Figure 1 describes the architecture of our proposed model for sentiment classification on texts. In our proposed model, at first, we modify the provided reviews by applying specific filters, and we use the prepared dataset by applying the parameters and implementing our proposed model for evaluation in the process step. The goal of this paper is to present a powerful method for binary classification. Discussions in more details are presented in the following:

Dataset

In this paper, we use the Movie Reviews (MR) dataset that was introduced by Pang et Lee in the literature [27]. It is a collection of a movie review with negative and positive texts where each review contains a sentence. Table 1 shows the details of the MR dataset.

Reviews preprocessing

Data preprocessing is an important step in data mining and machine learning projects [35–41]. The reviews are composed of incomplete sentences and contain much noise and wording with a weak structure like incorrect

grammar, imperfect words, and words without application with high repetition. Also, unstructured data affects the performance of sentiment classification. First, we need a series of preprocessing on the reviews to reduce the problems and have a regular structure. Cleaning data by applying filters, dividing the data into training and test sets, and creating data sets with preferred words are some steps that have been done in our work. Without going into much detail, we prepared the data using the following method:

- Applying character filtering: Removal of all punctuation from each word in the reviews and also remaining tokens that are not alphabetic.
- Filtering out stop words: Stop words do not contain useful information in the field of sentiments for analysis, so those were deleted to modify the dataset.
- Filtering out short tokens: Removing all words with fewer repetitions. We used the words that were repeated more than once in the reviews.

In the next step, we divide the data into training and test sets. We used 100 positive reviews and 100 negative reviews as the test set (200 reviews) and the remaining 1800 reviews as the training dataset. This is a 90% split of training data and 10% of test data.

Processing

Figure 2 describes the architecture of our proposed model for evaluating sentiment analysis. In this section, more details in the context of various deep learning algorithms are discussed.

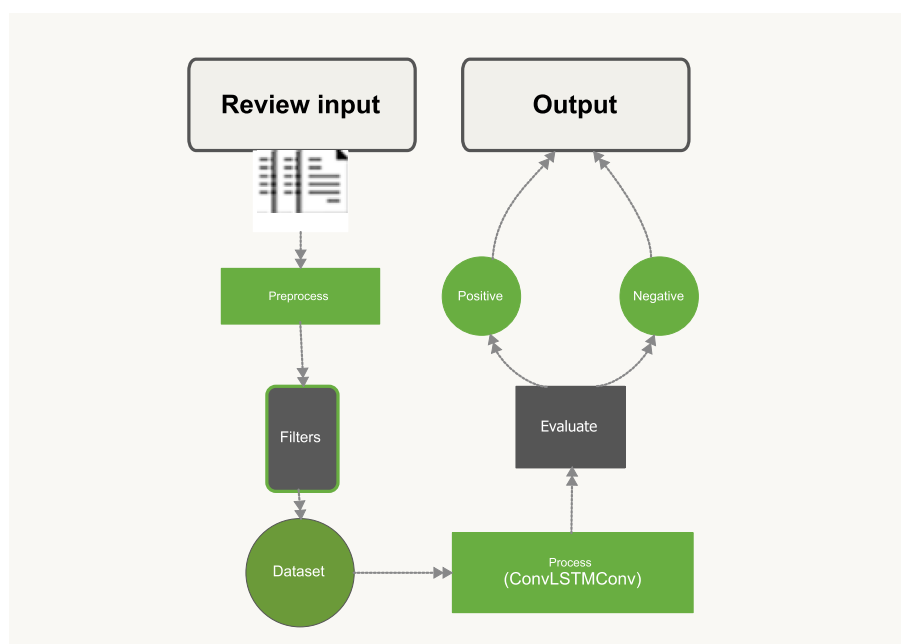


Fig. 1 General block diagram for our ConvLSTMConv proposed model

Table 1 The details of MR dataset [27]

Dataset	Number of negative		Number of positive		Number of review
	Train	Test	Train	Test	
MR2004	900	100	900	100	2000

Neural network architecture

Artificial Neural Network (ANN) architectures were widely used in the literature ([15, 42, 43]). Figure 3 shows a simple feed-forward NN with 3 layers as the input layer (L_1), hidden layer (L_2), and output layer (L_3). There is also a connection between two neurons that has a parameter called weight and is represented by w and applied to calculate the output.

Deep learning (DL), as a new generation of ANN, is a subset of a broader family of machine learning found on ANN. It can learn how to perform tasks using multi-layer deep networks and enhance the power of learning of NNs [15].

In [43], it has been stated that NN was introduced for the first time in the field of language modeling based on Markov’s assumption. For example, the probability of the sequence of the word W_1^N is decomposed as:

$$p(w_1^I) = \prod_{i=1}^I p(w_i | w_{i-n+1}^{i-1}) \tag{1}$$

And a trigram feed-forward NN was proposed that contains equations as follows:

$$y_i = A_1 \hat{w}_{i-2} \circ A_1 \hat{w}_{i-1} \tag{2}$$

$$z_i = \sigma(A_2 y_i) \tag{3}$$

$$p(c(w_i) | w_{i-2}, w_{i-1}) = \varphi(A_3 z_i) | c(w_i) \tag{4}$$

$$p(w_i | c(w_i, w_{i-2}, w_{i-1})) = \varphi(A_{4,c(w_i)} z_i) | w_i \tag{5}$$

Where \hat{w}_{i-2} and \hat{w}_{i-1} in Eq. 2 are one-hot encoded predecessor words for w_{i-2} and w_{i-1} and A_1 is the weight matrix that applies to all and then two vectors are concatenated to build activation layer y_i .

A standard NN that is inspired by the biological structure of the brain consists of information processing units called neurons and are used in different layers. Input neurons are activated through the sensor of peripheral perception (sensors perceiving the environment), and other neurons are activated by the weighting connections of the previously active neurons [15, 42]. A neural network for learning should provide a set of values for weights between neurons using the information flowing through them. Each neuron reads the neuron’s output in the previous layer and processes the information it needs, and produces the outputs for the next layer [15].

The general formula is the following, where b is the BIAS; weights of connections are w_i , f is a nonlinear activation function (AF).

$$f(W^t x) = f\left(\sum_{i=1}^3 W_i x_i + b\right) \tag{6}$$

The most common activation functions are Sigmoid function, hyperbolic tangent function (Tanh), and rectified linear function (ReLU). Their formulas are as follows:

$$f(W^t x) = Sigmoid(W^t x) = \frac{1}{1 + \exp(-W^t x)} \tag{7}$$

$$f(W^t x) = tanh(W^t x) = \frac{e^{W^t x} - e^{-W^t x}}{e^{W^t x} + e^{-W^t x}} \tag{8}$$

$$f(W^t x) = Relu(W^t x) = \max(0, W^t x) \tag{9}$$

The Sigmoid function receives a value range between 0 and 1, and a real-valued number as the firing rate of a neuron: 0 for not firing or 1 for firing. The hyperbolic tangent functions as a zero-centered output range and uses $[-1, 1]$

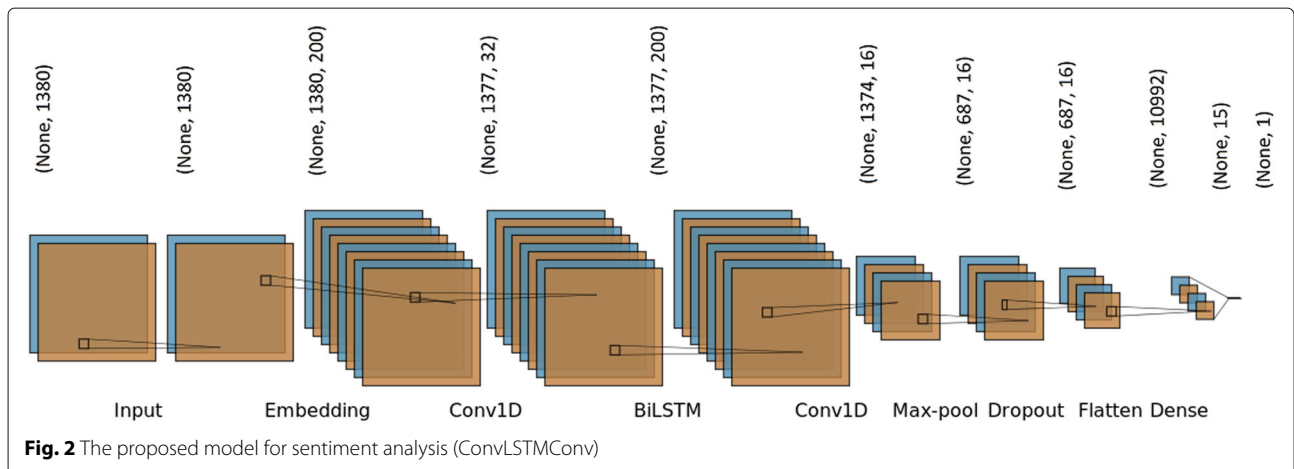
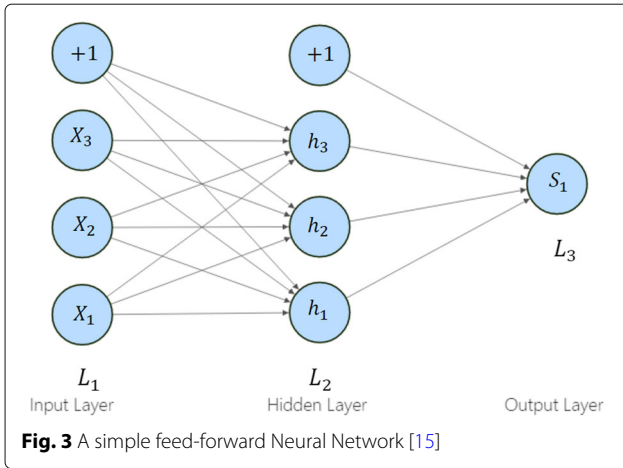


Fig. 2 The proposed model for sentiment analysis (ConvLSTMConv)



Instead of $[0, 1]$. For Relu function, if the input is less than 0, its activation will be thresholded at zero.

The Softmax function is used as the output neuron and is a logistic function. The function definition is as follows:

$$\sigma(x)_j = \frac{e^{x_j}}{\sum_{k=1}^K e^{x_k}} \text{ for } j = 1, \dots, K \quad (10)$$

In general, Softmax is usually used for the final classification at the final layer of a NN.

Convolutional layer architecture

CNN is a kind of feed-forward neural network used in deep learning, which was originally used in computer vision and included a convolutional layer to create the local features and a pooling layer for summarizing the representative features [44, 45].

Convolution layers in the artificial neural network play the role of a feature extractor that extracts the local features. This means that CNN establishes the specific local communication signals using a local connection pattern between neurons in the adjacent layer. Such a feature is useful for classifying in NLP, as it is expected that strong local clues should be found for the class, but these clues may appear in different places at the input. The convolutional and pooling layers allow CNNs to find local indicators, regardless of their location.

LSTM layer architecture

One of the presented models in a recurrent neural network (RNN) is a Long short-term memory (LSTM) network that can learn long-term dependencies. Some problems, such as gradient vanishing and exploding problems in the standard RNN was a reason to develop the LSTM model as a good solution [46]. The standard LSTM network has an architecture with an input layer that is connected to the LSTM layer. It contains the recurrent connections that are connected from the cell output units to the cell input units, input gates, output gates, forget

gates, and then cell output units are connected to the output layer [47]. We can calculate them as in the following equations [47]:

$$W = n_c \times n_c \times 4 + n_i \times n_c \times 4 + n_c \times n_o + n_c \times 3 \quad (11)$$

The number of the memory cell is n_c , n_i is equal to the number of input units; the number of output units is n_o .

For computations of the LSTM network, there is a mapping of an input sequence as $x = (x_1, \dots, x_T)$ and an output sequence $y = (y_1, \dots, y_T)$ with the activations, using the following formulas:

$$i_t = \sigma(W_{ix}x_t + W_{im}m_{t-1} + W_{ic}c_{t-1} + b_i) \quad (12)$$

$$f_t = \sigma(W_{fx}x_t + W_{fm}m_{t-1} + W_{fc}c_{t-1} + b_f) \quad (13)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot g(W_{cx}x_t + W_{cm}m_{t-1} + b_c) \quad (14)$$

$$o_t = \sigma(W_{ox}x_t + W_{om}m_{t-1} + W_{oc}c_t + b_o) \quad (15)$$

$$m_t = o_t \odot h(c_t) \quad (16)$$

$$y_t = W_{ym}m_t + b_y \quad (17)$$

Table 2 shows all the variables with their descriptions that have been used in the above formulas. With the proposed LSTM architecture with both recurrent and non-recurrent projection layers, the equations are as follows:

$$i_t = \sigma(W_{ix}x_t + W_{ir}r_{t-1} + W_{ic}c_{t-1} + b_i) \quad (18)$$

$$f_t = \sigma(W_{fx}x_t + W_{fr}r_{t-1} + W_{fc}c_{t-1} + b_f) \quad (19)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot g(W_{cx}x_t + W_{cr}r_{t-1} + b_c) \quad (20)$$

$$o_t = \sigma(W_{ox}x_t + W_{or}r_{t-1} + W_{oc}c_t + b_o) \quad (21)$$

$$m_t = o_t \odot h(c_t) \quad (22)$$

$$r_t = W_{rm}m_t \quad (23)$$

$$p_t = W_{pm}m_t \quad (24)$$

$$y_t = W_{yr}r_t + W_{yp}p_t + b_y \quad (25)$$

Table 2 Variables and their description

Variables	Description
W	weight matrix
W_{ix}	the matrix of weights from the input gate to the input
b	bias vector
b_j	the input gate bias vector
σ	the logistic Sigmoid function
i	input gate
f	forget gate
o	output gate
c	cell activation vector m
g	the cell input activation functions
h	the cell output activation functions
\odot	the product of the vectors
r	the recurrent unit activations
p	optional non-recurrent unit activations

Pooling layer architecture

The Pooling layer is one of the most widely used elements in CNN. One of its applications is dimension reduction for abstract representation, reducing the number of parameters that are used and consequently reducing the computation time of models. One of the most common models of pooling structure is called Max pooling. We use this pooling layer after the convolutional layer, and its filter size is usually set to 2×2 pixels ([48, 49]).

In [50, 51], Max pooling is defined as a downsampling operation where the application is the extraction of the most important features. The Max pooling layer is a layer that takes the input feature and converts to a feature with lower dimensions, and the Max pooling is calculated using the equation below:

$$rs_i = \max([h_j]_i, \dots, [h_{n-k+1}]_i), \quad (26)$$

where $[h_j]_i$ shows the i_{th} element in the vector h_j .

$$x_{p,i,j}^n = f(\max_{0 \leq u,v \leq M_n-1} X_{p,iS_n+u,jS_n+v}^{n-1}) \quad (27)$$

In Equation (27), it is used without weight. Node (i, j) are also connected to the input nodes in an $M \times M$.

Experiment and results

In this paper, we used deep learning algorithms such as CNN and LSTM using Python and Keras environment for sentiment analysis. We used the word embedding layer, called GloVe, a pre-trained word vectors, and an unsupervised learning algorithm, to obtain vector representations for words.

Experimental environment

We evaluated our ConvLSTMConv-based binary classification model on the MR2004 database [27]. MR2004 contains 2000 reviews in negative and positive polarities and each of which has 1000 samples. Some examples are presented in Fig. 4. For fair evaluation, we chose the training and the test sets as the same for preprocessing. The training and the test sets contain 90% and 10% of total samples, respectively. We described the training criteria and improvement techniques in the previous section. These training criteria and improvement techniques can be combined in various ways. In all experiments, we trained in the mini-batch mode with size 8.

We conducted our experiments on Google services. We used Google Drive to store our dataset, which is a cloud-based file storage service provided by Google, and allows users to store files on the servers and share files. We also used the Google Colaboratory system for our work which, is a free cloud service from Google for AI developers that supports Jupyter notebooks. In Google Colaboratory, we can use Python with additional libraries such as Keras, OpenCV and etc., to develop deep learning applications.

Numerical environment

In this section, we elaborate more details about numerical values of parameters and also hyperparameters in our proposed models and the results. We define a model with an

```
reporter love story two teenagers one discovered comet team astronauts
ship messiah save world firstly nothing outstandingly inferior making
film anything outstandingly good plot holes make film corny stupid honest
moved trailer film isnt saying much mimi leders followup peacemaker
equally incompetent big stars wasted perhaps im annoyed release
peacemaker us overshadowed far superior thriller assignment obvious title
represents big boom result collision also connotes heavy impact human
lives however film simply fails note effects wornout substandard
screenplay limited acting director continued sad run terms goodfilmmaking
credentials shes still making good money though leonis unfortunate
character news reporter foundation story cast film suffers many
characters need explored robert duvalls aging astronaut lifeless morgan
freemans president restricted well righteous president means hes
interesting leonis character appealing one played reasonable conviction
rather peculiar showing reporting msnbc definitely undervalued director
```

Fig. 4 A simple example of a test file from MR2004 [27] after preprocessing

input channel for processing the movie review text. The channel consists of the following elements:

- The input layer that specifies the length of the input sequences
- Embedding layer that is regulated to 100-dimensional vocabulary size
- Two convolution1D layers with separate filters and kernel size
- A bidirectional GRU layer
- A convolution1D layer with filters and kernel size
- MaxPooling1D layer to stabilize output from the convolution layer
- Dropout layer with $p=0.4$
- Flatten layer to reduce 3D output to 2D for concatenation

This channel reaches into a single vector and is processed by a dense layer with 15 neurons and Softmax activation function and an output layer with one neuron and Sigmoid activation function. More details are as follows:

After the input layer, this model uses an embedding layer as the hidden layer. The embedding layer requires vocabulary size, real value vector space size, and the maximum length of input documents. We used a pre-trained Glove model with a 100-dimensional vector space. Then we used two convolution1D layers with 64 and 32 filters respectively with kernel size 4 and Relu activation function, a bidirectional GRU layer with 80 neurons, and another convolution1D layer with the filter size 16, kernel size 4, and Relu activation function.

By changing the parameters, we are looking for better results. By reducing the two convolution layer to a single layer with a filter of 32 and changing the bidirectional GRU layer to a bidirectional LSTM layer with 50 neurons and a dropout with $p = 0.2$, we achieved a

more acceptable result than before. The model details are as follows:

- The input layer
- Embedding layer that is regulated to 100-dimensional vocabulary size
- A convolution1D layer with filters of 32, kernel size of 4 and Relu AF
- A bidirectional LSTM layer with 50 neurons
- A convolution1D layer with filters of 16, kernel size of 4 and Relu AF
- MaxPooling1D layer with a pool size of 2
- Dropout layer with $p=0.2$
- Flatten layer

This channel reaches into a single vector and is processed by a dense layer with 15 neurons and Softmax activation function and an output layer with one neuron and Sigmoid activation function, same as before. Running this model after 100 epochs, in the best performance of this model, was achieved in the training dataset with 89.17% accuracy and 83.00% in the test dataset, which is higher than the previous model.

The final changes in the next model, and the change in the parameters yielded a more acceptable result than the previous models. The detail of the proposed network structures is presented in Table 3.

After applying this model, the training dataset was achieved 89.02% accuracy and 89.02% accuracy in the test dataset. The best results were obtained with these modifications compared to previous models. The experiment results for several steps are presented in Table 4. The details about the hyperparameters are as follows:

- The input layer
- Embedding layer that is regulated to 200-dimensional vocabulary size

Table 3 Structure of our proposed model

Layers	Type	#of feature map	Feature map size	Window size	#of parameters
E1	Embedding	200	—	—	8,855,400
C2	Convolution	32	32×32	4×4	25,632
L3	Bi-LSTM	200	—	—	106,400
C4	Convolution	16	16×16	4×4	12,816
P5	Max pooling	16	—	2×2	0
D6	Dropout	16	—	—	0
F7	Flatten	—	—	—	0
D8	Dense	15	—	—	164,895
D9	Dense	1	—	—	16
Total parameters					9,165,159
Trainable					9,165,159
Non-trainable					0

Table 4 Results of train and validation in ConvLSTMConv

Epochs	Loss		Accuracy	
	Train	Validation	Train	Validation
1	0.9886	0.9755	0.5006	0.5000
2	0.9635	0.9516	0.5222	0.5500
3	0.9407	0.9298	0.5244	0.5350
4	0.9197	0.9098	0.5556	0.5300
5	0.9004	0.8914	0.5250	0.5300
...
20	0.7363	0.7361	0.5872	0.5700
21	0.7258	0.7251	0.6117	0.5950
22	0.7131	0.7127	0.6328	0.6150
23	0.6931	0.6879	0.6689	0.6550
24	0.6775	0.6822	0.6794	0.6550
...
50	0.3586	0.4461	0.8711	0.8250
51	0.3564	0.4691	0.8822	0.8050
52	0.3867	0.4257	0.8650	0.8400
53	0.3545	0.3726	0.8902	0.8902

- A convolution1D layer with filters of 32, kernel size of 4 and Relu AF
- A bidirectional LSTM layer with 100 neurons
- A convolution1D layer with filters of 16, kernel size of 4 and Relu AF
- MaxPooling1D layer with a pool size of 2
- Dropout layer with $p=0.35$
- Flatten layer

This channel reaches into a single vector and is processed by a dense layer with 15 neurons and Softmax activation function with L1 and L2 regularizers. The Combined regularization method is deployed for optimization and reducing overfitting with $l2 = 0.01$ (kernel regularizer) and $l1 = 0.001$ (activity regularizer). The output layer also designed with one neuron and Sigmoid activation function, as said before. For the compilation step, we used Stochastic Gradient Descent (SGD) as the optimizer input parameter with a learning rate of 0.09, the decay of 0.0009, and momentum of 0.8.

Figure 5 shows the accuracy and loss functions for our proposed model in sentiment analysis, respectively. Finally, Table 5 compares our best achievement with previous works on MR2004. The obtained results indicate that our proposed model based on ConvLSTMConv outperforms other approaches.

Conclusion and future works

The main goal of sentiment analysis for the market prediction is the recognition of customer's opinion about the

available products. It can pave the way for improvement and prevent future defects and flaws. In this paper, we presented a simple model for analyzing sentiment and opinions, which includes determining the positive and negative sentiments of the films. Our proposed model includes preprocessing on raw texts, feature extraction, and classification methods for classification and analysis. The preprocessing section is an important part that includes the correction of problems such as incomplete sentences, weak grammatical words, and words without application with high repetition for sentiment analysis that have a profound effect on classification performance. Applying changes to these raw texts can improve the results.

In our work, a word embedding model for word representation and a combination of feed-forward neural networks models (the CNN) and recurrent models (the LSTM) with parametric changes for sentiment analysis are presented. We examined our experiments through storage on Google Cloud and computing on Google Colaboratory. In our proposed model, feature learning and training were combined in one step. While many researchers are focusing on very deep and complex architectures for different tasks, we have deployed two CNNs in combination with an LSTM layer. In this work, we have implemented a binary classification model for analyzing sentiments in texts at different stages with varying parameters and optimizing it several times, to get performance improvement as much as possible. At the beginning of our work we used the layers of Conv, GRU and Conv and we were able to obtain acceptable results by parametric optimization. With deploying our unique strong proposed structure as Conv-LSTM-Conv, and optimization of parameters shown in Table 3 and Fig. 2, we were able to achieve a result of 89.02% with low number of epochs and the minimum time required.

The best result on the Pang and Lee dataset (2004) was obtained by Nguyen et al., which is an empirical study for the sentiment polarity classification. They used rating-based features based on a linear regression model from external independent dataset with 233600 movie reviews, and then checked rating-based and N-gram features into a machine learning-based approach to the Pang and Lee dataset. Because they used the N-Gram features and 10 fold cross validation model, it requires complexity and high execution time to run the training on a stand-alone database. Our proposed model is more suitable and applicable to the design of embedded and mobile systems because of the simplicity of the model and the speed of execution and the acceptable results obtained.

This article focused only on the sentiment classification into two classes of positive and negative class (binary classification) but the SA is not limited just to

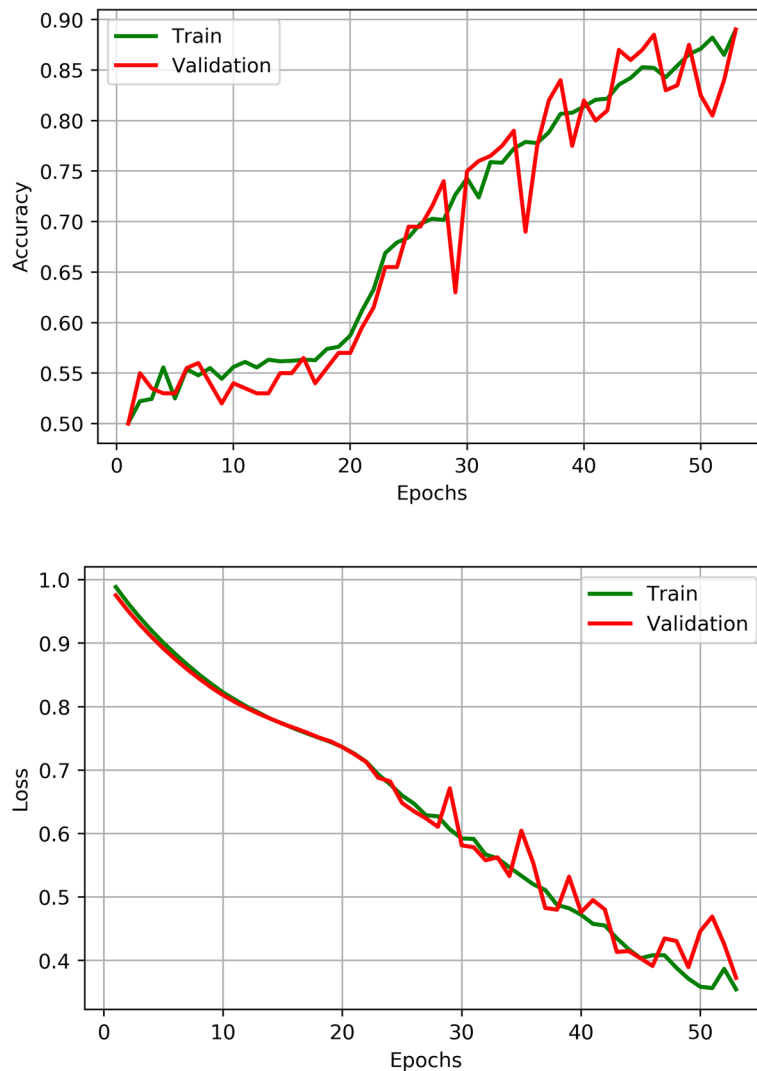


Fig. 5 Accuracy and loss function of our proposed model for sentiment analysis

the determination of positive and negative polarity. In real world, there are different situations such as happiness, anger, hatred, sadness and so on which can affect the opinion of reviewers. There are many influential factors in this area, some of which are fleeting and affecting opinions

at that moment. Some comments may also include sarcasm that makes it difficult to achieve the right result. Automatic generation of coherent and meaningful text using some advanced deep learning approaches such as generative adversarial network (GAN) in particular with

Table 5 Comparison among our proposed model and previous works

Author	Accuracy%	Method Description
Pang and Lee[27]	87.20%	Subjective summarization based on minimum cuts
Kennedy and Inkpen [28]	86.20%	Contextual Valence Shifters
Martineau and Finin [29]	88.10%	TFIDF Weighting
Maas et al. [30]	88.90%	A mix of unsupervised and supervised techniques to learn word vectors
Tu et al.[31]	88.50%	Word embedding using vector kernel + tree-based word dependency integrated with grammar relations
Nguyen et al.[32]	87.95%	Two-step classification of support vector machine and Naive Bayes
Our proposed model	89.02%	The combination of the convolution layer and the LSTM layer and the convolution layer (ConvLSTMConv)

emphasizing on conditional text are targeted as our future works. With using conditional text GAN, we will be capable not only to create our synthetic datasets but also we can customize it for more complex SA classification problems. It is hoped that we will focus on these factors in future work and take an effective step in improving the accuracy results.

Abbreviations

ABSA: Aspect-based sentiment analysis; ANN: Artificial neural network; CNN: Convolutional neural network; Conv: Convolutional; DCNN: Deep convolutional neural network; DL: Deep learning; GloVe: Global vectors; LSTM: Long short-term memory; MaxE: Maximum entropy; MR: Movie reviews dataset; NB: Naive Bayes; NLP: Natural language processing; NN: Neural network; OM: Opinion mining; ReLU: Rectified linear function; RNN: Recurrent neural network; SA: Sentiment analysis; SGD: Stochastic gradient descent; SVM: Support vector machines

Acknowledgements

Not applicable.

Authors' contributions

M.Bahaghighat; methodology, formal analysis, investigation, validation, writing–review and editing scientific papers, project administration, M.Ghorbani; information gathering, methodology and software, algorithm developer, writing–review and editing, formal analysis, investigation, validation, visualization, QXin; formal analysis, investigation, validation, writing–review and editing scientific papers, and information gathering, F.Ozen; formal analysis, investigation, validation, writing–review and editing scientific papers. The author(s) read and approved the final manuscript.

Authors' information

Mohsen Ghorbani got MSc in Computer Engineering from Raja University. He is currently the AI researcher in Artificial Intelligence in Science and Technology (AIST) Lab. Mahdi Bahaghighat got his PhD from the Electrical Engineering Department of AmirKabir University of Technology (AUT) in 2017. He is working as the Assistant Professor and chairman of the electrical engineering group at Raja University. He is the head of Artificial Intelligence in Science and Technology (AIST) Lab. His current research interests include Signal, Image and Video Processing, Computer Vision, Artificial Intelligence, Machine Learning, Deep Learning, Sensor Networks, and Wireless Multimedia Transmission. Qin Xin graduated with his Ph.D from University of Liverpool, UK in December 2004. Currently, he is working as a Full Professor of Computer Science in the Faculty of Science and Technology at the University of the Faroe Islands (UoFI), Faroe Islands. Prior to joining UoFI, he had held variant research positions in world leading universities and research laboratory including Senior Research Fellowship at Universite Catholique de Louvain, Belgium, Research Scientist/Postdoctoral Research Fellowship at Simula Research Laboratory, Norway and Postdoctoral Research Fellowship at University of Bergen, Norway. His main research focus is on design and analysis of sequential, parallel and distributed algorithms for various communication and optimization problems in wireless communication networks, as well as cryptography and digital currencies including quantum money. Moreover, he also investigates the combinatorial optimization problems with applications in Bioinformatics, Data Mining and Space Research. Currently, Prof. Dr. Xin is serving on Management Committee Board of Denmark for several EU ICT projects. Prof. Dr. Xin has published more than 70 peer reviewed scientific papers. His works have been published in leading international conferences and journals, such as ICALP, ACM PODC, SWAT, IEEE MASS, ISAAC, SIROCCO, IEEE ICC, Algorithmica, Theoretical Computer Science, Distributed Computing, IEEE Transactions on Computers, Journal of Parallel and Distributed Computing, IEEE Transactions on Dielectrics and Electrical Insulation, and Advances in Space Research. He has been very actively involved in the services for the community in terms of acting (or acted) on various positions (e.g., Session Chair, Member of Technical Program Committee, Symposium Organizer and Local Organization Co-chair) for numerous international leading conferences in the fields of distributed computing, wireless communications and ubiquitous intelligence and computing, including IEEE MASS, IEEE LCN, ACM SAC, IEEE ICC, IEEE Globecom,

IEEE WCNC, IEEE VTC, IFIP NPC, IEEE Sarnoff and so on. He is the Organizing Committee Chair for the 17th Scandinavian Symposium and Workshops on Algorithm Theory (SWAT 2020, Torshavn, Faroe Islands). Currently, he also serves on the editorial board for more than ten international journals. Figen Ozen is the Assistant Professor of the Department of Electrical and Electronics Engineering at Halic University, Istanbul, Turkey.

Funding

Not applicable.

Availability of data and materials

Not applicable.

Competing interests

Not applicable.

Author details

¹Department of Engineering, Raja University, Qazvin, Iran. ²Faculty of Science and Technology, University of the Faroe Islands, FO 100, Tórshavn, Faroe Islands. ³Electrical and Electronics Engineering Department, Haliç University, Beyoğlu İstanbul, Turkey.

Received: 24 October 2019 Accepted: 17 February 2020

Published online: 05 March 2020

References

- Naghdehforushha SA, Bahaghighat M, Salehifar MR, Kazemi H (2018) Design of planar plate monopole antenna with vertical rectangular cross-sectional plates for ultra-wideband communications. *Facta Univ Ser Electron Energetics* 31(4):641–650
- Purwita AA, Soltani MD, Safari M, Haas H (2019) Terminal orientation in ofdm-based lifi systems. *IEEE Trans Wirel Commun*. <https://doi.org/10.1109/twc.2019.2920132>
- Bahaghighat M, Naghdehforushha A, Salehifar MR, Mirfattahi M (2018) Designing straight coaxial connectors for feeder and jumpers in cellular mobile base stations. *Acta Technica Napoc Electron Telecommunicatii* 59(1)
- Hasani S, Bahaghighat M, Mirfatahia M (2019) The mediating effect of the brand on the relationship between social network marketing and consumer behavior. *Acta Technica Napoc* 60(2):1–6
- Wang Y, Ye Z, Wan P, Zhao J (2019) A survey of dynamic spectrum allocation based on reinforcement learning algorithms in cognitive radio networks. *Artif Intell Rev* 51(3):493–506
- Bahaghighat M, Motamedi SA (2017) Psnr enhancement in image streaming over cognitive radio sensor networks. *ETRI J* 39(5):683–694
- Bahaghighat M, Motamedi SA, Xin Q (2019) Image transmission over cognitive radio networks for smart grid applications. *Appl Sci* 9(24):5498
- Bahaghighat M, Motamedi SA (2016) It-mac: Enhanced mac layer for image transmission over cognitive radio sensor networks. *Int J Comput Sci Inform Secur* 14(12):234
- Medhat W, Hassan A, Korashy H (2014) Sentiment analysis algorithms and applications: A survey. *Ain Shams Eng J* 5(4):1093–1113
- Vinodhini G, Chandrasekaran R (2012) Sentiment analysis and opinion mining: a survey. *Int J* 2(6):282–292
- Esmaeli Kelishomi A, Garmabaki A, Bahaghighat M, Dong J (2019) Mobile user indoor-outdoor detection through physical daily activities. *Sensors* 19(3):511
- Bahaghighat MK, Sahba F, Tehrani E (2012) Textdependent speaker recognition by combination of lbg vq and dtw for persian language. *Int J Comput Appl* 51(16):23
- Hussein DME-DM (2018) A survey on sentiment analysis challenges. *J King Saud Univ Eng Sci* 30(4):330–338
- Hashem IAT, Yaqoob I, Anuar NB, Mokhtar S, Gani A, Khan SU (2015) The rise of "big data" on cloud computing: Review and open research issues. *Inf Syst* 47:98–115
- Zhang L, Wang S, Liu B (2018) Deep learning for sentiment analysis: A survey. *Wiley Interdiscip Rev Data Min Knowl Discov* 8(4):1253
- Morente-Molinera JA, Kou G, Samuylov K, Ureña R, Herrera-Viedma E (2019) Carrying out consensual group decision making processes under social networks using sentiment analysis over comparative expressions. *Knowl Based Syst* 165:335–345

17. Alaei AR, Becken S, Stantic B (2019) Sentiment analysis in tourism: capitalizing on big data. *J Travel Res* 58(2):175–191
18. Xie X, Ge S, Hu F, Xie M, Jiang N (2019) An improved algorithm for sentiment analysis based on maximum entropy. *Soft Comput* 23(2):599–611
19. Alam S, Yao N (2018) The impact of preprocessing steps on the accuracy of machine learning algorithms in sentiment analysis. *Comput Math Organ Theory*:1–17. <https://doi.org/10.1007/s10588-018-9266-8>
20. Liu B, Blasch E, Chen Y, Shen D, Chen G (2013) Scalable sentiment classification for big data analysis using naive bayes classifier. In: 2013 IEEE International Conference on Big Data. IEEE. pp 99–104. <https://doi.org/10.1109/bigdata.2013.6691740>
21. Sankar H, Subramaniaswamy V, Vijayakumar V, Arun Kumar S, Logesh R, Umamakeswari A (2019) Intelligent sentiment analysis approach using edge computing-based deep learning technique. *Softw Pract Experience*. <https://doi.org/10.1002/spe.2687>
22. Al-Smadi M, Al-Ayyoub M, Jararweh Y, Qawasmeh O (2019) Enhancing aspect-based sentiment analysis of arabic hotels' reviews using morphological, syntactic and semantic features. *Inf Process Manag* 56(2):308–319
23. Jianqiang Z, Xiaolin G, Xuejun Z (2018) Deep convolution neural networks for twitter sentiment analysis. *IEEE Access* 6:23253–23260
24. Dragoni M, Petrucci G (2018) A fuzzy-based strategy for multi-domain sentiment analysis. *Int J Approx Reason* 93:59–73
25. Sajadi MSS, Babaie M, Bahaghighat M (2018) Design and implementation of fuzzy supervisor controller on optimized dc machine driver. In: 2018 8th Conference of AI & Robotics and 10th RoboCup Iranopen International Symposium (IRANOPEN). IEEE. pp 26–31. <https://doi.org/10.1109/rios.2018.8406627>
26. Pham D-H, Le A-C (2018) Learning multiple layers of knowledge representation for aspect based sentiment analysis. *Data Knowl Eng* 114:26–39
27. Pang B, Lee L (2004) A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In: Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics. Association for Computational Linguistics. p 271. <https://doi.org/10.3115/1218955.1218990>
28. Kennedy A, Inkpen D (2006) Sentiment classification of movie reviews using contextual valence shifters. *Comput Intell* 22(2):110–125
29. Martineau J, Finin T, Joshi A, Patel S (2009) Improving binary classification on text problems using differential word features. In: Proceedings of the 18th ACM Conference on Information and Knowledge Management. pp 2019–2024. <https://doi.org/10.1145/1645953.1646291>
30. Maas AL, Daly RE, Pham PT, Huang D, Ng AY, Potts C (2011) Learning word vectors for sentiment analysis. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-volume 1. Association for Computational Linguistics. pp 142–150. <https://dl.acm.org/doi/10.5555/2002472.2002491>. <https://dl.acm.org/doi/pdf/10.5555/2002472.2002491?download=true>
31. Tu Z, He Y, Foster J, Van Genabith J, Liu Q, Lin S (2012) Identifying high-impact sub-structures for convolution kernels in document-level sentiment classification. In: Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers-Volume 2. Association for Computational Linguistics. pp 338–343. <https://dl.acm.org/doi/10.5555/2390665.2390742>. <https://dl.acm.org/doi/pdf/10.5555/2390665.2390742?download=true>
32. Nguyen DQ, Nguyen DQ, Pham SB (2013) A two-stage classifier for sentiment analysis. In: Proceedings of the Sixth International Joint Conference on Natural Language Processing. Asian Federation of Natural Language Processing. pp 897–901. <https://www.aclweb.org/anthology/I13-1114/>
33. Bahaghighat M, Akbari L, Xin Q (2019) A machine learning-based approach for counting blister cards within drug packages. *IEEE Access* 7:83785–83796
34. Nguyen DQ, Nguyen DQ, Vu T, Pham SB (2014) Sentiment classification on polarity reviews: an empirical study using rating-based features. <https://doi.org/10.3115/v1/w14-2621>
35. Karimimehr N, Shirazi AAB, et al. (2010) Fingerprint image enhancement using gabor wavelet transform. In: 2010 18th Iranian Conference on Electrical Engineering. IEEE. pp 316–320. <https://doi.org/10.1109/iranianee.2010.5507055>
36. Bahaghighat MK, Akbari R, et al. (2010) "fingerprint image enhancement using gwt and dmf". In: 2010 2nd International Conference on Signal Processing Systems, vol. 1. IEEE. pp 1–253. <https://doi.org/10.1109/icpsps.2010.5555771>
37. Akbari R, Keshavarz M, Mohammadi J (2010) "legendre moments for face identification based on single image per person". In: 2010 2nd International Conference on Signal Processing Systems, vol. 1. IEEE. pp 1–248. <https://doi.org/10.1109/icpsps.2010.5555580>
38. Mohammadi J, Akbari R, Bahaghighat M (2010) "vehicle speed estimation based on the image motion blur using radon transform". In: 2010 2nd International Conference on Signal Processing Systems, vol. 1. IEEE. pp 1–243. <https://doi.org/10.1109/icpsps.2010.5555577>
39. Bahaghighat M, Mirfattahi M, Akbari L, Babaie M (2018) "designing quality control system based on vision inspection in pharmaceutical product lines". In: 2018 International Conference on Computing, Mathematics and Engineering Technologies (iCoMET). IEEE. pp 1–4. <https://doi.org/10.1109/icomet.2018.8346360>
40. Babaie M, Shirri ME, Bahaghighat M (2018) "a new descriptor for uav images mapping by applying discrete local radon". In: 2018 8th Conference of AI & Robotics and 10th RoboCup Iranopen International Symposium (IRANOPEN). IEEE. pp 52–56. <https://doi.org/10.1109/rios.2018.8406631>
41. Bahaghighat M, Motamedi SA (2018) "vision inspection and monitoring of wind turbine farms in emerging smart grids". *Facta Univ Ser Electron Energetic* 31(2):287–301
42. Schmidhuber J (2015) Deep learning in neural networks: An overview. *Neural Netw* 61:85–117
43. Sundermeyer M, Ney H, Schlüter R (2015) From feedforward to recurrent lstm neural networks for language modeling. *IEEE/ACM Trans Audio Speech Lang Process* 23(3):517–529
44. Kim D, Park C, Oh J, Lee S, Yu H (2016) Convolutional matrix factorization for document context-aware recommendation. In: Proceedings of the 10th ACM Conference on Recommender Systems. ACM. pp 233–240. <https://doi.org/10.1145/2959100.2959165>
45. Bayar B, Stamm MC (2016) A deep learning approach to universal image manipulation detection using a new convolutional layer. In: Proceedings of the 4th ACM Workshop on Information Hiding and Multimedia Security. ACM. pp 5–10. <https://doi.org/10.1145/2909827.2930786>
46. Wang Y, Huang M, Zhao L, et al. (2016) Attention-based lstm for aspect-level sentiment classification. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. pp 606–615. <https://doi.org/10.18653/v1/d16-1058>
47. Sak H, Senior A, Beaufays F (2014) Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition. arXiv preprint. arXiv:1402.1128
48. Chen K, Seuret M, Hennebert J, Ingold R (2017) Convolutional neural networks for page segmentation of historical document images. In: 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR). IEEE Vol. 1. pp 965–970. <https://doi.org/10.1109/icdar.2017.161>
49. O'Shea K, Nash R (2015) An introduction to convolutional neural networks. arXiv preprint. arXiv:1511.08458
50. He B, Guan Y, Dai R (2018) Convolutional gated recurrent units for medical relation classification. In: 2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM). IEEE. pp 646–650. <https://doi.org/10.1109/bibm.2018.8621228>
51. Kim I-J, Xie X (2015) Handwritten hangul recognition using deep convolutional neural networks. *Int J Doc Anal Recogn (IJ DAR)* 18(1):1–13

Publisher's Note

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