IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received October 16, 2018, accepted October 29, 2018, date of publication November 9, 2018, date of current version December 3, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2879481

Attention-Based Memory Network for Text Sentiment Classification

HU HAN, JIN LIU, AND GUOLI LIU
School of Electronic & Information Engineering, Lanzhou Jiaotong University, Lanzhou 730070, China

Corresponding author: Hu Han (hanhu_lzjtu@mail.lzjtu.cn)

This work was supported in part by the National Social Science Foundation of China under Grant 17BXW071, in part by the National Natural Science Foundation of China under Grant 71461017, and in part by the Technology Program of Lanzhou Science and Technology Bureau under Grant 214162.

ABSTRACT In order to explore the impact of different memory modules under the framework of memory network for aspect level sentiment classification, we use convolutional neural networks (CNN) and bidirectional long short term memory (BiLSTM) to design four kinds of memory network models in this paper. The first model uses CNN for building a memory module, which is capable of capturing local information in documents. The second uses BiLSTM for building another memory module, which captures sequence information in documents. At the same time, the following two models use CNN and BiLSTM for building memory module——one builds a hierarchical neural network by using CNN and BiLSTM for building memory module, which combines both local and sequence information together. The other, respectively, uses CNN and BiLSTM for building two memory modules, respectively, which captures local information and sequence information through different modules. And then, we combine the final representations, which are generated from the different memory networks, for further sentiment classification. Experiments on laptop and restaurant datasets demonstrate that our four methods achieving better results than MemNet. In particular, the latter two models achieve better performance than the first two models and feature-based support vector machine approach.

INDEX TERMS Sentiment classification, deep memory network, convolutional neural networks, bidirectional long short term memory.

I. INTRODUCTION

Opinions are key influence to almost all of our activities, since whenever we need to make a decision, e.g., when we purchase products, we want to know others' opinions. Nowadays, many people show their opinions on the Internet, which is beneficial for researchers to devise automatic procedures to identify whether these textual reviews/opinions are positive or negative. Mining opinions from these texts is known as sentiment analysis, which is a special task of text classification whose objective is to automatically classify a document according to the sentiment polarities of opinions which contains, e.g., positive or negative, like or dislike [1]–[3].

Aspect level sentiment classification is a fine-grained task in the field of sentiment analysis. Since it provides more complete and in-depth results [4], [5]. The sentiment polarity of a sentence depends on not only the content but also the aspect. For example, in sentence ‘‘USB3 Peripherals are noticably less expensive than the ThunderBolt ones’’, there are two aspects, ‘‘USB3 Peripherals’’ and ‘‘ThunderBolt’’, and the sentiment polarity of ‘‘ThunderBolt ’’ is negative while ‘‘USB3 Peripherals’’ is positive. Most existing works use machine learning algorithms, and build sentiment classifier from sentences with manually annotated polarity labels. One of the most successful approaches in literature is feature based SVM [6]. Experts could design effective feature templates and make use of external resources like parser and sentiment lexicons. In recent years, neural network approaches are drawing growing attention capacity on learning powerful text representation from data [7], [8]. Nguyen and Shirai proposed a method which is an extension of RNN (Recursive Neural Network) that takes both dependency and constituent trees of a sentence into account [9]. Futher more, neural networks combined with attention mechanisms gaining much popularity on aspect level sentiment classification [10]–[12]. Attention-based LSTM, which can concentrate on different parts of a sentence when different aspects are taken as input, is proposed by Wang *et al.* [13]. Target-Dependent LSTM (TD-LSTM) and Target-Connection LSTM are proposed by Tang [14], the relatedness of a target word with its context words is taken into account in the model. It selects the relevant parts of contexts to infer the sentiment polarity towards the target.

Memory network is a general machine learning framework introduced by Weston [15]. Its central idea is inference with a long-term memory component, which could be read, written to, and jointly learned with the goal of using it for prediction. In 2016, Tang *et al.* [16] proposed to use memory network for aspect level sentiment classification, which captures importance of context words. It is verified that the proposed approach performs better than LSTM architectures. However, local information and sequence information from original sentence are not taken into account, and it only uses the sequence of word vectors as memory module.

In order to further explore impact of different memory modules for aspect level sentiment classification, we use CNN and BiLSTM for designing four kinds of memory network models in this paper. The first model uses CNN for building a memory module, which is capable of capturing local information in documents. The second model uses BiLSTM for building another memory module, which is capable of capturing sequence information in documents. And at the same time, the following two models use CNN and BiLSTM for building memory module — one of them builds a hierarchical neural network by using CNN and BiLSTM to build memory module, which combines both local and sequence information together the other one of them respectively uses CNN and BiLSTM for building two memory networks, Then we combine the final representations together, which are generated from the different memory networks, for further sentiment classification. which captures local information and sequence information through different modules. Experimental results show that our method outperforms most of the baseline methods and the state-of-the art approaches. The main contribution of this work are as follows:

a) Under the framework of deep memory network, we use CNN and BiLSTM for designing different memory modules for aspect level sentiment classification.
b) A novel memory network framework is proposed. We respectively use CNN and BiLSTM for building double memory modules, which captures local information and sequence information through different modules. Then we combine the final representations, which are generated from the different memory networks, for further sentiment classification.
c) The experimental results demonstrate that our models achieve better performance comparing with the model only using the input words vectors as memory module. Especially, the last two models achieve better performance. At the same time it is proved that using CNN and BiLSTM to build memory module at the same time is effective.

II. RELATED WORK

This work is connected to four research areas in natural language processing. We briefly describe related studies in each area.

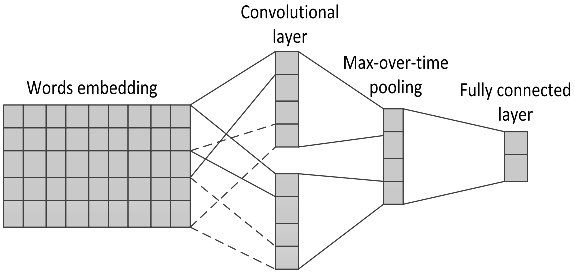

FIGURE 1. Convolutional neural networks.

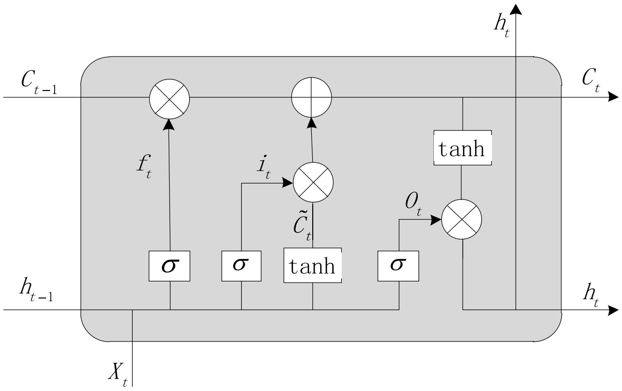

FIGURE 2. Long short term memory networks.

A. CONVOLUTIONAL NEURAL NETWORKS (CNN)

In recent years, Convolutional Neural Network(CNN) has been widely applied to many areas and achieved remarkable success [17]–[19]. It studies the input local feature and captures important feature information mainly through convolutional layer and pooling layer. In natural language processing tasks, CNN does not require a lot of pre-processing operation on texts, which substantially reduces the work of feature engineering. As is shown in Figure 2, CNN mainly involves input layer, convolutional layer, pooling layer and fully connected layer. The input layer is a vector representation for the input data. For a given sentence of length n, the input layer matrix can be represented as following:

$$E \in \mathrm{R}^{d \times n} \tag{1}$$

Where d is the word vector dimension. the convolutional layer using different filters convolve the input matrix, extracts local features and obtains feature vector map of convolutional kernels, which is shown in formula 2:

$$c = f(W \cdot x + b) \tag{2}$$

Where x is the word vector matrix of convolutional kernel windows, W is the weight matrix, b is a bias term and f is a activation function for convolutional kernels. Pooling layer is a important neural layers for CNN, For the feature vector graph obtained by the convolutional layer, we can extract important feature by negative-sampling operation for the feature vector graph through pooling layer. Simultaneously, the pooling layer can output a fixed-size matrix. We can get an output having same dimensions for sentence inputs of different lengths and convolutional kernels of different sizes through pooling layer, and then those outputs can be passed to a fully connected layer and get the final classification results.

B. LONG SHORT-TERM MEMORY (LSTM)

In feed-forward or recurrent networks, when the computational graph becomes very deep, the neural network optimization algorithm will face a problem of long-term dependence - because the deeper structure makes the model lose the ability to learn the previous information, making optimization extremely difficult. This problem is more prominent in RNNs. The Long Short-Term Memory (LSTM) and its derivatives can address this problem well [20]–[22]. The basic architecture of LSTM is illustrated in Figure 2.

For the t-th word in a sentence, the LSTM takes as input the word embedding x_t, the previous output h_{t-1} and cell state C_{t-1} and computes the next output h_t and cell state C_t. The detailed computation process is as follow:

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \tag{3}$$

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \tag{4}$$

$$\overset{\sim}{C_t} = \tanh(W_C x_t + U_C h_{t-1} + b_C) \tag{5}$$

$$C_t = f_t \odot C_{t-1} + i_t \odot \overset{\sim}{C_t} \tag{6}$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \tag{7}$$

$$h_t = o_t \odot \tanh(C_t) \tag{8}$$

where $\sigma(\cdot)$ denotes the logistic sigmoid function. The operation $\odot$ denotes the element-wise vector product. At each time step t, there are an input gate i_t, a forget gate f_t, an output gate o_t, a memory cell C_t and a hidden unit h_t, h_0 and C_0 can be initialized to 0 and the parameters of the LSTM is W, U, b.

C. ATTENTION

Inspired by human visual attention, the attention mechanism which is introduced into the Encoder-Decoder framework to select the reference words in source language for words in target language and in machine translation, is proposed by Bahdanau *et al.* [23]. It is also used in image caption generation and natural language question answering [24], [25]. It is known that not all words or sentences contribute equally to these work. In order to capture the crucial components over different semantic levels, many research introduce attention mechanism to document classification and aspect level sentiment classification. Attention based hierarchical neural network was proposed for document classification [26]. It applies attention mechanisms at the word- and sentence-level, which make the model to pay more or less attention to individual words and sentences when constructing the representation of the document. Chen proposed a hierarchical neural network which incorporate user and product information via word and sentence level attention sentiment classification [27].

D. MEMORY NETWORKS

Recently, Weston introduced memory networks as an inference components combined with a long-term memory component [15], this memory can be read and written. Memory network architecture consists of an array of objects called memory m and four components I, G, O and R. where m is an array of objects such as an array of vectors. Among these four components, input was convert to internal feature representation. G updates old memories with new input, and O generates an output representation given a new input and the current memory state, and R outputs a response based on the output representation. Motivated by the success of memory network in many NLP field [28], [29], Tang introduced a deep memory network model named as MemNet for aspect level sentiment classification [16] It used the sequence of word vectors as memory module and adopted the multiple attentions as computational layers to read and write memory modules. Chen used the Bidirectional LSTM(BiLSTM) for building the memory module and adopted the multiple attentions with a GRU network to read memory information [30].

III. METHODOLOGY

In this section, we describe our models in detail. Firstly, the task definition and notation are given. Afterwards, we give an overview of the framework of memory network. Lastly, we give four strategies to build memory cell from input sentence and give a brief introduce to how to evaluate aspect-level sentiment classification.

A. TASK DEFINITION AND NOTATIONS

We represent a review as a sentence S with n words $S = \{w_1, w_2, \cdots, w_i, \ldots w_n\}$,where w_i is an aspect word. Aspect-level sentiment classification aims to infer the numeric rating (1-5 or 1-10 stars) or sentiment polarity(positive, neutral, negative) of these reviews towards the aspect w_i according to their text information. For example, the sentiment polarity of sentence "Best of all is the warm vibe, the owner is super friendly and service is fast" towards aspect "vibe" and "service" is both positive.

When dealing with a text corpus, we use **word2vec to** embed each word into a low-dimensional, continuous and real-valued semantic space $L \in R^{d\times|V|}$ where d is the dimension of word vectors and $|V|$ is the vocabulary size.

B. AN OVERVIEW OF THE APPROACH

The basic structure of our method is the same as MemNet proposed by Tang *et al.* [16], one difference between our model and MemNet is which we redesign four kinds of memory models by using CNN and BiLSTM. In order to better introduce our work, the basic structure of MemNet is described in Figure 3, which consist of four modules: embedding layer, memory cell, multiple computational layers and Softmax layer. In every computational layer, we take aspect vector as input to adaptively select important slice

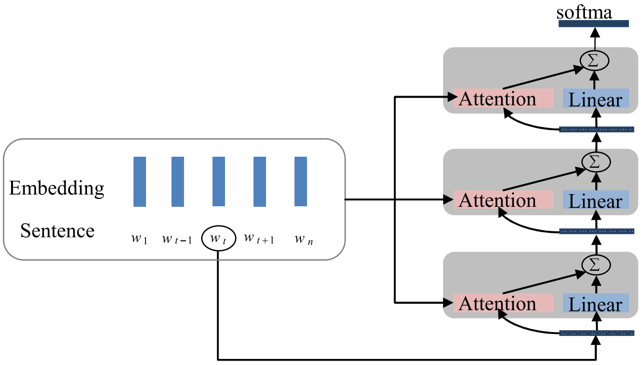

FIGURE 3. MemNet structure.

from memory module output through attention layer. The computational layer's output and the linear transformation of aspect vector are summed and the result is considered as the input of next computational layer.

Taking an embedding layer output as external memory **m** and an aspect vector v as computational layer's input, the attention model outputs a continuous vector *vec*. The output vector is computed as a weighted sum of each piece of memory in **m**, namely

$$vec = \sum_{i=1}^{k} \alpha_i m_i \tag{9}$$

where α_i measures the importance for each piece of memory m_i for current sentence and k is the memory size. For each piece of memory m_i we compute its semantic relatedness with aspect, the score function is defined as:

$$h_i = \tanh(W_H[m_i; v] + b) \tag{10}$$

where W_H is weight matrices Then we feed h_i to softmax function to calculate the final importance scores $\{\alpha_1, \alpha_2, \ldots, \alpha_k\}$

$$\alpha_i = \frac{\exp(h_i)}{\sum_{j=1}^{k} \exp(h_j)} \tag{11}$$

C. MEMORY CELL

In this work, we study four strategies to encode the input sentence in the memory cell. The first model is Cnn_MemNet which takes word vectors as input to CNN layer in order to compose word context representation and learn local information from words as memory module. The second model is BiLstm_MemNet which takes word vectors as input to BiLSTM layer in order to build the memory. The third model is CnnBiLstm_MemNet which builds a hierarchical neural network by using CNN and BiLSTM for building memory module. The last model is Cnn_BiLstm_MemNet which respectively uses CNN and BiLSTM for building two memory modules and captures local information and sequence information through different modules. For detailed information, please see the following introduction.

Model 1 (Cnn_MemNet): In the comment: "This dress is too expensive, but the quality not good", obviously "good" is a positive word, but when we added "not" before "good", the sentiment polarity of "quality" is negative. So capturing local information from input sentence has huge impact to sentiment analysis accuracy. Due to CNN being capable of capturing context information from sequence, we take word vectors as input to CNN layer to compose word context representation and learn local information from words as memory module. An illustration of Cnn_MemNet is given in Figure 4.

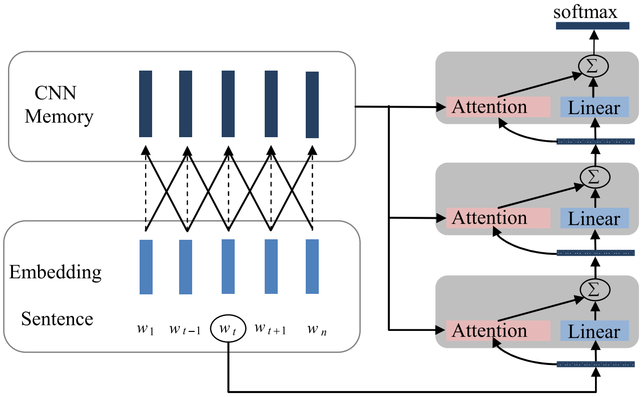

FIGURE 4. Cnn_MemNet.

Given an input word w_i as current word, we employ filters with window sizes h = 3:

$$w_{context(i)} = f(W_{i-1}^{j} \cdot w_{i-1} + W_i^j \cdot w_i + W_{i+1}^{j} \cdot w_{i+1} + b_i^j) \tag{12}$$

$$m = [w_{context(1)}, w_{context(2)}, \cdots, w_{context(n)}] \tag{13}$$

Model 2 (BiLstm_MemNet): MemNet [16] simply used the sequence of word vectors as memory, which cannot synthesize phrase-like features in the original sentence. It is straightforward to achieve the goal with the models of RNN family. In this paper, we use Deep Bidirectional LSTM (DBLSTM) [30] for building the memory which records all information to be read in the subsequent modules. At each time step t, the forward LSTM not only outputs the hidden state $\vec{h}_t^l$ at its layer $l(\vec{h}_t^0 = v_t)$ but also maintains a memory $\vec{c}_t^l$ inside its hidden cell. The update process at time t is as follows:

$$i = \sigma(\vec{W}_i \vec{h}_t^{t-1} + \vec{U}_i \vec{h}_{t-1}^{l}) \tag{14}$$

$$f = \sigma(\vec{W}_f \vec{h}_t^{t-1} + \vec{U}_f \vec{h}_{t-1}^{l}) \tag{15}$$

$$o = \sigma(\vec{W}_o \vec{h}_t^{l-1} + \vec{U}_o \vec{h}_{t-1}^{l}) \tag{16}$$

$$g = \tanh(\vec{W}_g \vec{h}_t^{l-1} + \vec{U}_g \vec{h}_{t-1}^{l}) \tag{17}$$

$$\vec{c}_t^l = f \cdot \vec{c}_{t-1}^l + i \cdot g \tag{18}$$

$$\vec{h}_t^l = o \cdot \tanh(\vec{c}_t^l) \tag{19}$$

where σ and tanh are sigmoid and hyperbolic tangent functions, $\overrightarrow{h} = [\overrightarrow{h^l_1}, \overrightarrow{h^l_2}, \overrightarrow{h^l_3}, \cdots, \overrightarrow{h^l_t}]$ is forward LSTM. The backward LSTM does the same thing, except that its input sequence is reversed $\overleftarrow{h^l_t}$. In our framework, we use BiLSTM to build the memory $m = [\overrightarrow{h}, \overleftarrow{h}]$ and it generally performs well in aspect level sentiment classification, the illustration of BiLstm_MemNet is given in Figure 5.

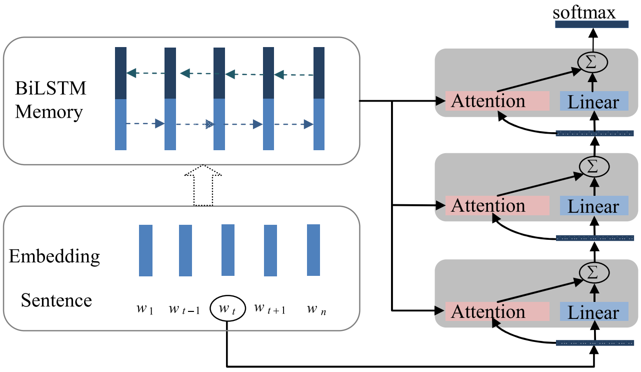

FIGURE 5. **BiLstm_MemNet.**

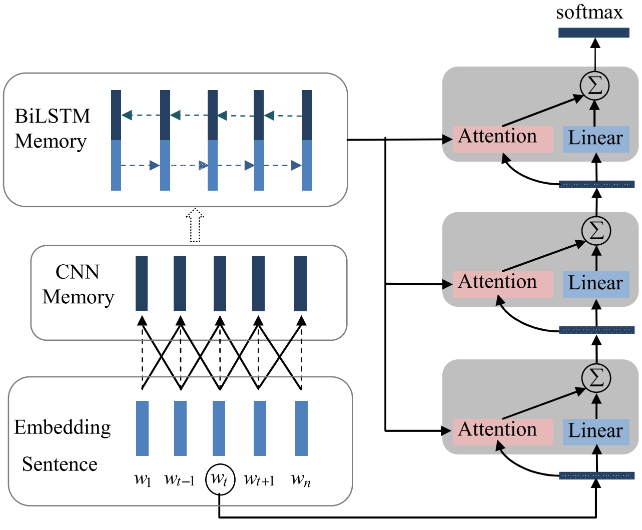

FIGURE 6. **CnnBiLstm_MemNet.**

Model 3 (CnnBiLstm_MemNet): CNN and BiLSTM are used to build a hierarchical neural network as memory module, which combines both local and sequence information together. Firstly, we embed each word into a low dimension semantic space, and take word vectors as input to CNN layer for the purpose of composing word context representation.

Given an input word w_i as current word, filter window size as 3, and the output of CNN is:

$$w_{context(i)} = f(W^j_{i-1} \cdot w_{i-1} + W^j_i \cdot w_i + W^j_{i+1} \cdot w_{i+1} + b^j_i) \tag{20}$$

Secondly, word context vectors which is the output of CNN are then fed into BiLSTM layer to learn memory representation. The illustration of CnnBiLstm_MemNet is given in Figure 6.

Model 4 (Cnn_BiLstm_MemNet): In this model, CNN and BiLSTM are respectively used for building two memory modules, and capturing local information and sequence information through different modules. Firstly, we apply two individual memory network to generate two representations. And then we design a combined strategy to make use of the two representations for training and final prediction. The illustration of Cnn_BiLstm_MemNet is given in Figure 7.

D. MODEL TRAINING

We apply our model to aspect level sentiment classification under supervised learning framework. Sentence representation d is extracted from the words in the Sentence, and it is a high level representation of the document. The sentiment classifier is built from documents with gold standard sentiment labels.

We use sigmoid to build the classifier because its outputs can be interpreted as conditional probabilities. Softmax is calculated as given in formula 21:

$$p_c = \frac{\exp(d_c)}{\sum_{k=1}^{C} \exp(d_k)} \tag{21}$$

Where C is the category number, p_c is predicted probability of sentiment class c. In our model, cross-entropy error between gold sentiment distribution and our model's sentiment distribution is defined as loss function for optimization, where D is the training set.

$$L = -\sum_{d \in D} \sum_{c \in C} p^g_c(d) \cdot \log(p_c(d)) \tag{22}$$

IV. EXPERIMENTAL RESULTS

In this section we evaluate our approach. Firstly we introduce the experimental setting and then report empirical results.

A. EXPERIMENTAL SETTING

We evaluate the effectiveness of our approach on two datasets from SemEval 2014, one is from Laptop domain and another is from Restaurant domain [4] The statistics of the datasets are summarized in Table 1. We split each corpus into training and testing sets in the proportion of 8:2.

We use the 300-dimensional word embeddings pre-trained by Glove and also use metrics Accuracy which measures the overall sentiment classification performance.

$$Accuracy = \frac{T}{N} \tag{23}$$

Where T is the numbers of predicted sentiment rating that are identical with gold sentiment ratings, and N is the numbers of documents.

B. EXPERIMENTAL RESULTS

We compare our memory network with several baseline methods for Laptop and Restaurant datasets.

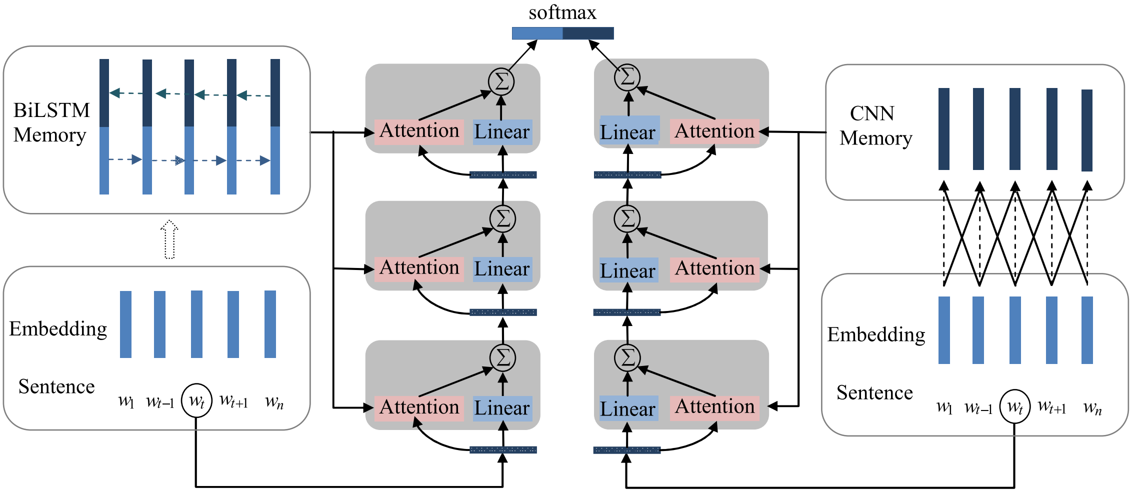

FIGURE 7. Cnn_BiLstm_MemNet.

TABLE 1. Details of the experimental datasets.

Dataset		Negative	Neutral	Positive
Laptop Reviews	Train	858	454	980
	Test	128	171	340
Restaurant Reviews	Train	800	632	2159
	Test	195	196	730

TABLE 2. Experimental results.

Model	Laptop	Restaurant
Feature+SVM	70.49	80.16
TD-LSTM	71.83	78.00
MemNet	70.33	78.16
Cnn_MemNet	70.89	78.73
BiLstm_MemNet	71.35	79.62
CnnBiLstm_MemNet	71.85	80.46
Cnn_BiLstm_MemNet	**72.41**	**80.96**

(1) **Feature+SVM** [6]: It extracts some text features such as surface features, lexicon features and parsing features, and train a SVM for sentiment classification.

(2) **TD-LSTM** [14]: It uses two LSTM networks towards the aspect, and incorporate one attention on the outputs of forward and backward LSTMs

(3) **MemNet** [16]: It applies attention multiple times on the word embeddings, and the last attention's output is fed into softmax for prediction. Obviously, MemNet is a simplistic form of our approach, and it's means memory cell is word embedding.

We use the same Glove word vectors and tune the hyper parameters on the validation sets and use ADAM to update parameters when training. Experimental results are given in Table 2.

It can be found that feature+SVM outperforms other methods except for CnnBiLstm_MemNet and Cnn_BiLstm_MemNet on Restaurant datasets, which demonstrates the importance of a powerful feature representation for aspect level sentiment classification. Among five memory network models, Cnn_MemNet, BiLstm_MemNet, CnnBiLstm_MemNet and Cnn_BiLstm_MemNet all perform better than MemNet, which indicates that CNN can extract local feature and BiLstm can learn the long-term dependencies and the positional relation of feature of the whole sentence. Particularly BiLstm_MemNet is a few better than Cnn_MemNet, which explains that BiLstm_MemNet not only synthesize features of word sequences but also preserves local information partially.

C. EFFECTS OF WIDTHS OF CONVOLUTIONAL FILTER

As is shown in Table 3, the performance by using word context is better than that only using word embedding for sentiment classification. Although LSTM has the ability to capture sequence semantics, CNN can capture local information, like phrase 'not bad'. Noting that word context is heavily dependent on filter size In order to study the effects of word context on sentiment semantic learning, we examine the performance of our model under different filter size and number of filters. The performance of the model for varying widths in CNN layer is presented in Table 3, and we find that filter size has distinct impact on the ability of CNN

TABLE 3. The effects of widths of convolutional filter.

Dataset		Laptop		Restaurant	
Model		Cnn_MemNet	CnnBiLstm_MemNet	Cnn_MemNet	CnnBiLstm_MemNet
Width	3	70.73	71.72	78.60	80.40
	4	70.80	**71.85**	**78.73**	**80.46**
	5	**70.89**	71.79	78.67	**80.46**

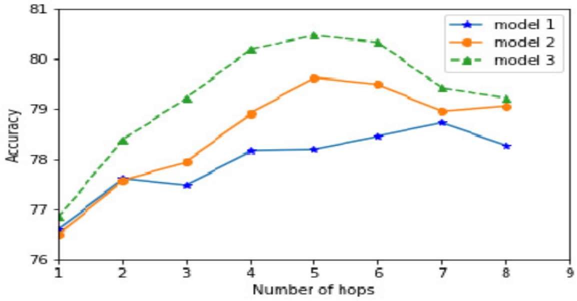

FIGURE 8. The impact of attention layers on the restaurant dataset.

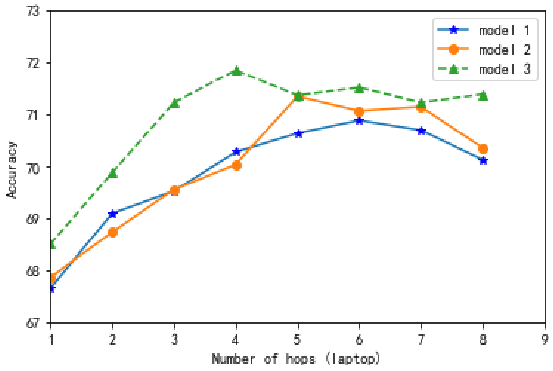

FIGURE 9. The impact of attention layers on the Laptop dataset.

to capture word context semantics and the optimal filter size.

D. EFFECTS OF ATTENTION LAYERS

One major setting that affects the performance of our model is the number of attention layers. We evaluate our models with 1 to 8 attention layers. The classification accuracy of model 1, model 2 and model 3 on the restaurant and Laptop datasets are shown in Figure 8 and Figure 9

As can be seen that model 2 achieves the best performance on the two datasets with 5 attention layers. Model 1 achieves the best performance on the restaurant dataset with 7 attention layers, and also achieves the best performance on the Laptop dataset with 6 attention layers. it is proved that using multiple attention layers might be sufficient to capture the sentiment features in complicated cases than using single attention layer. However, the performance is not monotonically increasing with the number of attention layers. For the restaurant dataset, using 8 attention layers is worse than using 7 attention layers in model 1, and using 6 attention layers is worse than using 5 attention layers in model 2. And we can see that the same thing happens on the Laptop dataset. It is because that the model will become more difficult to train and generalize with the complexity increasing.

TABLE 4. Effects of diffrernt Attention layers.

L	*R*	Laptop	Restaurant
4	6	72.29	80.75
4	7	71.60	79.94
4	8	71.33	79.52
5	6	**72.41**	**80.96**
5	7	72.00	80.29
5	8	71.00	79.46
6	6	71.45	79.70
6	7	71.54	79.84
6	8	71.40	79.59

Based on the above experimental results, we design nine combinatorial strategies to evaluate the effects of attention layers for model 4. For convenience, we use L and R to represent the left and right parts of model 4.

The specific combination strategy and experimental results are shown in Table 4.

The results demonstrate that suitable combination can effectively improve the performance of the model. For example, the fourth combination method, as shown in Figure 10, achieves the best performance among five memory network models. All combination methods get better results than model 1, and 6 combination methods get better results than model 2. It is proved that capturing local information and sequence information through different modules is very effective.

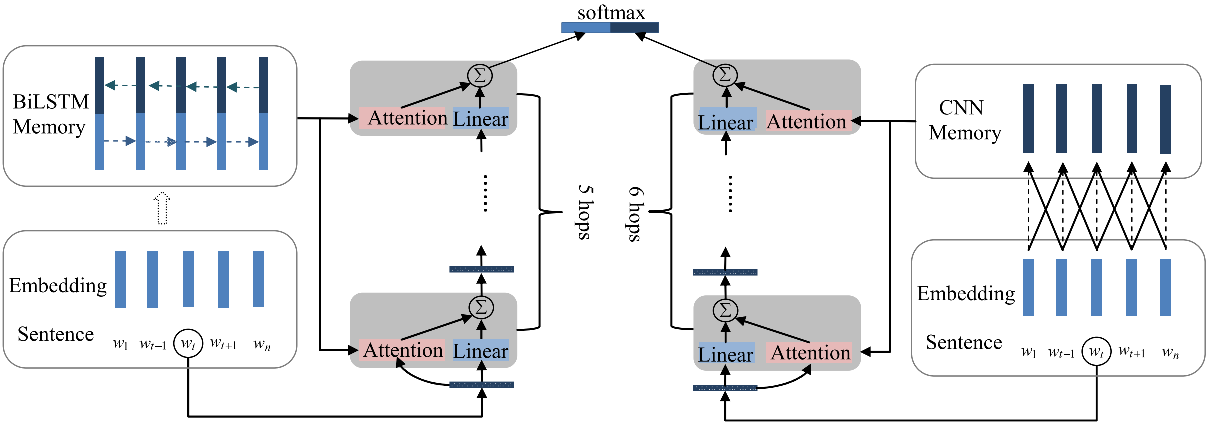

FIGURE 10. **The structure of the best combination model.**

V. CONCLUSION

In the paper, we design four kinds of memory network models under the framework of deep memory network. The first two models respectively use CNN and BiLSTM for building memory module, and it aims to capture local information or sequence information in documents. The next two models use CNN and BiLSTM for building memory module at the same time, and its goal is not only to capture local information, but also to capture sequence information. Experimental results demonstrate that all our models get better performance than MemNet, and the next two models are better than the first two models. It is proven that using CNN and LSTM to build memory modules is effective, which can capture local information and sequence information. Especially, the last model achieves the best result, which means that memory network framework with double memory models is effective and feasible.

REFERENCES

[1] B. Pang and L. Lee, "Opinion mining and sentiment analysis," *Found. Trends Inf. Retr.*, vol. 2, nos. 1–2, pp. 1–135, 2008.

[2] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up? Sentiment classification using machine learning techniques," in *Proc. EMNLP*, 2002, pp. 79–86.

[3] B. Liu and L. Zhang, "A survey of opinion mining and sentiment analysis," in *Mining Text Data*. New York, NY, USA: Springer, 2012, ch. 1, pp. 415–463.

[4] M. Pontiki, D. Galanis, J. Pavlopoulos, H. Papageorgiou, I. Androutsopoulos, and S. Manandhar, "Semeval-2014 task 4: Aspect based sentiment analysis," in *Proc. 8th Int. Workshop Semantic Eval. (SemEval)*, 2014, pp. 27–35.

[5] S. Kiritchenko, X. Zhu, C. Cherry, and S. Mohammad, "Detecting aspects and sentiment in customer reviews," in *Proc. 8th Int. Workshop Semantic Eval. (SemEval)*, 2014, pp. 437–442.

[6] S. Kiritchenko, X. Zhu, C. Cherry, and S. M. Mohammad, "NRC-Canada-2014: Detecting aspects and sentiment in customer reviews," in *Proc. 8th Int. Workshop Semantic Eval.*, 2014, pp. 437–442.

[7] Z. Zhang and M. Lan, "Ecnu: Extracting effective features from multiple sequential sentences for target-dependent sentiment analysis in reviews," in *Proc. 9th Int. Workshop Semantic Eval.*, Denver, CO, USA, 2015, pp. 736–741.

[8] X. Wang, Y. Liu, C. Sun, B. Wang, and X. Wang, "Predicting polarities of tweets by composing word embeddings with long short-term memory," in *Proc. 53rd Annu. Meeting Assoc. Comput. Linguistics 7th Int. Joint Conf. Natural Lang. Process.*, Beijing, China, 2015, pp. 1343–1353.

[9] T. H. Nguyen and K. Shirai, "PhraseRNN: Phrase recursive neural network for aspect-based sentiment analysis," in *Proc. Conf. Empirical Methods Natural Language Process.*, 2015, pp. 2509–2514.

[10] V. Mnih, N. Heess, and A. Graves, "Recurrent models of visual attention," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, Montreal, QC, Canada, 2014, pp. 2204–2212.

[11] W. Linlin, Z. Cao, G. de Melo, and Z. Liu, "Relation classification via multi-level attention CNNs," in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics*, Berlin, Germany, 2016, pp. 1298–1307.

[12] P. Zhou *et al.*, "Attention-based bidirectional long short-term memory networks for relation classification," in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics*, Berlin, Germany, 2016, pp. 207–212.

[13] Y. Wang, M. Huang, X. Zhu, and L. Zhao, "Attention-based LSTM for aspect-level sentiment classification," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, Austin, TX, USA, 2016, pp. 606–615.

[14] D. Tang, B. Qin, X. Feng, and T. Liu. (2015). "Effective LSTMs for target-dependent sentiment classification." [Online]. Available: https://arxiv.org/abs/1512.01100

[15] J. Weston, S. Chopra, and A. Bordes. (2014). "Memory networks." [Online]. Available: https://arxiv.org/abs/1410.3916

[16] D. Tang, B. Qin, and T. Liu, "Aspect level sentiment classification with deep memory network," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2016, pp. 214–224.

[17] J. Li, J. Li, X. Fu, M. A. Masud, and J. Z. Huang, "Learning distributed word representation with multi-contextual mixed embedding," *Knowl.-Based Syst.*, vol. 106, pp. 220–230, Aug. 2016.

[18] N. Kalchbrenner, E. Grefenstette, and P. Blunsom, "A convolutional neural network for modeling sentences," in *Proc. 52nd Annu. Meeting Assoc. Comput. Linguistics*, 2014, pp. 1–11.

[19] Y. Kim, "Convolutional neural networks for sentence classification," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, Doha, Qatar, 2014, pp. 1746–1751.

[20] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[21] M. Hu and B. Liu, "Mining and summarizing customer reviews," in *Proc. 10th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2004, pp. 168–177.

[22] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 3104–3112.

[23] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proc. Int. Conf. Learning Represent. (ICLR)*, 2015, pp. 1–15.

[24] K. Xu *et al.*, "Show, attend and tell: Neural image caption generation with visual attention," in *Proc. ICML*, 2015, pp. 2048–2057.

[25] W. Yin, H. Schütze, B. Xiang, and B. Zhou, "ABCNN: Attention-based convolutional neural network for modeling sentence pairs," *Trans. Assoc. Comput. Linguistics*, vol. 4, no. 11, pp. 259–272, 2016.

[26] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, "Hierarchical attention networks for document classification," in *Proc. NAACL*, 2016, pp. 1480–1489.

[27] H. Chen, M. Sun, C. Tu, Y. Lin, and Z. Liu, "Neural sentiment classification with user and product attention," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2016, pp. 1650–1659.

[28] A. Graves, G. Wayne, and I. Danihelka. (2014). "Neural turing machines." [Online]. Available: https://arxiv.org/abs/1410.5401

[29] S. Sukhbaatar, A. Szlam, J. Weston, and R. Fergus, "End-to-end memory networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 2431–2439.

[30] P. Chen, Z. Sun, L. Bing, and W. Yang, "Recurrent attention network on memory for aspect sentiment analysis," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2017, pp. 452–461.

HU HAN was born in Lanzhou, China, in 1977. He received the M.S. degree in computer application technology and the Ph.D. degree in traffic information engineering and control from Lanzhou Jiaotong University, China, in 2005 and 2011, respectively. He is currently an Associate Professor with the School of Electronic & Information Engineering, Lanzhou Jiaotong University. His current research interests include machine learning and data mining.

JIN LIU was born in Lüliang, Shanxi, China, in 1994. She received the bachelor's degree from the College of Physics and Electronic Engineering, Northwest Normal University, China, in 2016. She has been a graduate student with the School of Electronic and Information Engineering, Lanzhou Jiaotong University, China, since 2016. Her current research interests include natural language processing and deep learning.

GUOLI LIU was born in Baiyin, Gansu, China, in 1995. She received the bachelor's degree from the Department of Electrical Engineering, Shanghai Maritime University, China, in 2017. Since 2017, she has been a graduate student with the School of Electronic and Information Engineering, Lanzhou Jiaotong University. Her current research interests include sentiment analysis and deep learning.