IJICTR
International Journal of Information & Communication Technology Research

Volume 11- Number 3 – Summer 2019 (57 -67)

Improving the Performance of Text Sentiment Analysis using Deep Convolutional Neural Network Integrated with Hierarchical Attention Layer

Hossein Sadr
Department of Computer Engineering
Rasht Branch, Islamic Azad University
Rasht, Iran
Sadr@qiau.ac.ir

Mir Mohsen Pedram*
Department of Electrical and Computer Engineering
Faculty of Engineering, Kharazmi University
Tehran, Iran
Pedram@khu.ac.ir

Mohammad Teshnehlab
Industrial Control Center of Excellence
Faculty of Electrical and Computer Engineering, K. N. Toosi University
Tehran, Iran
Teshnehlab@eetd.kntu.ac.ir

Received: 2 March 2019 - Accepted: 20 June 2019

Abstract—**Sentiment analysis is considered as one of the most essential tasks in the field of natural language processing and cognitive science. In order to enhance the performance of sentiment analysis techniques, it is necessary to not only classify the sentences based on their sentimental labels but also to extract the informative words that contribute to the classification decision. In this regard, deep neural networks based on the attention mechanism have achieved considerable progress in recent years. However, there is still a limited number of studies on attention mechanisms for text classification and especially sentiment analysis. To fill this lacuna, a Convolution Neural Network (CNN) integrated with attention layer is presented in this paper that is able to extract informative words and assign them higher weights based on the context. In the attention layer, the proposed model employs a context vector and tries to measure the importance of a word as the similarity between the context vector and word vector. Then, by integrating the new vectors obtained from the attention layer into sentence vectors, the new generated vectors are used for classification. In order to verify the performance of the proposed model, various experiments were conducted on the Stanford datasets. Based on the results of the experiments, the proposed model not only significantly outperforms other existing studies but also is able to consider the context to extract the informative words which can be considered as a value in analysis and application.**

Keywords-Natural language processing, Sentiment analysis, Deep Learning, Convolutional neural network, Attention mechanism

I. INTRODUCTION

During the past few years, a large number of texts containing people's opinions, sentiments, attitudes, emotions, and cognition have been rapidly produced due to the explosive growth of social media. Considering the fact that collecting and analyzing such a large amount of unstructured data is not possible, it

* Corresponding Author

has been tried to provide an efficient method to collect and process them automatically. The automatic process of text analysis and computational linguistics with the aim of extracting subjective information existing in the text is known as sentiment analysis [1, 2]. Sentiment analysis is considered as one of the most active research areas in the field of natural language processing and cognitive science which tries to classify a piece of text containing opinions based on its polarity and determine whether an expressed opinion about a specific topic, event or product is positive or negative [3, 4].

Since about a decade ago, many studies have been carried out to investigate the effects of traditional classification models, such as Support Vector Machine (SVM), Naïve Bayes, Logistic Regression, etc. in the task of sentiment analysis [5, 6]. Although machine learning models have achieved great success in this filed, they are still confronted with some limitations, notably manual feature engineering requirements [7, 8]. In other words, the classification performance of machine learning models is highly dependent on the extracted features and they play an important role in obtaining higher classification accuracy [9]. To deal with these problems, deep learning models have been extensively employed as an alternative to traditional machine learning models and have achieved impressive results [10]. Deep learning consists of artificial neural networks that are modeled on similar networks that are present in the human brains and it can be claimed that the connection of deep learning and the human brain is far from clear now. In fact, deep learning allows computational models that are composed of multiple processing layers to learn representations of data with multiple levels of abstraction [11, 12]. Therefore, these models can automatically extract features and yield higher performance and accuracy. Deep learning contains various networks such as Convolutional Neural Network (CNN) [13], Recursive and Recurrent Neural Network (RNN) [14], and Deep Belief Network (DBN) [15] that presented surprisingly effective performances in vector representation, sentence and topic modeling, machine translation, and especially sentiment analysis [2, 16].

Despite the fact that deep learning models have been quite effective in the field of sentiment analysis, they still suffer from over-abstraction problems [17, 18]. It means that these models can only clarify the polarity of the documents and are not able to provide a depth understanding of the text such as identifying the main word that contributes to the polarity classification or finding opposite word or phrases to the overall sentiment of the document like negative words in the positive document or positive words in the negative document. In fact, they cannot consider that all words in a sentence are not equally important and some words have more impact on specifying the whole meaning of a document.

Considering the fact that deep learning models, unlike human brains, cannot pay more attention to the salient part of a text which results in a reduction in their effectiveness, a new direction of deep learning models has recently emerged that tries to simulate the attention mechanism found in human brains. Attention mechanism tries to focus on the more important part of a text (as the human brain while reading) and neglects the less important parts. In other words, the attention mechanism helps deep learning models to specify salient words and phrases and capture basic insight about documents. Attention mechanism, based on its achievement, has attracted many researchers in the field of computer vision and image recognition [19] and also has made its way to natural language processing in recent years and it has been used in various applications including machine translation [20, 21], image captioning [22], document classification [23, 24] and many more.

Following a similar line of research, we decided to employ the attention mechanism in the field of sentiment analysis. Convolutional neural networks are one of those prominent models that are commonly used for the task of sentiment analysis and have achieved significant results [25]. However, besides their remarkable results, they are still confronted with some limitations. Generally, they consider that all words in a sentence have equal contributions in the sentence meaning representation and are not able to extract informative words. To fill this lacuna, we decided to integrate the convolutional neural network and attention mechanism to present a powerful model for sentiment analysis of texts. The intuition behind our model is that all words in a sentence are not equally important and our model can identify the most informative words and phrases of sentences using attention layer by considering the context when phrase-word level sentiment labels are not available. Based on the proposed model, words with higher attention weight contain more valuable information. The main contribution of this paper is summarized as follows:

Convolutional neural network is integrated with hierarchical attention layer to focus on salient words and phrases in the text for the task of sentiment analysis.

The key difference of the proposed model compared to previous studies refer to its ability in considering the context and word interactions to measure the importance of a word in a sentence. Therefore, the proposed model is able to provide insight into which words carry more valuable information and contribute to the classification decision considering the context.

An extensive set of experiments were conducted in this paper to demonstrate the performance of the proposed model and based on the empirical results the proposed model not only obtained superior efficiency compared to traditional deep neural networks but also has a better performance compared to models that utilized attention mechanism.

The rest of the paper is organized as follows. Related studies are briefly described in Section II. Details of the proposed model are completely explained in Section III. Datasets, model configuration, training, and experimental results are described in Section IV. Conclusions and directions for future research are indicated in Section V.

II. RELATED WORK

There is no doubt that in recent years deep learning has made a revolution in researches in the field of

natural language processing and yielded to amazing technological advances in various tasks, such as machine translation, language modeling, text classification, document summarization, and so on[26]. In this regard, sentiment analysis, as one of the important aspects of natural language processing, has also a dramatic improvement using various deep learning models[2, 8].

Convolutional neural network is one of the most studied deep learning models in the field of sentiment analysis. Kim et al. [13] conducted a series of experiments based on one layer convolutional neural network for this aim. They trained their models on pre-trained vectors derived from the Word2Vec embedding model. They also employed multi-channel representation and various filer sizes and achieved comparable results. Against modeling sentences at the word level, Zhang et al. [27] presented a character level CNN for text classification that showed significant enhancement in classification accuracy. Moreover, Kalchbrenner et al.[28] proposed a dynamic CNN that utilized dynamic k-max pooling. While their model was able to handle input sentences of variable lengths, it could efficiently capture short and long-term dependencies. Yin and Schutze [29] presented a multichannel variable size CNN that employed combinations of various word embedding techniques as input as well as using variable-size convolutional filters for extracting features. In spite of the fact that CNNs have achieved significant results in the field of sentiment analysis, they are still confronting with some limitations. In fact, they cannot consider long distances sequential information which can have a great effect on the performance which is enhanced by increasing the length of the sentences [30, 31].

Recurrent neural network is another deep learning model that takes sequential data into consideration. Tai et al. [32] employed Long Short Term Memory (LSTM) network integrated with some complex units for sentiment analysis. They also conducted more experiments on two layers bidirectional LSTM and achieved significant results. Following a similar line of research, Kuta et al.[33] proposed tree structure gated recurrent neural network which was inspired by tree structure LSTM and adaptation of Gated Recurrent Unit (GRU) to recursive model. Moreover, Kumar et al.[34] proposed a dynamic memory network that processed input sentences and generated relevant classification where LSTM was used for encoding and decoding.

Besides these networks, a semi-supervised model known as the Recursive neural network has been also employed for the task of sentiment analysis which uses continuous word vectors as input and hierarchical structure. In this regard, Socher et al. [14] introduced a model, known as MV-RNN, that employed both matrix and vector with the aim of representing words and phrases in the tree structure. Although this model achieved considerable results compared to previous studies, it required a large number of parameters for training that were greatly dependent on the vocabulary size. To overcome this issue, Recursive Neural Tensor Network (RNTN) was proposed by Socher et al. [35] where the tensor-based compositional matrix was used instead of matrix representation for all nodes in the tree structure.

It must be noted that although deep learning models have achieved considerable results in the field of sentiment analysis, they consider all words in the sentences equally and are not able to focus on salient parts of the text [23]. To fill this lacuna, the attention mechanism has been recently adopted in many tasks of natural language processing, especially sentiment analysis due to its strength in providing an effective interpretation of the text. However, it must be taken into consideration that despite promising results of applying attention mechanism on deep neural networks, only a few studies have been conducted in the field of sentiment analysis that employed attention mechanism on deep neural network owing to the unavailability of word-level sentiment labels [17, 18].

To this end, Yang et al. [23] modified the RNN by adding weight that played attention role for the aim of text classification. Wang et al. [36] also proposed an attention-based LSTM network that could focus on various parts of the sentences. Following a similar line of research, Du et al.[37] conducted a series of experiments to prove that CNN is a suitable model for extracting attention from the text and proposed a model based on the combination of RNN and CNN based attention networks. In fact, they combined the recurrent neural network as a text encoder with a convolutional neural network as the attention extractor to extract attention from text sequences. Moreover, Gao et al. [38] used scaled dot product attention which is a type of self-attention for the task of text classification to focus on words having more impact in classification.

Shin et al. [39] also used attention vectors for capturing global features of sentences aiming to focus of salient part of the text. Based on their proposed method, the attention weighs of sing word and multiple words are both considered. In addition, Wang et al. [40] also examined two other attention mechanisms where LSTM and attentive pooling were used to compute the attention weight. Kokkino et al. [41] also proposed a tree structure attention network for sentiment classification which was based on the development of recursive models. In the following, Zhang et al. [42] also proposed a method that used semantic embedding, sentiment embedding, and lexicon embedding for text encoding. In their method, they also used three different attention mechanism, known as attention vector, LSTM and attentive pooling that were integrated with convolutional neural network while hand-crafted features were also utilized as additional information.

Inspired by the study of Du et al.[37], we decided to propose a novel CNN integrated with attention layer for the aim of sentiment analysis in this paper. Despite previous studies, the proposed model applies attention mechanism after the convolutional layer to extract informative words existing in the sentences by assigning a higher weight to them that leads to the creation of the new representation of word vectors. The key difference of the proposed model is that it uses context to discover if a sequence of words is relevant rather than only filtering them without considering the context. In fact, the intuition behind the proposed model

is that all words in a sentence are not equally relevant and for detecting the most relevant words, it is necessary to consider the interaction of words. It is also worth mentioning that although attention mechanism is a familiar subject that has been extensively employed in recent years, it is still in its new way of improvement, and conducting more studies in this field can be really imaginable

III. METHODOLOGY

The prominent downside of classical models is that they do not consider the fact that all words in a sentence are not semantically equal in determining the sentence meaning. In other words, identifying important words in sentences is generally context-dependence and the same words can have different importance in the various text. To overcome this issue, a convolutional neural network integrated with the hierarchical attention layer which is able to extract the words that are more significant to the meaning of the sentences is presented in this paper.

The proposed attention mechanism has two distinctive characteristics. 1) It employs the hierarchical nature of text data. In fact, words are composed of letters, sentences are composed of words and paragraphs are composed of sentences, and so on. Therefore, there is a hierarchy among parts that constitute a document and the proposed model employs this hierarchical structure where the words are encoded in the first level and then by using them, output probabilities are predicted in the final layer that presents the polarity of the whole sentence. 2) It has two levels which is first applied on word level and then combined to be applicable on the sentence level. In other words, word attention aligns words and weighs them based on how important are they in forming the meaning of a sentence. Therefore, the proposed attention mechanism can help to better understand the overall semantic structure of the document and classifies it. The proposed model consists of four layers and its diagram is depicted in Fig 1.

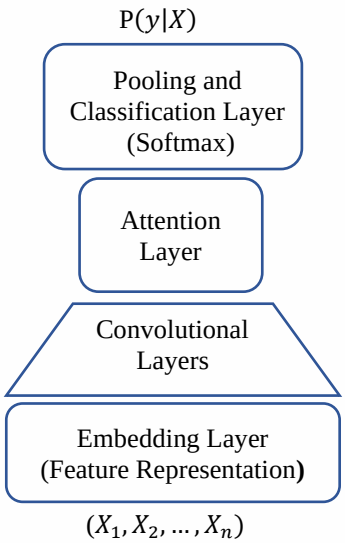

Figure 1. Diagram of the proposed method.

Firstly, by performing word embedding, word vectors of input sentences are extracted and then joined to form the initial input matrix for CNN. Secondly, convolutional operations with various filter sizes are applied to the input matrix to extract feature maps. Thirdly, feature maps extracted from similar filter sizes are merged and fed to the attention layer. In the following, by extracting the informative words via assigning a higher weight to them using attention mechanism and aggregating their representations to the previous features extracted by convolutional layer, new sentence vector are formed. Finally, new vectors are fed to the pooling layer and its outputs are then used for classification. More detailed mathematical deduction about each layer is provided as follows.

A. *Feature representation layer*

The convolutional neural network requires a sentence matrix as an input where each row represents a word vector. If the dimensionality of word vector is d and the length of a given sentence is s, the dimensionality of sentence matrix would be $s \times d$ where padding is set to zero before the first word and after the last word in the sentence. Setting the padding to zero makes the number of times that each word is included in receptive filed during the convolution the same without considering the word position in the sentence. As a result, the sentence matrix is denoted by $A \in \boldsymbol{\mathcal{R}}^{s \times d}$. In this paper, various word embedding techniques including Random vectors, Word2vec[43], Glove[44], and FastText [45] were used in our experiments to constructs the input matrix. With the random vectors, the input vectors were initialized randomly and updated along the training process while for the other three methods, word embedding vectors were separately trained using the same corpus for sentiment analysis.

B. *Convolutional layer*

To produce new features, the convolutional operation must be applied to sentence matrix. According to the fact that the sequential structure of a sentence has an important effect in determining its meaning, it is sensible to choose filter width equal to the dimensionality of word vectors (d). In this regard, only the height of filters(h), known as region size, can be varied.

Considering $A \in \boldsymbol{\mathcal{R}}^{s \times d}$ as a sentence matrix, convolution filter $H \in \boldsymbol{\mathcal{R}}^{\boldsymbol{h \times d}}$ is applied on A to produce its submatrix as new feature $A[i:j]$. As the convolution operation is applied repeatedly on the matrix A, $O \in \boldsymbol{\mathcal{R}}^{s-h+1 \times d}$ as the output sequence is generated (Eq. 1).

$$O_i = w \cdot A[i{:}\, i+h-1] \qquad (1)$$

Here $i = 1, \dots, s-h+1$ and $\cdot$ is the convolutional operator between the convolution filter and submatrix. Bias term $b \in \boldsymbol{\mathcal{R}}$ and an activation function are also added to each O_i. Finally, feature maps $C \in \boldsymbol{\mathcal{R}}^{\boldsymbol{s-h+1}}$ are generated (Eq.2).

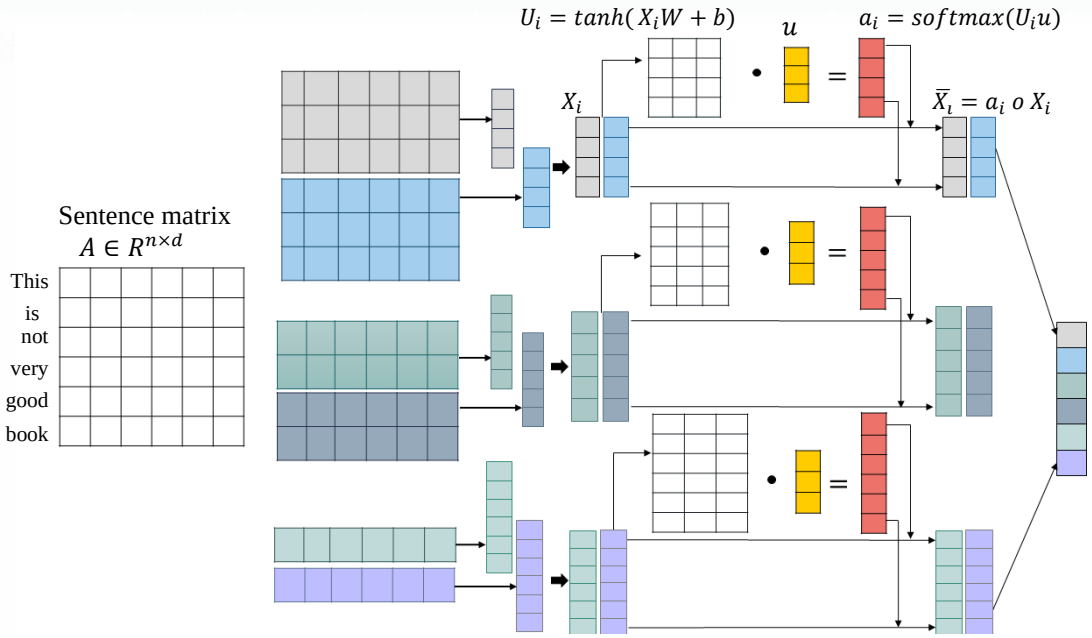

Figure 2. Structure of the convolutional neural network integrated with attention layer.

$$C_i = f(O_i + b) \quad (2)$$

C. Attention Layer

Whereas it is believed that all words in a sentence do not contribute equally to represent the meaning of a sentence, there is a need for a mechanism to emphasize such words that have more impact on the meaning of the sentences considering the context and interaction of words. For this aim, we decided to apply an attention mechanism on feature maps extracted from the previous layer. The overall structure of the convolutional neural network integrated with the hierarchical attention layer is presented in Fig 2.

To perform attention mechanism on the convolutional layer, feature maps extracted from the same filter size are aggregated and form a new matrix. Suppose that in the convolutional layer, M different region sizes are considered and for each of them m different filters are employed. Therefore, after applying $H_{ij} \in \boldsymbol{\mathcal{R}}^{\boldsymbol{h_i \times d}}$ filters on sentence matrix A where $i = 1,2,\dots,M$ and $j = 1,2,\dots,m$, $M \times m$ feature maps are obtained. By concating feature maps extracted from the same filter size, a new sentence matrix $X_i \in \boldsymbol{\mathcal{R}}^{n \times m}$ (Eq.3) is obtained. Where n is the number of words and each element of this matrix represents the feature extracted from the input using filters with the same size.

$$X_i = \begin{bmatrix} x_{1,1} & \cdots & x_{1,m} \\ \vdots & \ddots & \vdots \\ x_{n-c_i+1,1} & \cdots & x_{n-c_i+1,m} \end{bmatrix} \quad (3)$$

The objective of the attention mechanism is to assign specific weight to each row for extracting informative parts of the sentence. For this aim, firstly, new word matrix X_i is fed through a single layer perceptron using $w \in \boldsymbol{\mathcal{R}}^{\boldsymbol{m \times d}}$ and $U_i \in \mathcal{R}^{n-h_i+1 \times d}$ as a hidden representation of X_i is obtained (Eq.4).

$$U_i = \tanh(X_iW + b) \quad (4)$$

In the following, the importance of each word is measured as the similarity of U_i with a context vector $u \in \boldsymbol{\mathcal{R}}^{d \times 1}$ and to achieve the normalized importance weight $a_i \in \boldsymbol{\mathcal{R}}^{n-h_i+1 \times 1}$, *Softmax* function is used (Eq.5). Notably, the context vector u can be considered as a high-level representation to specify informative words and it is like the mechanism that is used in the memory networks [34, 46].

$$a_i = \text{softmax}(U_iu) \quad (5)$$

Notably, u is set to zero in the beginning to consider the same weight for various rows in the matrix of X_i and it is learned along the training process. After that, $\bar{X}_i$ (a new representation of X_i) is computed by multiplying each element of a_i to its corresponding row in X_i matrix ($\circ$ is element-wise product) (Eq.6).

$$\bar{X}_i = a_i \circ X_i \quad (6)$$

Generally, $\bar{X}_i$ is a new representation of X_i while the attention mechanism is applied to it in order to specify the informative words. The whole process of attention layer is schematically presented in Fig 3. As it can be clearly seen, after merging feature maps extracted from the same filter sizes, X_i matrix is created. Then, by applying a single layer perceptron, a new representation of X_i known as U_i is created. In the following, the normalized importance weight a_i, indicating the importance of each word, is computed as the similarity between U_i and the context vector u which is a hyper-parameter and is tuned during the training process. Finally, $\bar{X}_i$ is a new representation of X_i which is achieved by multiplying each element of a_i to its corresponding row in X_i. Generally, applying the attention mechanism leads to extracting informative

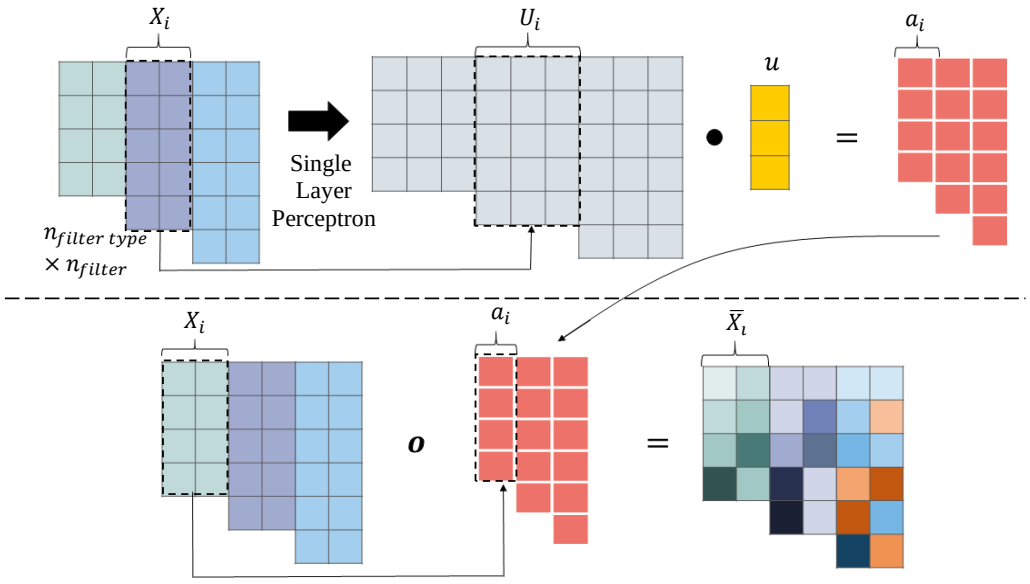

Figure 3. Schematic representation of the whole process of attention layer.

words and assigning more weight to them which is schematically illustrated in Fig 3 by darker colors.

D. Pooling and fully connected layer

While various feature maps, according to different filter sizes, are generated, a pooling function is required to induce fixed size vectors. Various strategies, such as average pooling, minimum pooling, and maximum pooling can be used for this aim and the idea behind them is to capture the most important feature from each feature map as well as reducing dimensionality. After applying the pooling function, generated features are then concated into a feature vector o_i. The feature vector is passed to a fully connected *Softmax* layer to specify the final classification. In other words, *Softmax* determines the probability distribution over all sentiment categories and is calculated as follows (Eq.7).

$$P_i = \frac{\exp(o_i)}{\sum_{j=1}^{C} \exp(o_j)} \tag{7}$$

To clarify the difference between the real sentiment distribution $\widehat{P}_i(C)$ and the distribution achieved from the model $P_i(C)$, cross-entropy as the loss function is employed (Eq.8). Where T is the training set and V is the sentiment categories.

$$Loss = -\sum_{s \in T} \sum_{i=1}^{V} \widehat{P}_i(C) \log(P_i(C)) \tag{8}$$

IV. EXPERIMNTS

A. Dataset

In order to have a comprehensive investigation of the effectiveness of the proposed method, Stanford datasets were used in the experiments of this paper. These datasets have been presented by the natural language processing laboratory of Stanford University and are known as the most important benchmark in this filed [35]. Considering the fact that most of the previous studies have conducted their experiments on these datasets, they have been also used in our experiment to provide us with this opportunity to be able to compare our proposed model with a wide range of existing models. Employed datasets are briefly explained as follows and their summary statistics after tokenization are presented in Table 1.

- SST1: It is the extended version of MR [47] dataset that has train/dev/test splits and fine-grained labels. Reviews in this dataset are categorized into five categories as negative, somewhat negative, neutral, somewhat positive, and positive [35].
- SST2: It is the modified version of SST1 that only includes binary labels (negative and positive) and neutral labels are eliminated [35].

It must be noted that SST1 and STT2 datasets are for sentence-level classification and Standard train/test sets of SST1, SST2 were used for conducting the experiments.

TABLE I. STATISTICS OF SST1 AND SST2 DATASETS.

Dataset	Class	Vocabulary Size	Average length	Text Size	Test size
SST1	5	18K	18	11855	2210
SST2	2	15K	19	9613	1821

B. Baselines

To illustrate the performance of the proposed method, it is compared with some baselines and state of the art models that are classified into 5 categories that are explained in details in the following.

Category A contains some traditional machine learning algorithms that have been used for the aim of sentiment analysis.

Category B contains models in the family of the convolutional neural network.

Category C includes models in the group of recurrent neural networks that consider the sequential order.

Category D contains recursive neural networks that utilize tree structure for classification.
Category E contains neural networks that utilize various kinds of attention mechanisms.
Category F contains the proposed model and its all variations.

- *CNN-Attention-Rand:* Random initialized vectors are used as the input of the proposed model.
- *CNN-Attention-Static:* Pre-trained word vectors achieved from Word2Vec are employed as the input of the proposed model. Notably, weights are not updated during the training process.
- *CNN-Attention-Non-Static:* Pre-trained word vectors achieved from Word2Vec are employed as the input and are also updated during the training process.
- *CNN-Attention-2channels*: Pre-trained word vectors achieved from Word2Vec and Random initialized vectors are combined and used as the input of the proposed model.
- *CNN-Attention-4channels*: Pre-trained word vectors achieved from Word2Vec, Glove, and FastText are employed as the input along with the random initialized vectors that are updated during the training process.

C. Model configuration and training

Preprocessing is considered as one of the most important components of training. In this regard, firstly the documents were split into sentences and Stanford core NLP was applied for the aim of tokenization. Extracted tokens were then used to obtain word embedding vectors using unsupervised Word2Vec [43], FastText [45] and Glove [44] models. The dimension of word vectors was considered as 200 and window size was set to 3. Word vectors were updated based on a learning rate of 0.1. It must be noted that the skip-gram structure was utilized in Word2Vec [43] and FastText [45] models while Glove [44] used the unigram structure.

ADADELTA update rule was employed for stochastic gradient descent with a learning rate of 0.01 while mini-batch was 25. In a convolutional neural network, filter size and number of filters were considered as hyperparameters. Their values for training the proposed model are presented in Table 2.

TABLE II. CONFIGURATION OF THE PROPOSED MODEL HYPER PARAMETERS.

Hyperparameters	Value
Filter region size	3,4,5
Number of filters	128
Dropout rate	0.5
Batch size	25
Activation Function	ReLU

As it is illustrated, it is found that filter size (3,4,5) and 128 filters yielded better results. Furthermore, the convolutional layer was regularized with a dropout rate of 0.5. ReLU (Rectified Linear Unit), known as a commonly used activation function in CNNs, was also used in the experiments of this paper. 60 epochs were also used for training the model.

D. Results

1) Classification performance

The results of the experiments over the introduced datasets are presented in Table 3 which contains 2 sections. Categories A to E contain some of the best state of the art models in the field of sentiment analysis and category F contains variations of the proposed model. It must be mentioned that the accuracies of the models in categories A to E are taken from their original papers.

As can be clearly seen, neural network-based models (Category B, C, D, and E) have better performance on average compared to traditional models that use machine learning techniques (Category A) which can obviously demonstrate the strength of deep learning models in this filed. Furthermore, among all neural network-based models (except Category F), DMN [34] and MVCNN [29] have the highest performances on SST1 and SST2 respectively.

Exploring the performance of the proposed model, it can be concluded that not only it has higher classification performance compared to models that used deep neural networks (categories B, C, and D) but also in presented superior efficiency in comparison to the existing models that adopted attention mechanism (category E). Therefore, it can be stated that applying attention mechanism on CNN has significantly enhanced the classification performance that can be due to the ability of the proposed model in considering the context along with assigning attention weight to words.

It is obvious that CNN-Attention-Rand has the lowest classification accuracy among all variations of the proposed model on both datasets while it used random initialized vectors as the input. Therefore, better performance of other variations can be also attributed to the employment of pre-trained vectors that can solve the semantic sparsity problem to some degree.

Additionally, it is obvious that vector representation has a great effect on the performance of the proposed model. Moreover, considering the fact that CNN-Attention-Static has the lowest classification accuracy besides CNN-Attention-Rand, it can be stated that updating word vectors during the training process can yield to obtain higher performance without considering if the word vectors were previously trained or not. Finally, CNN-Attention- 4channel has the highest classification accuracy as 92.34 % on SST2 dataset and CNN-Attention-2channel has the highest classification accuracy as 53.54 % on SST1 dataset. Overall, the higher classification performance of all variations of the proposed model can clearly demonstrate its superiority compared to other existing models for the task of sentiment analysis.

2) Context-dependent attention weight analysis

To have a broader view of the performance of the proposed model, more analysis has been carried out. In fact, the aim of applying the attention mechanism on CNN is to focus on more relevant words. Without the

TABLE III. RESULTS OF EXPERIMENTS ON SST1 AND SST2 DATASET. (CATEGORIES A TO E CONTAIN SOME OF THE BEST STATES OF THE ART MODELS AND CATEGORY F CONTAINS DIFFERENT VARIATIONS OF THE PROPOSED MODEL).

Category	Model	Datasets	
		SST1	SST2
A	NB[35]	41	81.8
	BiNB[35]	41.9	83.1
	SVM [31]	40.7	79.4
	WordVec-AVE [31]	32.7	80.1
B	CNN-1 layer [28]	37.4	77.1
	CNN-non static[13]	48	87.2
	CNN-multichannel[13]	47.4	88.1
	DCNN [28]	48.5	86.8
	MVCNN[29]	49.6	89.4*
C	LSTM[32]	46.2	85.2
	Bi-LSTM[32]	49.1	87.5
	Tree-LSTM[32]	51.0	88.0
	Tree-GRU[41]	50.5	88.6
	DMN [34]	52.1*	88.6
D	RecRNN[35]	43.2	82.4
	RNTN[35]	45.7	85.4
	MVRNN[14]	44.	82.9
E	Tree-GRU+ attention [41]	51.0	89.0
	Tree-BiGRU+attention[41]	52.4	89.5
	LSTM+RNN attention[36]	48.0	86.1
	CRAN-rand[37]	50.0	87.7
	CRAN-pretrain[37]	48.1	86.9
	CNN+LSTM attention[42]	49.22	88.77
	CNN+ Attentive Pooling [42]	49.58	88.81
	CNN+ Attention vector [42]	49.69	88.83
F	CNN-Attention-Rand	49.76	88.61
	CNN-Attention-Static	50.06	89.95
	CNN-Attention-Non-Static	51.61	90.64
	CNN-Attention- 2channel	**53.54**	91.92
	CNN-Attention- 4channel	52.72	**92.34**

attention mechanism, CNN might also work well and assign a high and low weight to important and not important words respectively without considering the context. While the importance of a particular word is highly dependent on the context, the goal of the proposed model is to capture context-dependence importance.

As an example, consider the word *good* that may be used in a review with the lowest rating because of negation or due to the fact that the user was happy with only one aspect of a product. To clarify the performance of the proposed model in recognition of the importance of the word based on the context, the distribution of the attention weight of words *good* and *bad* from the test split of SST1 dataset is presented in Fig 4 (a, b). According to the distribution, the assigned attention weight is on a scale of 0 to 1. It is also obvious that the words *good* and *bad* do not have a uniform distribution that can prove the potential of the proposed method in capturing diverse context and assigning context-dependent weight.

In order to have a more comprehensive analysis, the distributions of words *good* and *bad* are plot according to the rating of reviews in Fig 5 (a)-(e) corresponding to the ratings 1 to 5 respectively. Notably, the first row is related to word *good* and the second one is related to the word *bad*.

As can be clearly seen, in reviews with rating 1, the words *good* and *bad* have the lowest and highest weight

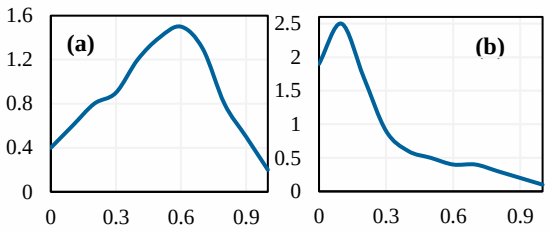

Figure 4. Aggregate distribution of attention weight of words good (a) and bad (b) on test split of SST1 dataset.

- X-axis = attention score in the scale of 0 to 1.
- Y-axis = attention weight of words good (a) and bad (b) from the test split of SST1.

respectively. In the following, as the rating is enhanced, the weight distribution for the word good is increased while it is decreased for the word bad. It indicates that positive words such as good have a more important role in higher rating reviews while negative words such as bad have more effect in lower rating reviews.

In other words, it can be stated that for the word *good* as the rating goes higher, the distribution also shifts higher. In contrast, the word *bad* has a higher distribution in poor ratings while it decreases for good ones. This indicates that the proposed model is able to capture the importance of words considering the context.

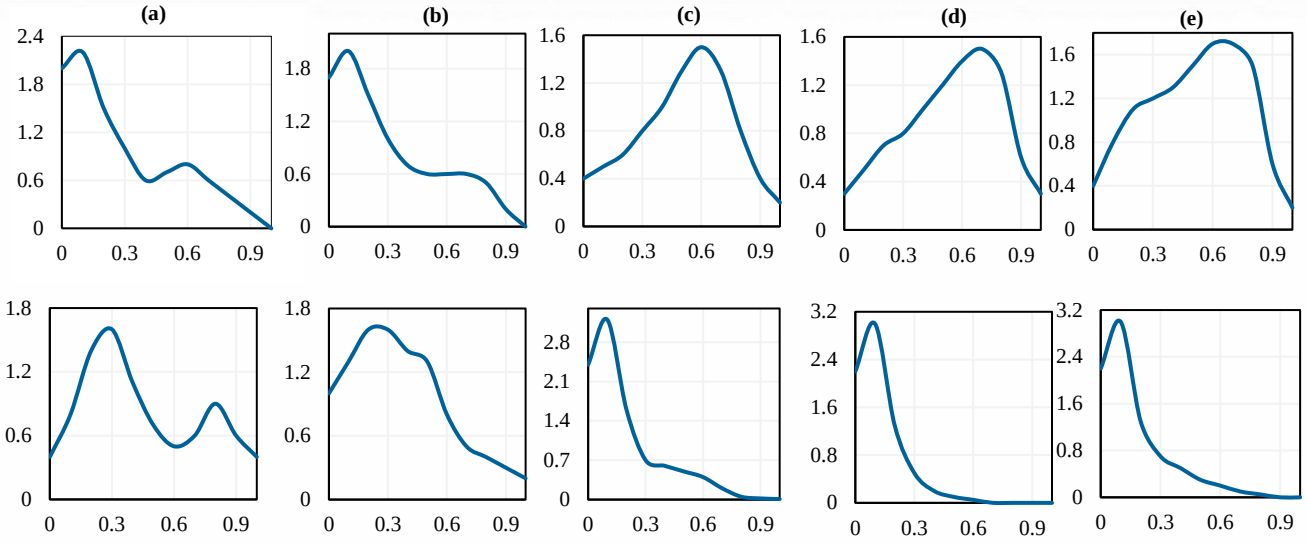

Figure 5. Stratified distribution of attention weight of words good (first row) and bad (second row) according to the ratings (1 to 5) of reviews which are presented in subfigures (a)-(e) respectively.

- X-axis = attention score in the scale of 0 to 1.
- Y-axis = attention weight of words good (first row) and bad (second row) from the test split of SST1 according to the ratings.

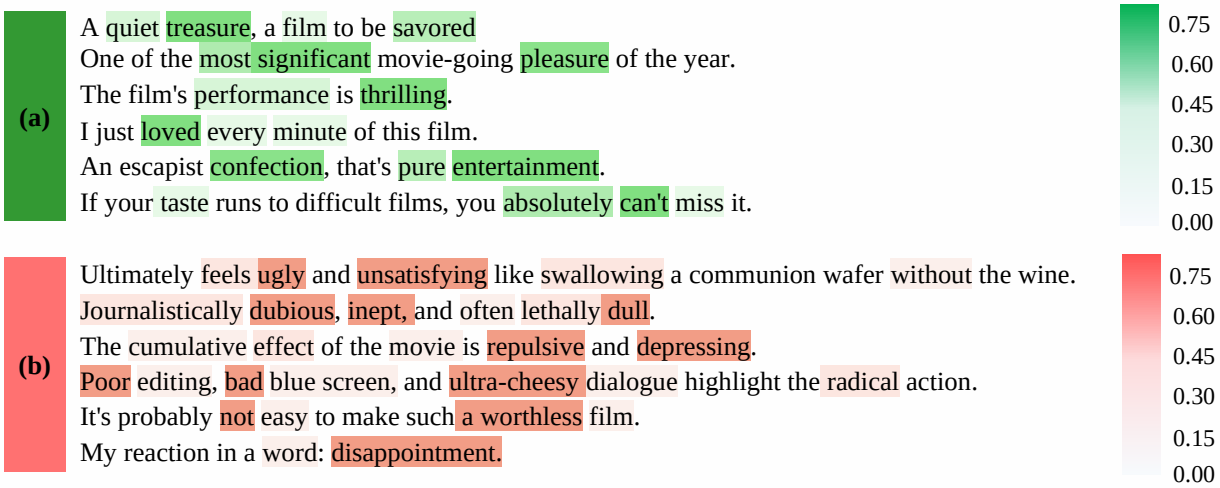

Figure 6. Examples of visualizing attention layer for sentences in SST2 dataset. Subfigure (a) represents sentences that are classified as positive and subfigure (b) represents sentences that are classified as negative.

1) Visualization of attention layer

For further analysis and to illustrate the potential of the proposed model in focusing on informative words in the sentences, the influence of the attention layer is schematically visualized in Fig 6 for several sentences of SST2 dataset where each line corresponds to a sentence and each of the subfigures (a) and (b) represents sentences that are categorized as positive and negative respectively. The words highlighted in green and red are also representing positive and negative informative words respectively that are highly scored using the proposed model and particularly context vector. It must also be taken into consideration the visualization is performed based on the results achieved from CNN-Attention-4channel due to its highest classification accuracy among all variations of the proposed model.

It must be mentioned that for visualization, highly positive and negative sentences were selected and word weights were computed using context vector in attention layer. According to the used hierarchical structure, each word weigh is normalized using the sentence weight to ensure than only informative words in sentences according to the context are emphasized. To visualize, $\sqrt{p_s}\, p_w$ is displayed where $\sqrt{p_s}$ shows the important words in unimportant sentences to make sure that they are not completely invisible.

As it can be clearly seen, the proposed model is able to successfully capture semantically positive and negative phrases and assign them more weight (highlighted with darker colors). Consequently, CNN-Attention- 4channel can not only efficiently select the words carrying strong sentiments such as *significant*, *thrilling* and *entertainment* as positive words and *ugly*, *unsatisfying* and *dull* as negative words but also is able

to deal with complex cross sentences context like *"It's probably not easy to make such worthless film"*. In this sentence, if a model only considers the words, it may categorize the sentence as a positive review as well as assigning higher weight to the word *easy*. However, the proposed model considers the context of the sentence and figures it out as a negative review by assigning a low weight to positive word *easy*.

V. CONCLUSION

In this paper, a new convolution neural network integrated with the attention mechanism for the aim of sentiment analysis is proposed. The key difference of the proposed model compared to previous studies refers to its ability in considering the context to extract the informative words. In this regard, feature maps with the same size that were extracted from the convolutional layer are merged and fed to a one layer perceptron to provide a hidden representation. Then, the attention mechanism is performed by measuring the importance as the similarity been the context vector and word vector. In the following, the sentence vectors are formed by aggregating informative word vectors obtained from attention layer into extracted feature maps that are then used for classification.

Based on the results of experiments, employing attention mechanism over convolutional neural network serves two benefits. In fact, not only the proposed model significantly outperforms all previous studies, particularly models that used attention mechanism, but also it is able to highlight informative words that effectively contribute to predicting the overall classification decision. Moreover, the influence of various word embedding techniques has been also explored in this paper and according to the empirical results, integrating various word embedding techniques for providing input matrix and updating them along the training process has also led to classification accuracy enhancement.

Following a similar line of study, employing the proposed model in tasks, such as text generation, machine translation, and sequence-to-sequence learning as well as performing it on other textual datasets can be considered as a significant future research topic. Considering the effect of word semantic similarity and relatedness for measuring the importance of the word is also worth exploring.

REFERENCES

[1] Z. Li, Y. Fan, B. Jiang, T. Lei, and W. Liu, "A survey on sentiment analysis and opinion mining for social multimedia," *Multimedia Tools and Applications*, vol. 78, no. 6, pp. 6939-6967, 2019.

[2] H. Sadr, M. M. Pedram, and M. Teshnehlab, "A Robust Sentiment Analysis Method Based on Sequential Combination of Convolutional and Recursive Neural Networks," *Neural Processing Letters*, pp. 1-17, 2019.

[3] S. M. H. Chowdhury, S. Abujar, M. Saifuzzaman, P. Ghosh, and S. A. Hossain, "Sentiment Prediction Based on Lexical Analysis Using Deep Learning," in *Emerging Technologies in Data Mining and Information Security*: Springer, 2019, pp. 441-449.

[4] A. R. Pathak, B. Agarwal, M. Pandey, and S. Rautaray, "Application of Deep Learning Approaches for Sentiment Analysis," in *Deep Learning-Based Approaches for Sentiment Analysis*: Springer, 2020, pp. 1-31.

[5] N. C. Dang, M. N. Moreno-García, and F. De la Prieta, "Sentiment Analysis Based on Deep Learning: A Comparative Study," *Electronics*, vol. 9, no. 3, p. 483, 2020.

[6] H. Sadr, M. Nazari, M. M. Pedram, and M. Teshnehlab, "Exploring the Efficiency of Topic-Based Models in Computing Semantic Relatedness of Geographic Terms," *International Journal of Web Research*, vol. 2, no. 2, pp. 23-35, 2019.

[7] H. Kaur, V. Mangat, and Nidhi, "A Survey of Sentiment Analysis techniques," *International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC)*, p. 5, 2017 2017.

[8] Q. T. Ain *et al.*, "Sentiment Analysis Using Deep Learning Techniques: A Review," *(IJACSA) International Journal of Advanced Computer Science and Applications*, p. 10, 2017.

[9] P. Angelov and A. Sperduti, "Challenges in deep learning," in *Proc. European Symp. on Artificial NNs*, 2016, pp. 485-495.

[10] O. Habimana, Y. Li, R. Li, X. Gu, and G. Yu, "Sentiment analysis using deep learning approaches: an overview," *Science China Information Sciences*, vol. 63, no. 1, pp. 1-36, 2020.

[11] X. Xie, S. Ge, F. Hu, M. Xie, and N. Jiang, "An improved algorithm for sentiment analysis based on maximum entropy," *Soft Computing*, vol. 23, no. 2, pp. 599-611, 2019.

[12] Q. Zhang, L. T. Yang, Z. Chen, and P. Li, "A survey on deep learning for big data," *Information Fusion*, vol. 42, pp. 146-157, 2018.

[13] Y. Kim, "Convolutional neural networks for sentence classification," *arXiv preprint arXiv:1408.5882*, 2014.

[14] R. Socher, B. Huval, C. D. Manning, and A. Y. Ng, "Semantic Compositionality through Recursive Matrix-Vector Spaces," *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, Association for Computational Linguistics.*, 2012.

[15] P. Ruangkanokmas, T. Achalakul, and K. Akkarajitsakul, "Deep Belief Networks with Feature Selection for Sentiment Classification," *Uksim.Info,pp. 16*, 2016.

[16] H. Sadr, M. M. Pedram, and M. Teshnehlab, "Multi-View Deep Network: A Deep Model Based on Learning Features From Heterogeneous Neural Networks for Sentiment Analysis," *IEEE Access*, vol. 8, pp. 86984-86997, 2020.

[17] G. Lee, J. Jeong, S. Seo, C. Kim, and P. Kang, "Sentiment Classification with Word Attention based on Weakly Supervised Learning with a Convolutional Neural Network," *arXiv preprint arXiv:1709.09885*, 2017.

[18] T. Nasukawa and J. Yi, "Sentiment analysis: Capturing favorability using natural language processing," in *Proceedings of the 2nd international conference on Knowledge capture*, 2003, pp. 70-77.

[19] J. Ba, V. Mnih, and K. Kavukcuoglu, "Multiple object recognition with visual attention," *arXiv preprint arXiv:1412.7755*, 2014.

[20] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[21] H. Sadr and M. Nazari Solimandarabi, "Presentation of an efficient automatic short answer grading model based on combination of pseudo relevance feedback and semantic relatedness measures," *Journal of Advances in Computer Research*, vol. 10, no. 2, pp. 1-10, 2019.

[22] A. Show, "Tell: Neural image caption generation with visual attention," *Kelvin Xu et. al.. arXiv Pre-Print*, vol. 83, p. 89, 2015.

[23] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, "Hierarchical attention networks for document classification," in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2016, pp. 1480-1489.

[24] H. Sadr, M. N. Soleimandarabi, M. Pedram, and M. Teshnelab, "Unified Topic-Based Semantic Models: A Study in Computing the Semantic Relatedness of Geographic Terms," in *2019 5th International Conference on Web Research (ICWR)*, 2019: IEEE, pp. 134-140.

[25] T. Sharma, A. Bajaj, and O. P. Sangwan, "Deep Learning Approaches for Textual Sentiment Analysis," in *Handbook of Research on Emerging Trends and Applications of Machine Learning*: IGI Global, 2020, pp. 171-182.

[26] T. Young, D. Hazarika, S. Poria, and E. Cambria, "Recent trends in deep learning based natural language processing," *arXiv preprint arXiv:1708.02709*, 2017.

[27] X. Zhang, J. Zhao, and Y. LeCun, "Character-level convolutional networks for text classification," in *Advances in neural information processing systems*, 2015, pp. 649-657.

[28] N. Kalchbrenner, E. Grefenstette, and P. Blunsom, "A convolutional neural network for modelling sentences," *arXiv preprint arXiv:1404.2188*, 2014.

[29] W. Yin and H. Schütze, "Multichannel variable-size convolution for sentence classification," *arXiv preprint arXiv:1603.04513*, 2016.

[30] X. Wang, W. Jiang, and Z. Luo, "Combination of Convolutional and Recurrent Neural Network for Sentiment Analysis of Short Texts," 2016.

[31] C. DU and L. HUANG, "Sentiment Classification Via Recurrent Convolutional Neural Networks," *DEStech Transactions on Computer Science and Engineering*, no. cii, 2017.

[32] K. S. Tai, R. Socher, and C. D. Manning, "Improved semantic representations from tree-structured long short-term memory networks," *arXiv preprint arXiv:1503.00075*, 2015.

[33] M. Kuta, M. Morawiec, and J. Kitowski, "Sentiment Analysis with Tree-Structured Gated Recurrent Units," *Springer International Publishing AG 2017*

[34] A. Kumar *et al.*, "Ask me anything: Dynamic memory networks for natural language processing," in *International conference on machine learning*, 2016, pp. 1378-1387.

[35] R. Socher *et al.*, "Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank," 2013.

[36] Y. Wang, M. Huang, and L. Zhao, "Attention-based LSTM for aspect-level sentiment classification," in *Proceedings of the 2016 conference on empirical methods in natural language processing*, 2016, pp. 606-615.

[37] J. Du, L. Gui, R. Xu, and Y. He, "A Convolutional Attention Model for Text Classification," in *National CCF Conference on Natural Language Processing and Chinese Computing*, 2017: Springer, pp. 183-195.

[38] S. Gao, A. Ramanathan, and G. Tourassi, "Hierarchical convolutional attention networks for text classification," in *Proceedings of The Third Workshop on Representation Learning for NLP*, 2018, pp. 11-23.

[39] B. Shin, T. Lee, and J. D. Choi, "Lexicon integrated cnn models with attention for sentiment analysis," *arXiv preprint arXiv:1610.06272*, 2016.

[40] L. Wang, Z. Cao, G. De Melo, and Z. Liu, "Relation classification via multi-level attention cnns," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2016, pp. 1298-1307.

[41] F. Kokkinos and A. Potamianos, "Structural attention neural networks for improved sentiment analysis," *arXiv preprint arXiv:1701.01811*, 2017.

[42] Z. Zhang, Y. Zou, and C. Gan, "Textual sentiment analysis via three different attention convolutional neural networks and cross-modality consistent regression," *Neurocomputing*, vol. 275, pp. 1407-1415, 2018.

[43] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[44] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532-1543.

[45] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *arXiv preprint arXiv:1607.04606*, 2016.

[46] S. Sukhbaatar, J. Weston, and R. Fergus, "End-to-end memory networks," in *Advances in neural information processing systems*, 2015, pp. 2440-2448.

[47] B. Pang and L. Lee, "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*, 2005: Association for Computational Linguistics, pp. 115-124.

Hossein Sadr received his Ph.D. degree in Computer Software Engineering from Islamic Azad University, Iran, in 2020, and his M.Sc. degree in the same filed from Islamic Azad University, Science and Research Branch, Iran, in 2013. He is also a member of Intelligent Systems Scientific Society of Iran (ISSSI). He is currently a lecturer in the Department of Computer Engineering at Islamic Azad University and is actively involved in the organization of a number of flagship conferences and workshops as well as cooperating as a reviewer with reputable journals, such as IEEE Access and Neural Processing Letters. His main areas of research are Natural Language Processing, Information Retrieval, Machine Learning, Deep Neural Networks, and Cognitive Science.
Email: Sadr@qiau.ac.ir

Mir Mohsen Pedram received his Ph.D. degree in Electrical Engineering from Tarbiat Modarres University, Tehran, Iran, 2003, M.Sc. degree in Electrical Engineering from Tarbiat Modarres University, Tehran, Iran, 1994 and his B.Sc. degree in Electrical Engineering from Isfahan University of Technology, Isfahan, Iran, 1990. He is currently an Associate Professor in the Department of Electrical and Computer Engineering at Kharazmi University. His main areas of research are Intelligent Systems, Machine Learning, Data Mining, and Cognitive Science.
Email: Pedram@khu.ac.ir

Mohammad Teshnehlab received the B.Sc. degree from Stony Brook University, USA, in 1980, the M.Sc. degree from Oita University, Japan, in 1990, and the Ph.D. degree from Saga University, Japan, in 1994. He is a faculty member of Electrical Eng. Department of K. N. Toosi University of Technology. Professor Teshnehlab is a member of the Industrial Control Center of Excellence and founder of Intelligent Systems Laboratory (ISLab). He is also a co-founder and member of Intelligent Systems Scientific Society of Iran (ISSSI) and a member of the editorial board of the Iranian Journal of Fuzzy Systems (IJFS), International Journal of Information & Communication Technology Research (IJICTR), and Scientific Journal of Computational Intelligence in Electrical Engineering. His research areas are Artificial Rough and Deep Neural Networks, Fuzzy Systems and Neural Nets, Optimization, and Expert Systems.
Email: Teshnehlab@eetd.kntu.ac.ir