The Journal of Supercomputing (2022) 78:10149–10175
https://doi.org/10.1007/s11227-021-04208-2

ACNN-TL: attention-based convolutional neural network coupling with transfer learning and contextualized word representation for enhancing the performance of sentiment classification

Hossein Sadr[1] · **Mojdeh Nazari Soleimandarabi**[2]

Accepted: 12 November 2021 / Published online: 21 January 2022

Abstract

Due to the rapid growth of textual information on the web, analyzing users' opinions about particular products, events or services is now considered a crucial and challenging task that has changed sentiment analysis from an academic endeavor to an essential analytic tool in cognitive science and natural language understanding. Despite the remarkable success of deep learning models for textual sentiment classification, they are still confronted with some limitations. Convolutional neural network is one of the deep learning models that has been excelled at sentiment classification but tends to need a large amount of training data while it considers that all words in a sentence have equal contribution in the polarity of a sentence and its performance is highly dependent on its accompanying hyper-parameters. To overcome these issues, an Attention-Based Convolutional Neural Network with Transfer Learning (ACNN-TL) is proposed in this paper that not only tries to take advantage of both attention mechanism and transfer learning to boost the performance of sentiment classification but also language models, namely Word2Vec and BERT, are used as its the backbone to better express sentence semantics as word vector. We conducted our experiment on widely-studied sentiment classification datasets and according to the empirical results, not only the proposed ACNN-TL achieved comparable or even better classification results but also employing contextual representation and transfer learning yielded remarkable improvement in the classification accuracy.

Keywords Natural language processing · Sentiment analysis · Deep learning · Convolutional neural network · Attention mechanism · Transfer learning · Word representation

✉ Hossein Sadr
Sadr@qiau.ac.ir

Extended author information available on the last page of the article

1 Introduction

Considering the expeditious extension of the internet and information technology, various data can be found and shared easily on the web. As a result, we are faced with a large amount of textual data in the form of text that is produced by users when giving some comments or indicating their sentiment on entities. Owing to the fact that it is not possible to easily analyze these large amounts of unstructured data, it has been tried to present effective approaches to automatically collect and process them. The automatic process of computational linguistics and text analysis aiming to extract emotional information from the text is called sentiment analysis which is nowadays considered as one of the prominent aspects of text analytics and cognitive science [1].

From the point of machine learning view, sentiment analysis is known as a classification problem that aims to categorize textual data into negative or positive classes. Support Vector Machine (SVM), Convolutional Neural Network (CNN), Recursive and Recurrent Neural Network (RNN), Long Short Term Memory (LSTM) can be mentioned as the representatives of supervised learning methods that have been used in this filed [2]. It must be mentioned that despite deep learning models have obtained remarkable development in this regard [3, 4], their efficiency is not still satisfactory and most of the presented approaches are not able to make use of the potential of deep learning properly [5, 6].

CNN is known as one of the significant deep neural networks that has been broadly leveraged for the task of sentiment analysis and achieved amazing results. Despite the effectiveness of convolutional neural networks, they are still confronted with some problems [7–9]. First, CNNs can only define the polarity of documents and cannot present a comprehensive undressing of the text like recognizing the salient words that may have a great effect in final polarity classification [5, 10, 11]. It means that, unlike human brains, convolutional neural networks cannot add emphasis on the informative part of a text which yields a decrease in their performance [6, 12]. Second, CNNs need a great number of training data to accurately train the model while they have various parameters that must be preciously tuned [13].

On the other hand, previous studies have also shown that using pre-trained models for word representation can be beneficial for various natural language processing tasks, especially sentiment analysis [14, 14]. One of the substantial pre-trained models is word embedding like Word2Vec [15], Glove [16], and ELMo [17] that are often leveraged to generate word vectors that are employed as additional features besides the main task. Another kind of pre-trained models is sentence-level like UML-FiT [18], OpenAI GPT [19], and BERT [20]. Noteworthy, although utilizing pre-trained models for word representation has yielded amazing results, their potential and influence have been yet to be fully explored. Moreover, despite word embedding, the effect of sentence-level pre-trained models has been rarely explored and there is still little research aiming to use them to improving the performance on sentiment analysis [14].

Accordingly, an Attention-Based Convolutional Neural Network with Transfer Learning (ACNN-TL) is presented. The proposed model aims to take advantage

of both attention mechanism and transfer learning besides exploring the importance of different word representations (Word2Vec and BERT) for increasing the effectiveness of sentiment analysis. Notably, attention mechanism is used to simulate the human brain while reading and emphasize the more important part of a text as well as neglecting the less important parts. While every word in a sentence does not have the same contribution to the sentence polarity, the introduced attention mechanism can help the convolutional neural network to specify the most important words and phrases of sentences by taking the context into when phrase-word level sentiment labels are not accessible [12, 21–23]. Transfer learning is also utilized in our proposed model to overcome the large training data requirement as well as enhancing the final classification performance. In this regard, the model is first trained on a large size dataset (source domain) and it is then transferred to a smaller dataset (target domain) [13].

Moreover, to better express sentence semantics, we also utilized Word2Vec and BERT structures for word representation on the backbone of our proposed model and investigated their influences on the target task. It must be mentioned that although the influence of using Word2Vec for vector representation has been extensively explored for various natural language tasks [2, 24], there is little research to employ BERT for enhancing the accuracy of text classification, especially sentiment analysis, and its potential has yet to be fully investigated. BERT is employed in our proposed model because it is able to understand the meaning of each word according to context both to the right and to the left of the word. Moreover, considering that BERT aims to forecast missing words in the text as well as analyzing every sentence with no specific direction, it has superior performance at understanding the meaning of homonyms compared to existing NLP methods, like embedding methodologies. To this end, we decided to use both of them individually as the first layer of our proposed model so as to make a comparison among them and highlight BERT potentials.

Furthermore, while a drawback of CNNs is that they need expert practitioners to define the model hyper-parameters, a set of experiments was conducted to investigate the proposed model sensitivity and obtain the desirable hyper-parameters values to enhance the classification efficiency. Briefly, the main contributions of this paper are mentioned in the following:

- To add more emphasis on the principal words and phrases of the text, we integrated CNN with a hierarchical attention mechanism to mimic the human brain for sentiment analysis. The proposed combinational model employs the attention mechanism before the pooling layer to provide an insight into which words have more remarkable information influencing the sentence polarity considering the context.
- To clearly demonstrate the influence of word representation models and enhance the performance of the target task, we utilized Word2Vec and BERT techniques individually as the backbone of our proposed model.
- To overcome the problem of large training data requirements, the effect of transfer learning on convolutional neural networks is explored and the effi-

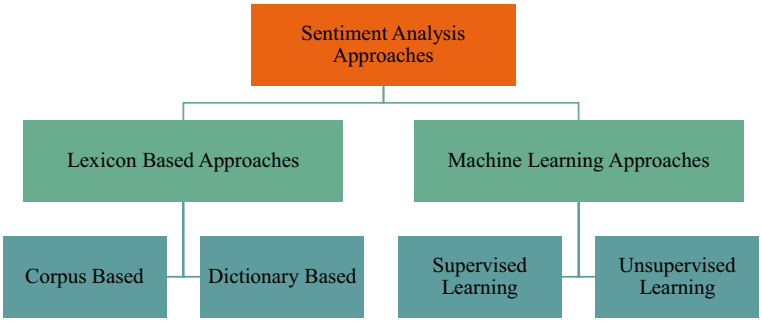

Fig. 1 Sentiment analysis approaches classification

ciency of the model with and without training the model on the target domain is investigated.

- A wide range of experiments was performed in this paper to not only achieve the optimal values for training the model but also demonstrate the sensitivity of the parameters and the effect of differing hyper-parameters on the transfer learning efficiency as well as choosing the best datasets to be utilized as the source and target domains.
- According to the results of experiments, the proposed model that employed BERT as the word representation model obtained higher classification accuracy compared to existing models, and applying transfer learning has also remarkably enhanced the overall accuracy.

The following of the paper is categorized as follows. Literature review is presented in Sect. 2. Section 3 includes the details of the proposed model 3. Model configuration, implementation details, and results of experiments are mentioned in Sect. 4. Section 5 also includes conclusions and directions for future research.

2 Related work

Sentiment analysis methods can be mainly classified into lexicon-based and machine learning techniques (Fig. 1). Lexicon-based approaches commonly compute the polarity of a sentence as the sum of polarities of individual words or phrases. These approaches employ a lexicon or dictionary which contains words and their corresponding polarity labels. While generating a semantic polarity lexicon requires human intervention besides being costly and time-consuming, recent studies have changed their direction to machine learning techniques and they have been mainly employed for sentiment classification [25]. Even though by the development of deep learning, deep neural networks have been at the center of attention and it can be said that they have made a revolution in different fields, especially various NLP tasks [8, 26]. Nowadays, the effect of deep learning in various tasks like text classification, document summarization, machine translation, language modeling, etc. is

completely obvious and sentiment analysis as one of the prominent aspects of NLP also obtained considerable improvement using deep neural networks [8, 27]. To provide a comprehensive comparison of the previous studies and highlight exiting challenges, existing studies are investigated based on three different aspects. Firstly, studies based on the employed method of the word representation are explored. Secondly, studies considering the utilized deep neural networks for sentiment analysis are fully studied. Thirdly, the impact of using the attention mechanism on the deep neural network is mentioned. Finally, the influence of transfer learning is investigated are studies in this era are discussed. More details are provided in the following section.

The first step in sentiment classification is word representation which aims to convert a text into fixed-size vectors [15]. Owing to the fact that the number of words in vocabulary obtained after the pre-processing step is limited, many studies have focused on tackling the problem of learning word embedding, and the model presented by Mikolov et al. [15], namely Word2Vec, was the first promising model in this regard. They used unlabeled text to train the continuous semantic representation of words. Similarly, Pennington et al. [16] utilized a co-occurrence matrix and performed training only on non-zero elements to generate semantic word embedding.

However, the mentioned word representation models are context-free and produce single word embedding for each word [20]. It means that the word "bank" would have the same vector in "riverbank" and "bank deposit" while the context can make difference in the meaning of the word "bank". To overcome this issue, recent language models have focused on generating contextual word embedding. Accordingly, ELMo [17] is a model that tried to extract context-sensitive features from the language model where the contextual representation of each token is the concatenation of its left-to-right and right-to-left representations. Devlin et al. [20] also proposed bidirectional encoder representations from transformers, namely BERT, to train deep bidirectional representations from unlabeled texts. BERT can incorporate information from bidirectional representations which are very important in representing words in natural language.

While word representation is completed, the next step is to combine word vectors into a document vector and then perform classification. In this regard, different deep neural networks were introduced that could tackle the lack of interactions among the target entity and its context. Moreover, different models like Recursive and Recurrent Neural Network (RNN) [28, 29], Convolutional Neural Network (CNN) [30], Deep Belief Network (DBN) [31], and Recursive Auto Encode (RAE) have been widely utilized for sentiment classification.

Accordingly, Kuta et al. [32] introduced tree structure GRU that used gated recurrent units in the tree structure LSTM and recursive model. Tai et al. [33] used LSTM with several complicated units to perform sentiment analysis. They carried out extensive experiments on two layers of bidirectional LSTM and obtained remarkable results. Kim et al. [30] performed various experiments on a one-layer convolutional neural network. Their model was trained on Word2Vec pre-trained vectors besides leveraging multi-channel representation and various filer sizes to increase the classification performance. Zhang et al. [34] introduced a character-level that yielded remarkable improvement in classification performance. Furthermore, a multichannel

variable size CNN was presented by Yin and Schutze [35] that used the integration of different word embedding techniques as input. A dynamic CNN employing dynamic k-max-pooling was also presented by Kalchbrenner et al. [36]. Considering the fact that their proposed model could handle input sentences with variable lengths, it was able to efficiently capture short and long-term dependencies. Socher et al. [28] proposed MV-RNN which is a variation of recursive neural networks and used both matrix and vector to show words and phrases in the tree structure. Socher et al. [28] also presented Recursive Neural Tensor Network (RNTN) that used a tensor-based compositional matrix in the tree structure. A combinational model was then introduced by Sadr et al [2] that aimed to make use of both convolutional and recursive neural networks for sentiment analysis. To take advantage of intermediate features obtained from recursive and convolutional neural networks, they also used multi-view learning to apply classification and achieved considerable results [24].

Although deep neural networks have obtained remarkable results for sentiment analysis, they are still in the primary steps of their extension and there is yet to be explored [3, 4]. Considering every word in the sentences equally and not being able to emphasize important parts of the text can be mentioned as one of their important drawbacks [10]. To overcome this issue, *attention mechanism* has been extensively utilized in different NLP task, particularly sentiment analysis, in recent years because they can provide an effective interpretation of the text. As a matter of fact, the idea behind the attention mechanism refers to the visual attention mechanism found in humans that aims to emphasize the more salient part of the text instead of encoding the full sentence. To this end, RNN was modified by Yang et al. [10] via including a weight that played the attention role for the task of text classification. An attention-based LSTM was also introduced by Wang et al. [37] that was bale to emphasize various parts of the sentences. Yuag et al. [38] proposed a domain attention model for multi-domain sentiment analysis that employed domain representation as attention to choose the most suitable domain-related features in each domain. Notably, although applying attention mechanism on deep neural networks have yielded promising results, only a small number of researches have been carried out for sentiment classification and its influence on CNN has been rarely investigated.

In the following, Er et al. [39] investigated the efficiency of convolutional neural networks with an attentive pooling schema for sentence classification with the aim of retaining the most important features at the pooling stage. Zhao and Wu [40] also proposed an attention-based convolutional neural network where the attention mechanism was applied before the convolutional layer aimed to generate a context vector for each word. Therefore, their model could automatically capture long-term contextual information and correlation between non-consecutive words. Lee et al. [41] proposed a model for specifying keywords discriminating negative and positive documents utilizing a convolutional neural network with word attention. Based on their model, while the convolutional neural network was completely trained, the word attention mechanism was then implemented to identify high contributing words using weights of a fully connected layer.

Moreover, Yin and Schutze [42] proposed an attentive convolutional neural network that extracted high-level features not only from local context but also from nonlocal context using an attention mechanism that was commonly utilized

in RNNs. Their proposed model presented higher flexibility and efficiency in sentence modeling with variable size context. Liu et al. [43] also investigated the combination of attention mechanism and convolutional neural network and proposed an attention gated convolutional neural network for text classification where the specialized convolutional encoder was used to generate attention weights from feature context windows of various sizes. Accordingly, Basiri et al. [44] proposed an Attention-based Bidirectional CNN-RNN Deep Model that utilized two distinct bidirectional LSTM and GRU layers to extract both past and future contexts by taking temporal information flow in both directions into account. They also utilized an attention mechanism on the output to pay more or less attention to various phrases. Phan et al. [45] also presented a feature ensemble model related to tweets containing fuzzy sentiment by considering various elements like word-type, lexical, position, semantic, and words sentiment polarity.

Lack of enough training data can be mentioned as another challenge of deep neural networks while they need a large number of training data to accurately train the model. While the number of data is enhanced, their efficiency is also increased [46]. In this regard and considering the lack of available labeled training data, f a new concept known as *transfer learning* has emerged in recent years. Transfer learning is utilized when enough training data are not available to train the model efficiently [47]. As a result, a large dataset (source domain) is used to train the model and then is transferred to the smaller dataset (target domain). It is worth mentioning that in spite of the fact that transfer learning has been widely utilized for image processing, its utilization in NLP, particularly sentiment analysis, is not extensively explored [13]. To this end, the effectivity of transferring low-level neural layers in various tasks was investigated by Krizhevsky and Lee [48]. In another research [49], the influence of transferring high-level layers in a deep neural network from the source dataset to the target dataset was explored. Notably, the influence of transfer learning for sentiment analysis has not been extensively investigated [12].

Even though deep learning methods have obtained remarkable results for the task of sentiment classification, they are still confronted with some limitations, and a strong need for progress in this field is felt. According to the previous studies and the mentioned challenges, we proposed an Attention-Based Convolutional Neural Network with Transfer Learning (ACNN-TL) that takes advantage of both attention mechanism and transfer learning. In contrary to existing studies, an attention mechanism is utilized after the convolutional layer in ACNN-TL to obtain more important words existing in the sentences by giving them a higher weight which yields to the generation of the new representation of word vectors. Thereafter, the proposed model utilized transfer learning to transfer knowledge from a source domain to different but relevant target domains to increase the classification accuracy. Moreover, due to the remarkable efficiency of contextual word representation in various natural language tasks in recent years [2, 14], we came up with the idea to investigate its influence on our proposed model. To this end, we also utilized traditional and contextualized word representations, namely Word2Vec and BERT, on the top of our proposed model individually.

3 Proposed methodology

Convolutional neural networks are generally considered as appropriate candidates for NLP because they are able to extract low-level features as well as controlling the length of dependencies. As a result, a hierarchical abstract representation of text can be generated while multiple convolutional layers are utilized. Moreover, the most significant features can be extracted using the pooling layer. Nonetheless, using the pooling layer may lead to the loss of local features having precious information. To fill this lacuna, an attention layer is used before the pooling layer in our proposed model to identify the important features besides conquering the influence of unimportant features and assisting the pooling layer to find the really salient features based on the context. Furthermore, since enhancing the number of training data commonly yields to convolutional neural network performance increment, the influence of transfer learning on the proposed model was also explored. On the other hand, traditional neural network language models commonly use Word2Vec as a representation of text information. However, Word2Vec has a limitation and does not consider context while generating word vectors. Accordingly, we decided to explore the importance of using a pre-trained language model BERT as the contextual feature representation of text information besides Word2Vec on our proposed model.

Overall, the most important contribution of our proposed model is that it not only utilizes context to find if a sequence of words is related rather than only filtering them without taking the context into account but also uses transfer learning as a held hand approach to boost the classification accuracy. Moreover, the influence of different traditional and contextual word embedding techniques is also investigated in our proposed model. The following section explains the details of the Attention-based Convolutional Neural Network with Transfer Learning (ACNN-TL) which contains two distinct subsections.

3.1 Attention-based Convolutional Neural Network (ACNN)

The learning process flow of Attention-based Convolutional Neural Network (ACNN) is presented in Fig. 2. ACNN has four layers while the attention layer is located before the pooling layer to select more important information. ACNN also has two variations while the first one employs Word2Vec [15] and the second one employs BERT [20] to generate word vectors. As it is clear, as word

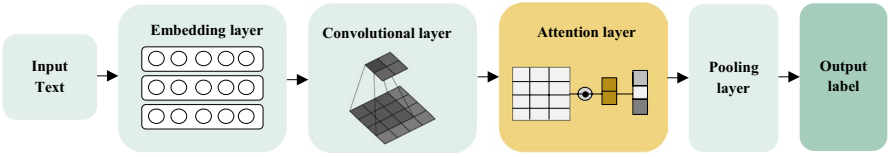

Fig. 2 Structure of the proposed ACNN

vectors of input sentences are generated, they are integrated to create the primary input matrix. Then, the convolutional filters are performed on the input matrix to extract the feature maps. While the training process is finalized, same-size feature maps are combined and given to the attention layer as a new matrix in the third layer. Thereafter, a higher weight is assigned to informative words using the attention mechanism. Then, by aggregating the representation of informative words to the features achieved by the convolutional layer, new sentence vectors are generated. Ultimately, new obtained word vectors are given to a fully connected layer to conduct classification. Details of each layer and its mathematical deduction are mentioned as follows.

3.1.1 Embedding layer

A sentence matrix is required as an input of convolutional neural network where each row refers to a word vector. Considering the embedding layer, ACNN has two variations while Word2Vec and BERT are individually utilized as the backbone to produce the contextualized representation of input sentences. Noteworthy, Word2Vec is two layers neural network which represents semantically similar words with neighboring points in the same vector space. Word2Vec is accessible in two various versions, namely Continuous Bag of Words (CBOW) and Skip-Gram. Considering the previous studies that confirm the effectiveness of Skip-Gram for sentiment analysis [14], it is also leveraged as the backbone of our proposed model. However, in spite of the remarkable success of Word2Vec in various NLP tasks, it is a context-independent word representation technique and therefore cannot consider the polysemy, and only one vector is generated for each word.

In contrast, BERT is a context-dependent word representation model that is obtained from masked language modeling and pre-trained bidirectional transformers. It can not only be utilized as a word representation technique to generate input vectors but also can be fine-tuned for a particular classification task [19]. It is worth mentioning that although BERT model performed well in text classification and obtained state-of-the-art results on many NLP tasks owing to its language comprehension ability, it is not still able to focus on information of phrases and salient words in the text. Therefore, we believe that using BERT and a backbone of ACNN can lead to great improvement in the overall performance.

In summary, if the word vector dimensionality is d and the given sentence length is s, the sentence matrix dimensionality, obtained from both Word2Vec and BERT, would be $s \times d$ and the sentence matrix is denoted by $A \in \mathcal{R}^{s \times d}$. The architectures of Word2Vec and BERT are illustrated in Fig. 3.

3.1.2 Convolutional layer

The convolutional layer is the primary building block of the CNN while it undertakes the principal portion of the computational load of the network. Convolutional operation must be performed to the sentence matrix to generate new features. Since the sequential structure of a sentence has a remarkable influence in identifying its

meaning, the filter width should be the same as the word vectors dimensionality (d). Accordingly, only filter height (h) that is known as region size can be different. In fact, a dot product between two matrices is applied in this layer. One matrix is the set of learnable parameters and the other one is the limited portion of the receptive field.

Assuming $A \in \mathcal{R}^{s \times d}$ as a sentence matrix, convolutional filter $H \in \mathcal{R}^{\boldsymbol{h \times d}}$ is performed on A to generate $A[i:j]$ as its submatrix. As the convolutional operation is repeatedly performed on the matrix of A, the output sequence $O \in \mathcal{R}^{s-h+1 \times d}$ is obtained (Eq. 1).

$$O_i = w \cdot A[i : i + h - 1] \tag{1}$$

where $i = 1 \ldots s - h + 1$ and $\cdot$ shows the dot product between two matrices of the convolution filter and input submatrix. An activation function and bias term $b \in \mathcal{R}$ are also included in each O_i. Untimely, feature maps $C \in \mathcal{R}^{\boldsymbol{s-h+1}}$ are produced (Eq. 2).

$$C_i = f(O_i + b) \tag{2}$$

3.1.3 Attention layer

While every word in a sentence does not have equal contribution in representing the sentence polarity and pooling layer in the convolutional neural network leads to loss of local features, there is a remarkable requirement for a mechanism to focus on words that have more influence on the polarity of the sentence considering the interaction and context of the words. To his end, an attention mechanism is applied to extracted feature maps.

In the following, feature maps that are obtained from the same filter size are aggregated to create a new matrix that is used in the attention layer. Assuming that M various region sizes are used and for each of them m various filters are utilized in the convolutional layer. Consequently, after performing $H_{ij} \in \mathcal{R}^{\boldsymbol{h_i \times d}}$ filters on sentence matrix

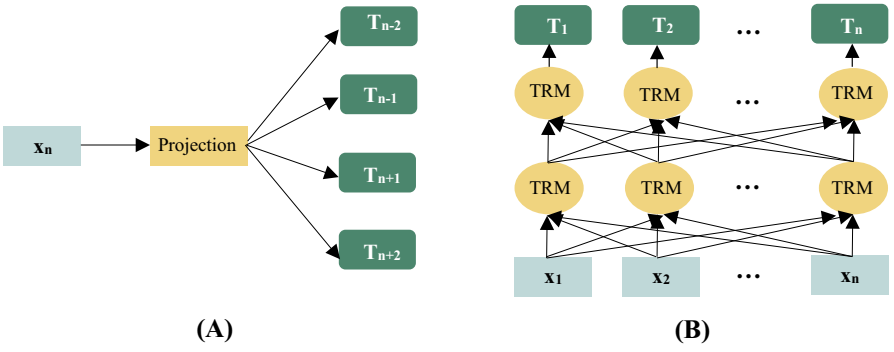

Fig. 3 **A** Word2Vec architecture (SkipGram version) where x_i is the input word, projection is the hidden layer that maps input word to output vectors and T_n is the generated output embedding vector [14]. **B** BERT architecture where x_i is the input word, TRM is the transformer block and T_n is the generated output embedding vector [19]

A ($i = 1, 2, \ldots, M$ and $j = 1, 2, \ldots, m$), $M \times m$ feature map is achieved. While feature maps obtained from the same filter size are merged, $X_i \in \mathcal{R}^{n \times m}$ is obtained as a new sentence matrix (Eq. 3). Here n refers to the word number and each element of this matrix shows the feature obtained from the input utilizing filters of the same size.

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

The main goal of the attention mechanism is to give particular weight to each row to extract important parts of the sentence. Accordingly, the new word matrix X_i is first given to a single layer perceptron utilizing $w \in \mathcal{R}^{\boldsymbol{m \times d}}$ and $U_i \in \mathcal{R}^{n-h_i+1 \times d}$ as a hidden representation of X_i is achieved (Eq. 4).

$$U_i = \tanh(X_i W + b) \tag{4}$$

Thereafter, the informatively of each word is calculated as the similarity of U_i with a context vector $u \in \mathcal{R}^{d \times 1}$ and to obtain $a_i \in \mathcal{R}^{n-h_i+1 \times 1}$ as the normalized importance weight, the Softmax function is utilized (Eq. 5). Noteworthy, the context vector u can be assumed as a high-level representation to determine important words [50, 51].

$$a_i = \text{softmax}(U_i u) \tag{5}$$

Specifically, the value of u is considered as zero at first to assume the same weight for different rows of the matrix X_i and its weight is updated during the training process. Thereafter, $\overline{X}_i$, as a new representation of X_i, is measured by multiplying each element of a_i to its corresponding row in X_i matrix ($\circ$ refers to the element-wise product) (Eq. 6).

$$\overline{X}_i = a_i \circ X_i \tag{6}$$

Consequently, as the attention mechanism is applied, $\overline{X}_i$ is obtained as a new representation of X_i. The process of the attention layer is illustrated in Fig. 4 while its pseudo-code is presented in Algorithm 1.

3.1.4 Pooling and fully connected layer

Considering the fact that different feature maps based on various filter sizes are produced, a pooling function is needed to produce fixed-size vectors. The main goal of the pooling layer is to express the most significant feature from each feature map as well as decreasing the dimensionality. Features that are generated from pooling layers based on each filter are then concatenated into a feature vector $\boldsymbol{o_i}$. The new feature vector is then fed to a fully connected Softmax layer to define the final class. In fact, Softmax is calculated as follows while it is used to determine the probability distribution over all sentiment categories (Eq. 7).

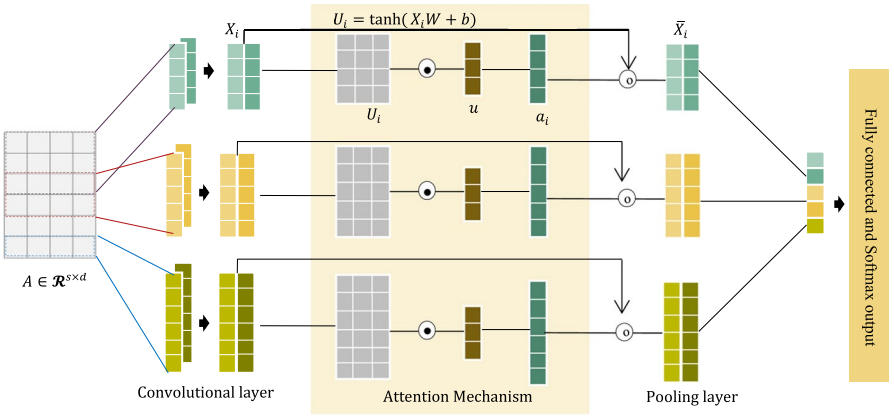

Fig. 4 ACNN structure

$$P_i = \frac{\exp\left(\boldsymbol{o_i}\right)}{\sum_{j=1}^{c} \exp\left(\boldsymbol{o_j}\right)} \tag{7}$$

To specify the difference between the obtained distribution $P_i(C)$ and the real sentiment distribution $\widehat{P}_i(C)$, cross-entropy is utilized as the loss function (Eq. 8).

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

where T refers to the training set and V indicates the sentiment label. Stochastic Gradient Descent (SGD) is also utilized to perform end-to-end training.

3.2 Attention-based Convolutional Neural Network with Transfer Learning (ACNN-TL)

Although deep learning can be considered as an efficient solution for various tasks, it is not a silver bullet being able to overcome all hidden problems especially when enough data is not available [3]. As a matter of fact, deep learning can be a perfect solution when enough training data is accessible and by increasing the number of training data, its efficiency is also enhanced. Convolutional neural network, as a representative of deep learning methods, is highly dependent on the size of training data. To solve the problem of insufficient training data, transfer learning has been introduced. In spite of the fact that transfer learning has been employed in various applications, its efficiency on NLP, particularly sentiment analysis, has been hardly investigated [13].

The idea behind transfer learning is to store knowledge learned from the source domain and then use it on various but relevant domains. Assume that $D_s = \left\{\left(x_{s1}, y_{s1}\right), \left(x_{s2}, y_{s2}\right), \ldots, \left(x_{sn}, y_{sn}\right)\right\}$ and

Algorithm: Attention-based convolutional neural network

Input: $A \in R^{n \times d}$ - A sequence of an independent variable obtained from the embedding layer
Output: A construct of ACNN
Parameters:
R= Number of rows in Matrix A
C= Number of columns in matrix A
k×*C*=Size of filters
m= Number of filters with the same size
n= Total number of filters
s= Stride of moving window
d= Dimension of content vector
b= Bias //Hyper-parameter

```
Def Applying Attention (Training data, R, C, k, m, n, s, d):
//Load vector representation obtained from embedding layer as an input
for c in range (0,n):// Convolutional kernel with specific number of filters, filter size and activation function
 for row in range (0,R):
  for col in range (0,C):
   for i in range (0,k):
    for j in range (0,C):
     //floating point mac
     temp=temp+ K [c][i][j]* A [s*row+i][s*col+j]
   F [c][row][col]= F [c][row][col]+temp
For all feature maps (F) obtained from the same filter sizes : //Applying attention mechanism on feature maps
 for i in range (0,m):
  for j in range (0,R-k+1):
   l=0
   X [j][i]= F [i][j][l]
  for i in range (0, R-k+1)://Applying one layer perceptron
   for j in range (0,d):
    U [i][j]=0
    for l in range (0,m):
     U [i][j]+=tanh ((X [i][l]*w[l][j])+b))
  for i in range (0, R-k+1): //Applying attention mechanism by multiplying U and content vector u
   j=0
   a [i][j]=0
    for l in range (0,d):
     a [i][j]+= Softmax (U [i][l]* u [l][j])
for i in range (0, m): // Updating feature maps
  for j in range (0 ,R-k+1):
   for l in range (0, R-k+1):
    t=0
    X̄[i][j][l]=X[i][j][l]*a[l][t]
return X̄
```

Pseudo-code of ACNN

$D_t = \{(x_{t1}, y_{t1}), (x_{t2}, y_{t2}), \ldots, (x_{tn}, y_{tn})\}$ respectively indicate the source and target domain and T_s and T_t show the source and target domain respectively. Furthermore, $x_{si} \in X_s$ and $x_{ti} \in X_t$ represent the *i*th data in the source and target domain and $y_{si} \in Y_s$ and $y_{ti} \in Y_t$ respectively indicate the *i*th label of the source and target domain. Consequently, transfer learning is utilized to increase the T_t of the target prediction function f_t at D_t using knowledge in D_s for T_s.

The employed transfer learning process is illustrated in Fig. 5. As it is obvious, ACNN is first trained on the source domain and then the trained model is transferred to the target domain. The suggested transferring also contains two various processes. In the first one ACNN is just tested on the target domain and it is not trained on the target domain (Fig. 5A) while in the second one ACNN is also trained and then tested on the target domain (Fig. 5B).

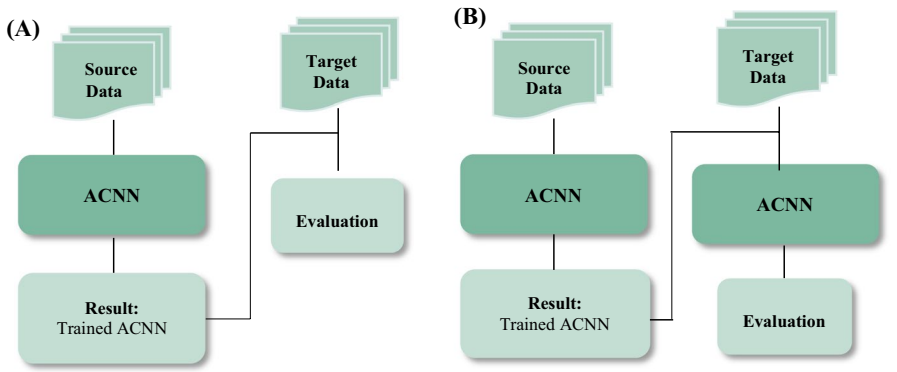

Fig. 5 The learning process using transfer learning. **A** Without training the model on target domain **B** With training the model on target domain

4 Experiments

Since the goal of this paper is on investigating the influence of word representation besides transfer learning, the experiments are divided into two sections. Firstly, the influence of the proposed attention mechanism on the efficiency of convolutional neural networks considering Word2Vec and BERT as the backbone for creating word representation is investigated. Secondly, the effect of transfer learning on the accuracy for sentiment classification besides investigating the sensitivity of the proposed model to different parameters on various datasets is explored. In this respect, more details about each set of experiments and obtained results besides the used datasets are presented in the following sections.

4.1 Dataset

Standard datasets are used in our experiments to prepare a comprehensive analysis of the effectiveness of the proposed model. Accordingly, six different datasets are used in our experiments to explore the effect of transfer learning. These datasets are categorized into two groups of source domains (D_s) and the target domain (D_t) where small size datasets were assumed as target domain while large size datasets were considered as the source domain. Summary statistics of the used datasets are mentioned in Table 1. As can be seen, target datasets are smaller in size and source datasets have larger values of metrics. More details of the used datasets are also mentioned in the following.

Table 1 Summary of source and target domain datasets

Dataset		C	L	S	V
D_S	AMZ-5	5	84	3,650,000	1,057,296
	AMZ-2	2	82	3,000,000	1,112,820
	YELP	2	141	560,000	246,735
	IMDB	2	257	25,000	81,321
D_T	SST-5	5	18	11,855	17,836
	SST-2	2	19	9613	16,185

*D_S: Source domain; D_T: Target Domain; *C*: Class numbers; *L*: Sentence length Average; *S*: Sentences number; *V*: Size of vocabulary

Source domain:

- Amazon Review (AMZ): It includes reviews related to Amazon products that were gathered by Zhang et al. [34]. It also includes two classes (AMZ-2) and five classes (AMZ-5) versions.[1]
- Yelp polarity review (YELP): It is a rather large size dataset including business reviews that are categorized into two classes[2] [34].
- IMDB: It has multi-sentence reviews about movies that are categorized into two classes[3] [52].

Target domain:

- Stanford Sentiment Treebank (SST): It is the commonly used dataset for the task of sentiment classification that has two classes (SST-2) and five classes (SST-5) versions [53].[4]

4.2 Effect of attention mechanism

This section includes experimental details that were conducted to explore the effect of the proposed attention mechanism on the convolutional neural network besides the effect of Word2Vec and BERT as word representation techniques. The empirical results are mentioned in the following.

4.2.1 Model configuration

Implementation was started with pre-processing which is generally known as an important step in NLP. To this end, CoreNLP[5] was used in our implementation. As previously mentioned, two different word representations techniques, namely Word-2Vec and BERT, were individually used as the backbone of the proposed model to

[1] https://goo.gl/bm0IkT.
[2] https://goo.gl/bm0IkT.
[3] https://goo.gl/NWatud.
[4] http://nlp.stanford.edu/sentiment/.
[5] https://stanfordnlp.github.io/CoreNLP/.

Table 2 Configuration of ACNN hyperparameters

Hyperparameters	Value
Size of filter region	3,4,5
Filter number	128
Dropout rate	0.5
Batch size	25
Activation Function	ReLU

not only enhance the efficiency of our proposed model but also make a comparison between context-independent and dependent representation models on the overall performance. In this regard, while preprocessing was completed and tokens were extracted, Skip-Gram and BERT were individually utilized as the backbone of ACNN to not only enhance the accuracy of our proposed model but also make a comparison between context-independent and dependent representation models on the overall performance.

To train word vectors using Word2Vec, the word vectors dimension was set to 200 while the window size was considered as 3. Word vectors were then updated with a learning rate of 0.01. To train BERT,[6] it was initialized based on $\text{BERT}_{\text{-base}}$ model [19] whose maximum length was 128, layer number was 12, embedding dimension was 768, and the learning rate was 0.0002. Therefore, three variations of the ACNN were employed in our implementation as follows:

- *ACNN-Rand:* Random initialized vectors are used as its input.
- *ACNN-Word2Vec:* Pre-trained word vectors obtained from Word2Vec are utilized as its input.
- *ACNN-BERT:* Pre-trained word vectors obtained from BERT are used as its input.

To train ACNN, ADADELTA update rule with a learning rate of 0.01 was used while the size of the mini-batch was set to 25. Hyper-parameters values are expressed in Table 2. As it is obvious, the filter size (3, 4, 5) and 128 filters yielded higher results on both datasets. Moreover, the dropout rate of 0.5 was used to regularize the convolutional layer while Rectified Linear Unit (ReLU) was leveraged as the activation. The model was also trained on 60 epochs. It must be mentioned that target datasets were just utilized in this set of experiments.

4.2.2 Results

To provide a baseline and carry out a fair comparison between the proposed model and other state-of-the-art, the proposed model was just trained on the target domains. So as to examine the efficiency of ACNN, we first compared it with traditional machine learning and deep learning-based models. To further understand the effect of employing contextual representation, we also compared our model with pre-trained

[6] https://github.com/google-research/bert.

Table 3 ACNN accuracy on target datasets

Group	Model			Accuracy %	
				SST-2	SST-5
A	NB [54]			81.8	41
	BiNB [54]			83.1	41.9
	SVM [55]			79.4	40.7
	WordVec-AVE [55]			80.1	32.7
B	CNN-1 layer [36]			77.1	37.4
	CNN-non static [30]			87.2	48
	CNN-multichannel [30]			88.1	47.4
	DCNN [36]			86.8	48.5
C	LSTM [33]			85.2	46.2
	Bi-LSTM [33]			87.5	49.1
	Tree-LSTM [33]			88.0	51.0
	Tree-GRU [56]			88.6	50.5
	Tree-GRU + attention [56]			89.0	51.0
	LSTM + RNN attention [37]			86.1	48.0
D	RecRNN [54]			82.4	43.2
	RNTN [54]			85.4	45.7
	MVRNN [28]			82.9	44.4
E	BERT-Base [57]			91.2	53.2
	BERT-Large [57]			93.1	55.5
	SentiBert [14]			92.7	56.1
	SentiBert w/o BERT [14]			86.5	50.3
	BERT + CNN [59]			90.1	51.7
	MT-DCNN [58]			94.3	-
	XLNet [59]			94.4	-
		Mean (std)	Max	Mean (std)	max
Proposed model	ACCN-Rand	88.64	88.91	49.68	49.78
	ACNN-Word2Vec	89.84	89.95	50.13	50.22
	ACNN-BERT	92.78	**92.81**	53.53	**53.65**

The highest values are highlighted in bold

*Some values are blank because the original authors of those papers did not publish their results

BERT and its variations that have been trained on SST-2 and SST-5 datasets. Accuracy comparison of ACNN against other existing models is provided in Table 3 where existing models are categorized into 5 groups from A to F. It is worth mentioning that the results of other existing models are taken from their original papers.

- *Group A* includes traditional machine learning models
- *Group B* includes convolutional neural network-based models.
- *Group C* includes recurrent neural network-based models.
- *Group D* includes recursive neural network-based models.

– *Group E* includes pre-trained BERT-based models and other models that utilized language models.

Noteworthy, due to the fact that the cost function of neural networks is commonly non-convex, training algorithms are only able to find the local optimum and consequently, various local optimums can be achieved during various runs of a model. To overcome this issue, we evaluated our proposed model using a mean and standard deviation of 5 runs while the best result among 5 runs is also mentioned.

By carefully considering Tables 3, it is obvious that models that are categorized in group E have relatively higher accuracy compared to other existing models. Moreover, among all variations of the proposed model, ACCN-Rand has the lowest accuracy on both datasets which can be attributed to the utilization of random initialized vectors as input. However, it is obvious that although it also has lower accuracy compared to the model of group E, it has higher accuracy compared to traditional deep learning-based models which can indicate the importance of incorporating the attention mechanism.

Considering other variations of our proposed model, namely ACNN-Word2Vec and ACNN-BERT, it can be concluded that vector representation has a considerable influence on the efficiency of ACNN while using pre-trained vectors led to an increase in the classification accuracy. Furthermore, it is also clear that employing BERT as the backbone of the ACNN has remarkably increased the performance of our proposed model while ACNN-BERT not only has higher accuracy compared to other variations of the proposed model but also presents comparable or better accuracy compared to the majority of the existing models which highlight the importance of contextualized representation besides the attention mechanism. It is worth mentioning that the higher accuracy of models like MT-DCNN and XLNet can be attributed to their baseline configuration and employment of enriched datasets for training while in this section our proposed model was just trained on target datasets. Accordingly, we hypothesized that using transfer learning besides considering larger size datasets as the source domain might increase the overall classification accuracy of our proposed model. Results indicated in the following sections present the effect of transfer learning and prove our hypothesis.

4.3 Effect of transfer learning

While insufficient training data can be considered an important challenge of deep neural networks, the influence of transfer learning on the efficiency of ACNN is explored in this paper. The following section includes more details in this regard.

4.3.1 Model configuration

Commonly, the free number of hyper-parameters can be mentioned as one of the drawbacks of CNNs. Due to the fact that values of hyper-parameters have considerable influence on the effectiveness of deep neural networks, the ACNN

hyper-parameters were optimized on the source domains and then the favorable parameters were then applied on the target domains using transfer learning. In this regard, the previously mentioned ACNN configuration was used as a baseline, and sets of various experiments were carried out to achieve their favorable values on a source domain to consider their influence on transfer learning. Noteworthy, ten-fold cross-validation where 10% of training data was randomly chosen as a test set was applied and each experiment was repeated 5 times and the average results are reported. It is worth mentioning that while ACNN-BERT presented the highest performance compared to other variations of the proposed model on both target domains, results that are provided in the following of this section are obtained by training this model on the source domain and other variations are ignored because training a large size dataset is a costly affair.

h● To investigate the impact of the *filter size*, various numbers of filter sizes were utilized while the other parameters were assumed constant. As it is illustrated in Table 4, various filter size has a significant influence on the effectiveness of the model and the best-obtained result is different with baseline value which indicates that the highest accuracies were obtained while the filter size was (4, 5, 6) in all source datasets.

Table 4 Filter size influence on the accuracy of ACNN-BERT based on various source domains

Filter size	Accuracy %			
	AMZ-2	AMZ-5	YELP	IMDB
(3, 4 ,5)	94.28	59.23	95.35	88.38
(4, 5, 6)	**94.36**	**59.84**	**96.72**	**90.74**
(6, 7, 8)	93.65	58.63	95.83	89.91
(8, 9, 10)	92.93	57.28	95.21	88.45
(9, 10, 11)	92.24	57.41	94.63	88.73
(14, 15, 16)	92.21	57.32	94.83	87.25
(3, 4, 5, 6)	93.08	58.12	94.23	86.23
(6, 7, 8, 9)	92.23	57.26	95.17	87.14

The highest values are highlighted in bold

Table 5 Number of filters influence on the accuracy of ACNN-BERT based on various source domains

Number of filters	Accuracy %			
	AMZ-2	AMZ-5	YELP	IMDB
128	93.25	58.12	95.63	**90.88**
256	93.92	58.74	**96.86**	89.73
300	**94.43**	**59.36**	96.32	88.64
512	94.21	59.12	92.87	87.13
450	92.36	57.28	93.24	88.63

The highest values are highlighted in bold

Table 6 Dropout rate influence on the accuracy of ACNN-BERT based on various source domains

Dropout rate	Accuracy %			
	AMZ-2	AMZ-5	YELP	IMDB
0.1	93.62	58.64	94.91	89.91
0.2	93.24	58.48	95.81	90.31
0.3	93.87	58.68	95.91	89.61
0.4	93.24	58.68	96.21	90.43
0.5	94.15	89.65	**96.81**	**90.91**
0.6	**94.92**	**59.93**	96.34	90.18
0.7	93.65	58.63	96.27	89.21
0.8	94.18	59.17	95.87	89.41
0.9	94.38	59.28	95.87	88.28

The highest values are highlighted in bold

Table 7 Activation function influence on the accuracy of ACNN-BERT based on various source domains

Activation function	Accuracy %			
	AMZ-2	AMZ-5	YELP	IMDB
Tanh	92.71	59.18	94.21	89.61
Softplus	92.31	52.41	95.23	90.21
ReLU	**94.39**	**59.61**	**96.92**	**90.97**
Linear	92.51	58.31	93.28	87.43

The highest values are highlighted in bold

- To investigate the impact of the *number of filters*, other configurations were kept constant while only the number of filters in each filter region was modified. Based on the empirical results (Table 5), it can be said that the number of filters has also a great effect on the efficiency of ACNN. The highest accuracy was achieved while the number of filters was 300 on both AMZ-2 and AMZ-5 datasets. Moreover, filter sizes of 256 and 128 respectively led to the highest classification accuracy on YELP and IMDB datasets.
- To explore the importance of dropout, different dropout rates on a scale of 0.1 to 0.9 were utilized. Based on the result of experiments (Table 6), the highest accuracy was obtained while the dropout rate was about 0.6 on AMZ-2 and AMZ-3 datasets and about 0.5 on IMDb and YELP datasets.
- To explore the impact of activation function, various activation functions, like linear, Tanh, ReLU, and SoftPlus were utilized in our experiments. Considering the result of experiments (Table 7), the ReLU function presented higher performance compared to other activation functions on all source datasets.

Table 8 Comparison of the performance of transfer learning with and without incremental learning

$D_S \rightarrow D_T$	Description	Accuracy (%)
AMZ-2 → SST-2	Transfer learning without incremental learning	90.23
	Transfer learning with incremental learning	94.48
AMZ-2 → SST-5	Transfer learning without incremental learning	52.73
	Transfer learning with incremental learning	56.23

Table 9 ACNN-TL Accuracy (%) on various source and target domain datasets

D_T	→ SST-2				→ SST-5			
D_S	AMZ-2	AMZ-5	YELP	IMDB	AMZ-2	AMZ-5	YELP	IMDB
ACNN-BERT	94.48	**94.51**	93.68	93.73	**56.23**	55.31	54.21	54.63

4.3.2 Results

As previously mentioned, the lack of enough labeled training data can be mentioned as one of the important challenges of deep neural networks while their efficiency is greatly related to the number of data and enhancing the number of training data has a remarkable influence on their performance. Transfer learning here comes to helps increase the size of the training set.

To this end, transfer learning was leveraged in our experiment and the proposed model was first trained on the source domains and then transferred to target domains. As it was previously mentioned, two various learning processes were utilized in our experiments (Fig. 4). Firstly, ACNN was directly employed for sentiment analysis in the target domain. Secondly, the model was incrementally trained on the target domain employing the favorable values that were previously discussed. The performance of the ACNN-BERT without and with incremental learning in the various source and target domains is mentioned in Table 8 which helps to provide a fair comparison between these two learning processes.

As it can be seen, the performance of the ACNN-BERT is lower when it is directly utilized for classification which confirms that the knowledge of the source domain is not enough to be performed on the target domain. On contrary, when the model is incrementally learned in the target domain, the accuracy is remarkably enhanced. It is worth mentioning that using incremental learning has also resulted in a higher accuracy of ACNN-BERT compared to other existing models. As a matter of fact, while AMZ-2 was used as the source domain and ACNN-BERT was incrementally learned in the target domain, an accuracy of 94.48 was obtained on SST2 dataset while the accuracy of the best existing model (XLNet) on SST2 dataset was about 94.4 which clearly demonstrates the priority of our proposed model.

Another set of experiments were also conducted to investigate the influence of using transfer learning with incremental learning on other datasets. Table 9 presents the results of using transfer learning with incremental learning on all variations of

Table 10 Number of words that are not present in the source data

Domain	SST-2	SST-5
AMZ-2	2477	331
AMZ-5	2492	348
YELP	3720	1697
IMDB	4003	2058

source and target domains to illustrate more analysis of the influence of transfer learning on sentiment analysis. As can be seen, using transfer learning has increased the overall classification accuracy of ACNN-BERT on both datasets.

It must be mentioned that different source domains resulted in various classification accuracy on target domains. Accordingly, it can be concluded that the primary challenge in transfer learning is deciding the most compatible source dataset for transferring knowledge to the target dataset. In this regard, the size of the source dataset and Out-Of-Vocabulary (OOV) can be considered as the important factors before deciding the most suitable source dataset. The number of OOV words for each target dataset not accessible in the source dataset is presented in Table 10. As it is clear, for example for IMDB and SST2 datasets, from the total 16,186 words of SST2, 4003 words are not present in IMDB.

As it is clear, AMZ-2 is the most semantically similar dataset to both SST-2 and SST-5 datasets while IMDB is the least semantically similar one. Considering the results of Table 2, it can be observed that AMZ-2 → SST-5 also obtained the highest classification accuracy. Similar result was also achieved for AMZ-5 → SST-2 while the source and target domains do not have the same number of classes. Therefore, it can be concluded while AMZ-2 and AMZ-5 datasets can be considered as good candidates for being used as the source datasets because they contain a large proportion of movie reviews that makes them semantically comparable to SST-2 and SST-5 datasets besides their large number of sentences which is much higher compared to other datasets that can provide richer contextual embedding. Finally, it can be mentioned that using transfer learning has greatly increased the classification accuracy which can be attributed to the large size of the source domain which provides richer embedding to help the model to better learn contextual information.

4.4 Discussion

Generally, convolutional neural networks assume that all words in a sentence have equal contributions in the sentence polarity and cannot highlight more important words. Furthermore, they are data-hungry and as the number of training data is enhanced, their performance is increased. To fill these lacunas, we proposed ACNN-TL in this paper that takes advantage of both attention mechanism and transfer learning. We also explored the effect of various word representation techniques, namely Word2Vec and BERT, on the performance of our proposed model. Extensive sets of experiments were carried out and based on the obtained results, it can be stated that:

- Using the proposed attention mechanism has considerably increased the classification performance of convolutional neural networks. In fact, the key difference of the ACNN compared to previous studies is that it uses the attention mechanism before the pooling layer which employs a context vector to discover if a sequence of words is relevant rather than only filtering them without considering the content.
- Using BERT as a contextualized representation technique on the backbone of ACNN has considerably improved the classification accuracy. As matter of fact, ACNN-BERT has higher performance compared to other variations of the proposed model and the majority of the state of the arts which clearly demonstrates the importance of using contextual word embedding.
- Although transfer learning with incremental learning can greatly enhance the accuracy, the performance of transfer learning in natural language processing is prone to the size and semantics of both source and target domain datasets. Based on the extensive set of experiments, it was found that size and OOV can be assumed as the significant factors for choosing the most suitable source dataset. As a result, as the source dataset is bigger and semantically more similar to the target dataset, higher performance can be archived.

5 Conclusion

Sentiment analysis has been at the center of attention in the last decade and the development of deep learning has made a great revolution in this field. In recent years, deep learning models have been widely explored and obtained remarkable results. However, the existing models have some limitation and their classification accuracy can be enhanced. To this end, we proposed attention-based convolutional neural network for the task of sentiment analysis that utilized the effectiveness of context-independent and context-dependent besides transfer learning to increase the overall classification performance. The proposed model uses an attention mechanism before the pooling layer to emphasize the important part of the sentences and considers the context for determining the polarity of sentences. The proposed model progressively forms sentence vectors by aggregating informative word vectors achieved from the attention layer into feature maps extracted from the convolutional layer and employs these newly generated features for classification. Moreover, the importance of context-independent and context-dependent representation techniques is also explored in our experiments. Finally, due to insufficient training data which is one of the prominent challenges in training deep neural networks, the effect of transfer learning besides the sensitivity of the proposed model to various parameters was also explored.

According to the empirical results, not only ACNN-BERT obtained the highest classification accuracy among all variations of the proposed model but also it outperformed the majority of existing models on both target datasets which clearly specifies the importance of using the attention mechanism besides contextualized vector embedding. In order to enhance the accuracy of ACNN-BERT, transfer learning with incremental learning was also utilized. To this end, ACNN-BERT was first

trained on the source domain and then fine-tuned on the target domain. Based on the obtained result, using transfer learning led to considerable improvement and ACNN-BERT obtained the highest accuracy compared to other existing models on both target datasets. Based on the empirical results, it can be also concluded that the semantic and of size the source domains have also a considerable influence on the efficiency of transfer learning. As a result, it is better to choose a bigger and semantically similar dataset as the source domain while transfer learning is applied.

As a part of future research, the proposed model can be utilized in different target domains or for other NLP tasks. Due to the fact that using contextualized representation as the backbone of our model, namely BERT, has incredibly improved the classification accuracy, using another context-dependent word embedding like RoBERTa is also worth exploring.

References

1. Salloum SA, Khan R, Shaalan K (2020) A survey of semantic analysis approaches. In: Joint European-US Workshop on Applications of Invariance in Computer Vision. Springer, pp 61–70
2. Sadr H, Pedram MM, Teshnehlab M (2019) A robust sentiment analysis method based on sequential combination of convolutional and recursive neural networks. Neural Process Lett 50:2745–2761
3. Yadav A, Vishwakarma DK (2020) Sentiment analysis using deep learning architectures: a review. Artif Intell Rev 53(6):4335–4385
4. Prabha MI, Srikanth GU (2019) Survey of sentiment analysis using deep learning techniques. In: 2019 1st International Conference on Innovations in Information and Communication Technology (ICIICT). IEEE, pp 1–9
5. Habimana O, Li Y, Li R, Gu X, Yu G (2020) Sentiment analysis using deep learning approaches: an overview. Sci China Inf Sci 63(1):1–36
6. Sadr H, Pedram MM, Teshnelab M (2019) Improving the performance of text sentiment analysis using deep convolutional neural network integrated with hierarchical attention layer. Int J Inf Commun Technol Res 11(3):57–67
7. 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
8. Pathak AR, Agarwal B, Pandey M, Rautaray S (2020) Application of deep learning approaches for sentiment analysis. In: Agarwal B, Nayak R, Mittal N, Patnaik S (eds) Deep learning-based approaches for sentiment analysis. Springer, Singapore, pp 1–31
9. Sadr H, Soleimandarabi MN, Pedram M, Teshnelab M (2019) Unified topic-based semantic models: a study in computing the semantic relatedness of geographic terms. In: 2019 5th International Conference on Web Research (ICWR). IEEE, pp 134–140
10. Yang Z, Yang D, Dyer C, He X, Smola A, Hovy E (2016) 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, pp 1480–1489
11. Soleymanpour S, Sadr H, Soleimandarabi MN (2021) CSCNN: cost-sensitive convolutional neural network for encrypted traffic classification. Neural Process Lett 53:3497–3523
12. Zhang Z, Zou Y, Gan C (2018) Textual sentiment analysis via three different attention convolutional neural networks and cross-modality consistent regression. Neurocomputing 275:1407–1415
13. Liu R, Shi Y, Ji C, Jia M (2019) A survey of sentiment analysis based on transfer learning. IEEE Access 7:85401–85412
14. Sun C, Qiu X, Xu Y, Huang X (2019) How to fine-tune Bert for text classification? In: China National Conference on Chinese Computational Linguistics. Springer, pp 194–206
15. Yin D, Meng T, Chang K-W (2020) SentiBERT: a transferable transformer-based architecture for compositional sentiment semantics. arXiv preprint, arXiv:2005.04114

16. Mikolov T, Chen K, Corrado G, Dean J (2013) Distributed representations of words and phrases and their compositionality. In: Nips
17. Pennington J, Socher R, Manning CD (2014) Glove: global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp 1532–1543
18. Peters ME et al (2018) Deep contextualized word representations. arXiv preprint, arXiv:1802.05365
19. Howard J, Ruder S (2018) Universal language model fine-tuning for text classification. arXiv preprint, arXiv:1801.06146
20. Radford A, Narasimhan K, Salimans T, Sutskever I (2018) Improving language understanding by generative pre-training
21. Devlin J, Chang M-W, Lee K, Toutanova K (2018) Bert: pre-training of deep bidirectional transformers for language understanding. arXiv preprint, arXiv:1810.04805
22. Sadr H, Nazari Solimandarabi M (2019) Presentation of an efficient automatic short answer grading model based on combination of pseudo relevance feedback and semantic relatedness measures. J Adv Comput Res 10(2):1–10
23. Sadr H, Nazari M, Pedram MM, Teshnehlab M (2019) Exploring the efficiency of topic-based models in computing semantic relatedness of geographic terms. Int J Web Res 2(2):23–35
24. Sadr H (2021) An intelligent model for multidimensional personality recognition of users using deep learning methods. J Inf Commun Technol 47(47)
25. Sadr H, Pedram MM, Teshnehlab M (2020) Multi-view deep network: a deep model based on learning features from heterogeneous neural networks for sentiment analysis. IEEE Access 8:86984–86997
26. Salloum SA, Khan R, Shaalan K (2020) A survey of semantic analysis approaches. In: Proceedings of the International Conference on Artificial Intelligence and Computer Vision (AICV2020), pp 61–70. Springer International Publishing, Cham
27. Aliakbarpour H, Manzuri MT, Rahmani AM (2021) Improving the readability and saliency of abstractive text summarization using combination of deep neural networks equipped with auxiliary attention mechanism. J Supercomput. https://doi.org/10.1007/s11227-021-03950-x
28. Young T, Hazarika D, Poria S, Cambria E (2018) Recent trends in deep learning based natural language processing. IEEE Comput Intell Mag 13(3):55–75
29. Socher R, Huval B, Manning CD, Ng AY (2012) Semantic compositionality through recursive matrix-vector spaces. In: Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, Association for Computational Linguistics
30. Al-Smadi M, Qawasmeh O, Al-Ayyoub M, Jararweh Y, Gupta B (2017) Deep recurrent neural network vs. support vector machine for aspect-based sentiment analysis of Arabic Hotels' reviews. J Comput Sci 27:386–393
31. Kim Y (2014) Convolutional neural networks for sentence classification. arXiv preprint, arXiv:1408.5882
32. Ruangkanokmas P, Achalakul T, Akkarajitsakul K (2016) Deep belief networks with feature selection for sentiment classification. In: Uksim.Info, p 16
33. Kuta M, Morawiec M, Kitowski J (2017) Sentiment analysis with tree-structured gated recurrent units. In: Ekštein K, Matoušek V (eds) Text, speech, and dialogue. TSD 2017. Lecture Notes in Computer Science, vol 10415. Springer, Cham, pp 74–82
34. Tai KS, Socher R, Manning CD (2015) Improved semantic representations from tree-structured long short-term memory networks. arXiv preprint, arXiv:1503.00075
35. Zhang X, Zhao J, LeCun Y (2015) Character-level convolutional networks for text classification. Adv Neural Inf Process Syst 28:649–657
36. Yin W, Schütze H, Xiang B, Zhou B (2015) Abcnn: attention-based convolutional neural network for modeling sentence pairs. arXiv preprint, arXiv:1512.05193
37. Kalchbrenner N, Grefenstette E, Blunsom P (2014) A convolutional neural network for modelling sentences. arXiv preprint, arXiv:1404.2188
38. Wang Y, Huang M, Zhao L (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
39. Yuan Z, Wu S, Wu F, Liu J, Huang Y (2018) Domain attention model for multi-domain sentiment classification. Knowl-Based Syst 155:1–10

40. Er MJ, Zhang Y, Wang N, Pratama M (2016) Attention pooling-based convolutional neural network for sentence modelling. Inf Sci 373:388–403
41. Zhao Z, Wu Y (2016) Attention-based convolutional neural networks for sentence classification. In: INTERSPEECH, pp 705–709
42. Lee G, Jeong J, Seo S, Kim C, Kang P (2017) Sentiment classification with word attention based on weakly supervised learning with a convolutional neural network. arXiv preprint, arXiv:1709.09885
43. Yin W, Schütze H (2018) Attentive convolution: equipping CNNS with RNN-style attention mechanisms. Trans Assoc Comput Linguist 6:687–702
44. Liu Y, Ji L, Huang R, Ming T, Gao C, Zhang J (2019) An attention-gated convolutional neural network for sentence classification. Intell Data Anal 23(5):1091–1107
45. Basiri ME, Nemati S, Abdar M, Cambria E, Acharya UR (2021) ABCDM: an attention-based bidirectional CNN-RNN deep model for sentiment analysis. Futur Gener Comput Syst 115:279–294
46. Phan HT, Tran VC, Nguyen NT, Hwang D (2020) Improving the performance of sentiment analysis of tweets containing fuzzy sentiment using the feature ensemble model. IEEE Access 8:14630–14641
47. Semwal T, Yenigalla P, Mathur G, Nair SB (2018) A practitioners' guide to transfer learning for text classification using convolutional neural networks. In: Proceedings of the 2018 SIAM International Conference on Data Mining. SIAM, pp 513–521
48. Zhuang F et al (2019) A comprehensive survey on transfer learning. arXiv preprint, arXiv:1911.02685
49. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. Adv Neural Inf Process Syst 25:1097–1105
50. Sermanet P, Eigen D, Zhang X, Mathieu M, Fergus R, LeCun Y (2013) Overfeat: integrated recognition, localization and detection using convolutional networks. arXiv preprint, arXiv:1312.6229
51. Sukhbaatar S, Weston J, Fergus R (2015) End-to-end memory networks. Adv Neural Inf Process Syst 28:2440–2448
52. Kumar A et al (2016) Ask me anything: dynamic memory networks for natural language processing. In: International Conference on Machine Learning, pp 1378–1387
53. Maas A, 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, Association for Computational Linguistics
54. Pang B, Lee L (2005) Seeing stars: exploiting class relationships for sentiment categorization with respect to rating scales. In: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics, pp 115–124. Association for Computational Linguistics
55. Socher R et al (2013) Recursive deep models for semantic compositionality over a sentiment Treebank. In: EMNLP
56. Du C, Huang L (2017) Sentiment classification via recurrent convolutional neural networks. In: DEStech Transactions on Computer Science and Engineering, no cii
57. Kokkinos F, Potamianos A (2017) Structural attention neural networks for improved sentiment analysis. arXiv preprint, arXiv:1701.01811
58. Munikar M, Shakya S, Shrestha A (2019) Fine-grained sentiment classification using BERT. In: 2019 Artificial Intelligence for Transforming Business and Society (AITB), vol 1, pp 1–5. IEEE
59. Zheng S, Yang M (2019) A new method of improving BERT for text classification. In: International Conference on Intelligent Science and Big Data Engineering, pp 442–452. Springer
60. Liu X, He P, Chen W, Gao J (2019) Improving multi-task deep neural networks via knowledge distillation for natural language understanding. arXiv preprint, arXiv:1904.09482
61. Yang Z, Dai Z, Yang Y, Carbonell J, Salakhutdinov RR, Le QV (2019) Xlnet: Generalized autoregressive pretraining for language understanding. Adv Neural Inf Process Syst 32:5753–5763

Authors and Affiliations

Hossein Sadr[1] · **Mojdeh Nazari Soleimandarabi**[2]

Mojdeh Nazari Soleimandarabi
Mojdeh.nazari@qiau.ac.ir

[1] Department of Computer Engineering Rahbord Shomal Institute of Higher Education, Rasht, Iran

[2] Cardiovascular Diseases Research Center, Department of Cardiology, Heshmat Hospital, School of Medicine, Guilan University of Medical Sciences, Rasht, Guilan, Iran