Convolutional Multi-Head Self-Attention on Memory for Aspect Sentiment Classification

Yaojie Zhang, Bing Xu, and Tiejun Zhao

***Abstract*—This paper presents a method for aspect based sentiment classification tasks, named convolutional multi-head self-attention memory network (CMA-MemNet). This is an improved model based on memory networks, and makes it possible to extract more rich and complex semantic information from sequences and aspects. In order to fix the memory network's inability to capture context-related information on a word-level, we propose utilizing convolution to capture *n*-gram grammatical information. We use multi-head self-attention to make up for the problem where the memory network ignores the semantic information of the sequence itself. Meanwhile, unlike most recurrent neural network (RNN) long short term memory (LSTM), gated recurrent unit (GRU) models, we retain the parallelism of the network. We experiment on the open datasets SemEval-2014 Task 4 and SemEval-2016 Task 6. Compared with some popular baseline methods, our model performs excellently.**

***Index Terms*—Aspect sentiment classification, deep learning, memory network, sentiment analysis (SA).**

I. Introduction

ASPECT based sentiment analysis (ABSA) [1]–[3] is a detailed sentiment analysis task which aims to analyze the sentiment polarity (positive, negative or neutral) expressed by different aspects of the same text. In many cases, we need to focus not only on the overall sentiment in product reviews, as in ordinary sentiment analysis (SA) tasks, but also on more detailed and in-depth sentiment expressions. The sentiment expressions of different aspects in a sentence may be different. For example, in the sentence "Good performance, but too little battery power.", there is a positive attitude towards "performance", but a negative attitude towards "battery". This task is important and challenging, and many shared task studies have been conducted in recent years, such as SemEval-2014 Task 4 [3], SemEval-2015 Task 12 [4], and SemEval-2016 Task 5 [5]. ABSA tasks are generally divided into aspect extraction (AE) subtasks [6] and aspect sentiment classification (ASC) subtasks [7]. With the development of a series of related research, the task definition of ABSA has become more complete. It is divided into three parts [8]: opinion target extraction (OTE), aspect category detection, and sentiment polarity (SP). This paper mainly studies SP task; that is, given a sentence with some aspects, how one analyzes the sentiment polarity of aspects in the sentence. SP/ASC can be divided into two types: aspect-category sentiment analysis (ACSA) and aspect-term sentiment analysis (ATSA) [9]. The main difference between ACSA and ATSA is that ACSA classifies many kinds of targets to be analyzed into several categories, and identifies the sentiment polarity of each aspect category in the sentences. The goal of ATSA is to directly identify the sentiment polarity of targets being analyzed, whose categories are uncertain. This paper studies these two tasks.

Early research used traditional methods based on rules [10] or statistics [11]. Support vector machine (SVM) with external resources [12] is one of the most successful methods, but its performance depends heavily on the construction of artificial features. Target dependent (TD)-LSTM (long short term memory) and target connection (TC)-LSTM [13] take the prediction target as the central word and build two LSTMs from left to right and from right to left. Considering that only using LSTM will result in information loss when processing long sequences, aspect-attention-aspect-embedding (ATAE)-LSTM [7] uses an aspect-related attention mechanism. However, these LSTM based methods are always difficult to integrate statements with dispersed important feature locations. For example, in the sentence "Everything except memory is terrible.", "except" and "terrible" have a positive effect on the word "memory". Reference [14] first applied memory network to ABSA and achieved good results. LSTM has strong aspect-sequence modeling ability, but it loses context-related information besides word-level, and lacks the modeling of complex semantic expression. Although multi-layer attention can alleviate this defect, it only focuses on the semantic relationship between aspect and sequence, and ignores the semantic relationship between the words of the sequence. There are many subsequent improvements based on memory in ABSA tasks [15]–[18], and they have all achieved good results, but most lose network parallelism.

To solve the aforementioned problems, we propose to use convolution to integrate text features of words and multi-words, and use a multi-head self-attention of transformer [19] encoder instead of recurrent neural network (RNN) to extract semantic information in the sequence. The output of the encoder is then used as memory. Convolutional multi-head self-attention is first proposed in hierarchical convolutional

Manuscript received February 22, 2020; accepted March 27, 2020. This work was supported by the National Key Research and Development Program of China (2018YFC0830700). Recommended by Associate Editor Chenglin Liu. *(Corresponding author: Yaojie Zhang.)*

Citation: Y. J. Zhang, B. Xu, and T. J. Zhao, "Convolutional multi-head self-attention on memory for aspect sentiment classification," *IEEE/CAA J. Autom. Sinica*, vol. 7, no. 4, pp. 1038–1044, Jul. 2020.

The authors are with the Laboratory of Machine Intelligence and Translation, Department of Computer Science, Harbin Institute of Technology, Harbin 150001, China (e-mail: 17862702586@163.com; hitxb@hit.edu.cn; tjzhao@hit.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JAS.2020.1003243

attention network (HCAN) [20]. HCAN is a hierarchical feature extraction method for document-level text classification. Finally, we classify the aspect's sentiment polarity with the help of an aspect-oriented memory network. In this way, the model considers long-term dependence information of aspect words and sequences by aspect attention, context-related information besides word-level by convolutional calculation and considers semantic-related information of sequence itself by self-attention. This is an improved model based on memory network, and makes it possible to extract more complex and richer semantic information from sequences and aspects. The whole model retains the parallelism of network computing. Each component is differentiable, and can be trained end-to-end with gradient descent. We evaluate our approach on four typical datasets: three from SemEval 2014's laptop dataset and restaurant review dataset [3], and one from SemEval 2016's tweets dataset [21]. We apply datasets to ACSA and ATSA tasks respectively. The experimental results show that our model performs well on different types of data for two kinds of tasks.

The rest of this paper is organized as follows. Section II introduces our methods in detail. Section III introduces our experimental results and analysis on open datasets. Section IV describes some of our summaries and future work directions.

II. METHOD

In this section, we will introduce our method for ACSA and ATSA tasks. The ACSA task is defined as: given a sentence and an aspect category, the model predicts the sentiment polarity (positive, negative or neutral) of the sentence to the aspect category. The ATSA task starts with being: given a sentence and an aspect (usually one or more words) that appears in the sentence. The model predicts the sentiment polarity of the sentence to the aspect. The overall structure of model is shown in Fig. 1.

A. Embedding

Embeddings is the input layer of the model. Sentences contain word embedding and position embedding, while aspect only contains word embedding. The input sentence $s = \{w_1, w_2, \ldots, w_n\}$ contains n words, and aspect is $a = \{w'_1, w'_2, \ldots, w'_m\} \subseteq s$ or $a = \{w'_{ac}\}$, where $m \in [1, n)$. We map all words to a low-dimensional continuous real space $\boldsymbol{E} \in \mathbb{R}^{|V| \times d}$, where d is the dimension of word vector and $|V|$ is the size of the dictionary. So, a sentence and aspect can be expressed as $\boldsymbol{s}_e \in \mathbb{R}^{n \times d}$, $\boldsymbol{a}_e \in \mathbb{R}^{m \times d}$ respectively.

Location information is important for ABSA tasks. In many cases, we must find modifiers associated with the aspect terms to determine the polarity of the aspects. This kind of association may have a syntactic dependency or other more complex semantic association. We use a randomly initialized position embedding to learn the representation of information associated with aspect terms. We regard aspect words as the central word, marked as 0. The words in front of the aspect words begin with the central words and are marked $-1, -2, \ldots$ in turn while the words after the aspect words are marked $1, 2, \ldots$ in turn (if there is no aspect words in the sentence, start with the first word, making it as 1). Then, mapping all integers to a real space $\boldsymbol{P} \in \mathbb{R}^{2L \times d}$. L is the longest sentence length, and d is the same dimension as word embedding. Then, we get $\boldsymbol{s}_p \in \mathbb{R}^{n \times d}$.

We add word embedding and position embedding to get sentence representation. The final embedding is represented as

$$\boldsymbol{E}^s = \boldsymbol{s}_e + \boldsymbol{s}_p \tag{1}$$

$$\boldsymbol{E}^a = \boldsymbol{a}_e. \tag{2}$$

B. Convolution Operation

In Fig. 2, we use text convolution on the sentence embedding $\boldsymbol{E}^s$ to extract features. Just like research in [20], we use the same convolution method with different parameters to generate three different feature matrices $\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}$ on $\boldsymbol{E}^s$. This has stronger manifestation than directly using the $\boldsymbol{E}^s$ as $\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}$, as each convolution operation can extract different features from $\boldsymbol{E}^s$. Moreover, convolution can extract n-gram information effectively and quickly. Unlike [20]'s research, in order to keep the sequence length unchanged before and after convolution, we add zero vectors before and after the original sequence. The advantage of this is to keep the aspect aligned with each token in the sequence when querying; Reference [20] does not need to do this because their goal is document-level text classification. We let x be k, q, v. Similarly, $\boldsymbol{x}$ is $\boldsymbol{k}, \boldsymbol{q}, \boldsymbol{v}$ and $\boldsymbol{X}$ is $\boldsymbol{K}, \boldsymbol{Q}, \boldsymbol{V}$. Convolution with the weight matrix $\boldsymbol{W}^x_{\text{conv}} \in \mathbb{R}^{w \times d}$ and bias variable $b^x_{\text{conv}} \in \mathbb{R}$ whose window size is w

$$x_i = f(\boldsymbol{W}^x_{\text{conv}} \cdot \boldsymbol{E}^s_{i:i+w-1} + b^x_{\text{conv}}) \tag{3}$$

where f is a exponential linear unit (ELU) or tanhyperbolic (Tanh), and i is the index of sequence after padding. Take d filters, and get matrices $\boldsymbol{X} \in \mathbb{R}^{n \times d}$.

The experiments in [20] show that using the ELU activation function is better than ReLU or other functions, because ELU can output negative values. When calculating word weight, the interaction between $\boldsymbol{Q}$ and $\boldsymbol{K}$ becomes more complex, and more complex semantic information can be extracted. It can be seen from the above operations that $\boldsymbol{Q}$, $\boldsymbol{K}$ and $\boldsymbol{V}$ are homogeneous. We only consider the interaction between $\boldsymbol{Q}$ and $\boldsymbol{K}$, and weighted $\boldsymbol{V}$ to achieve the purpose of self-attention. We give a more detailed introduction in the next section.

C. Multi-Head Self-Attention

We use self-attention to calculate the semantic correlation between each window feature and all other window features. Reference [19] introduces that multi-head attention can expand self-attention's attention ability, as shown in Fig. 3. Multi-head attention uses h parallel scaled $\sqrt{d}$ fold dot products self-attention, and divides $\boldsymbol{X}$ into $\boldsymbol{X}_l \in \mathbb{R}^{n \times (d/h)}$ by dimension, where l is the index of heads. The network pays attention to the feature of different parts, and cascades them at last

$$\boldsymbol{Z}_{li} = \sum_{i=1}^{n} \boldsymbol{\alpha}_i \boldsymbol{V}_{li} \tag{4}$$

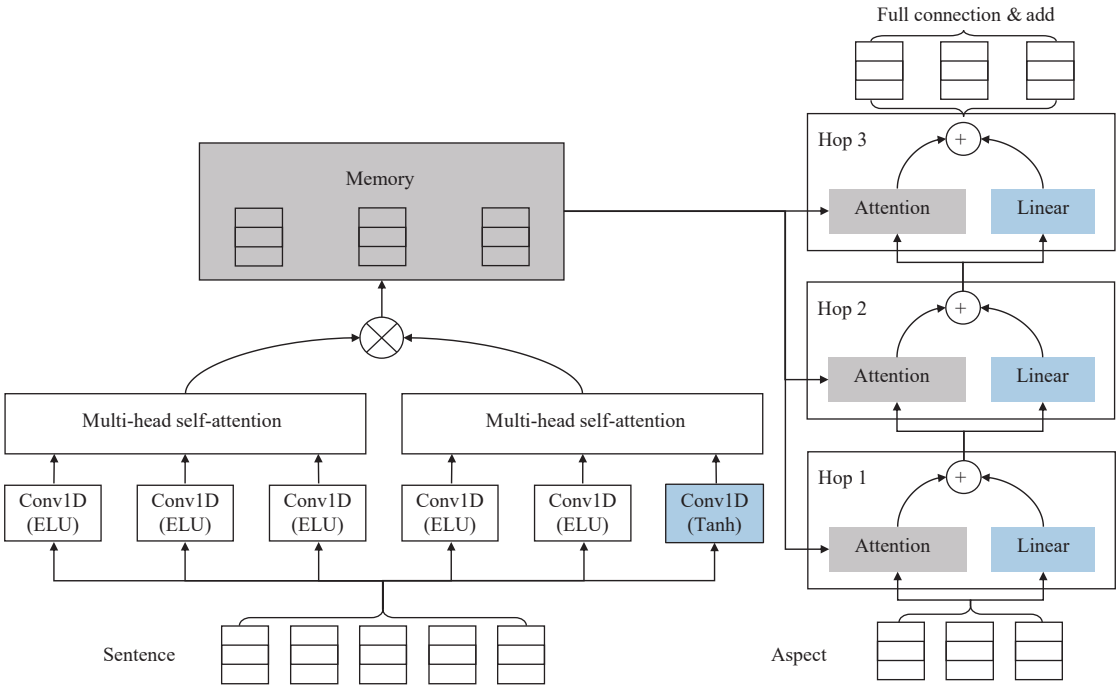


Fig. 1. The overall structure of the convolutional multi-head self-attention memory network (6 convolution processes and 2 multi-head self-attention processes use different parameters. Hops sharing parameters).

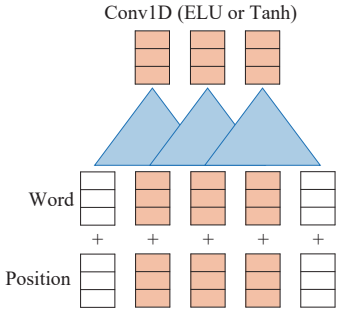


Fig. 2. The convolution process on embedding. ELU or Tanh activation functions can be used. The embedding of white is the zero vector.

$$\boldsymbol{\alpha}_i = \frac{\exp(\frac{\boldsymbol{Q}_{li}\boldsymbol{K}_{li}^T}{\sqrt{d}})}{\sum_i \exp(\frac{\boldsymbol{Q}_{li}\boldsymbol{K}_{li}^T}{\sqrt{d}})}. \tag{5}$$

Finally, we get the result $\boldsymbol{Z} \in \mathbb{R}^{n\times d}$ of the multi-head self-attention mechanism.

Reference [20] also raises the point that attention mechanisms focus on important contextual information only by weighted average, rather than capturing complex interactions. If we can use two convolutional multi-head self-attention and multiply the output, as shown in Fig. 1, we can make the complex interaction between words in the network capture sequences. Reference [9] also uses two convolutions with different activation functions and multiplies their outputs to implement the gating mechanism. The difference is that they want to use one of the convolution results to select the other, while we want to capture the deeper semantic interaction of the sequence.

We calculate two sets of feature matrices $\boldsymbol{Q}^1, \boldsymbol{K}^1, \boldsymbol{V}^1$, $\boldsymbol{Q}^2, \boldsymbol{K}^2, \boldsymbol{V}^2$ in the same way as (3), and in the second group, $\boldsymbol{V}^2$ uses Tanh as an activation function instead of ELU. The final results are obtained by multiplying the two sets of outputs. The output value of Tanh is between -1 and 1. Applying Tanh to the $\boldsymbol{V}$ of the second convolutional multi-head self-attention can prevent the output from getting too large or too small after they are multiplied. Let (4) and (5) be a function *Multi-HeadAtt* $(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V})$, then

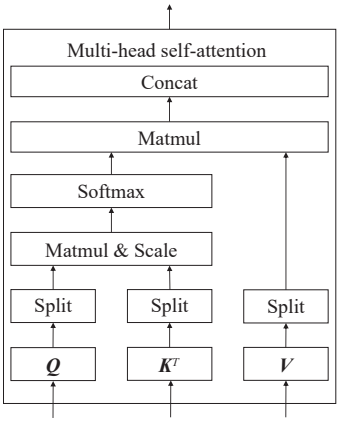


Fig. 3. Multi-head self-attention's calculation process.

$$\boldsymbol{Z}^u = MultiHeadAtt(\boldsymbol{Q}^u, \boldsymbol{K}^u, \boldsymbol{V}^u), \quad u \in \{1, 2\} \tag{6}$$

$$\boldsymbol{M} = \boldsymbol{Z}^1 \otimes \boldsymbol{Z}^2. \tag{7}$$

The final output is memory $\boldsymbol{M} \in \mathbb{R}^{n\times d}$, which stores semantic information.

D. Memory Network

Reference [14] introduces the memory network from question-answer tasks to ABSA tasks, which can use attention mechanisms to complete aspect sentiment analysis on context

memory. On the basis of semantic memory $\boldsymbol{M}$ extracted in Sections II-A, II-B, and II-C, we complete the acquisition of sentiment information related to aspect. Here, [14] maps all aspect words to embedding and averages them as whole aspect vectors, which lose a lot of aspect information. The difference is that we query each word of aspect terms by content attention, and then get multiple results through the softmax layer of shared parameters. Finally, we add up the predicted scores. In this way, when the model calculates the semantic correlation between aspect and memory, the required information will not be lost. The specific process is as follows:

$$\boldsymbol{vec}_j^{\text{att}} = \sum_{i=1}^{n} \beta_{ji} \boldsymbol{m}_i \tag{8}$$

$$\beta_{ji} = \frac{\exp(\tanh(\boldsymbol{W}_{\text{att}}[\boldsymbol{m}_i; \boldsymbol{E}_j^a] + b_{\text{att}}))}{\sum_{k}^{n} \exp(\tanh(\boldsymbol{W}_{\text{att}}[\boldsymbol{m}_k; \boldsymbol{E}_j^a] + b_{\text{att}}))} \tag{9}$$

where $j = 1, 2, \ldots, m$, $\boldsymbol{m}_i \in \mathbb{R}^{1\times d}$ is a slice of $\boldsymbol{M}$, $\boldsymbol{E}_j^a$ is the jth word vector of aspect embedding E^a. $\boldsymbol{W}_{\text{att}} \in \mathbb{R}^{1\times 2d}$ and $b_{\text{att}} \in \mathbb{R}$ are trainable parameters. This attention method can adaptively assign an importance score to each memory $\boldsymbol{m}_i$ according to its semantic relevance with aspect, and it is easy to train with other components in the end-to-end way. In a hop cell

$$\boldsymbol{vec}_j^{\text{lin}} = \boldsymbol{W}_{\text{lin}} \boldsymbol{E}_j^a + \boldsymbol{b}_{\text{lin}} \tag{10}$$

$$\boldsymbol{vec}_j = \boldsymbol{vec}_j^{\text{att}} + \boldsymbol{vec}_j^{\text{lin}} \tag{11}$$

where $\boldsymbol{W}_{\text{lin}} \in \mathbb{R}^{d\times d}$ and $\boldsymbol{b}_{\text{lin}} \in \mathbb{R}^{1\times d}$ are trainable parameters. After linear transformation of aspect embedding, we add its result to the result of attention. The parameters of multiple hops are shared. Finally, the model obtains the probability distribution through a softmax function

$$\hat{\boldsymbol{y}} = softmax(\sum_{j=1}^{m} (\boldsymbol{W}_{\text{full}} \boldsymbol{vec}_j + \boldsymbol{b}_{\text{full}})) \tag{12}$$

where $\boldsymbol{W}_{\text{full}} \in \mathbb{R}^{C\times d}$ and $\boldsymbol{b}_{\text{full}} \in \mathbb{R}^{1\times C}$ are trainable parameters. We use cross-entropy between estimated probability distribution $\hat{\boldsymbol{y}}$ and real probability distribution with L2 regulations.

III. Experiments

A. Datasets

Our experiments used four open datasets, two for aspect-category sentiment analysis (ACSA) tasks and two for aspect-term sentiment analysis (ATSA) tasks. Table I shows the statistics of datasets, where Res-ACSA, Res-ATSA and Lap-ATSA are customer comments on restaurants and laptops provided by SemEval-2014 Task 4[1] [3], and Tweet-ACSA is tweets provided by SemEval-2016 Task 6[2] [21].

The Res-ACSA dataset contains customer evaluations of five aspects of categories, namely "misc", "food", "service", "price" and "ambience". Res-ATSA is same as Res-ACSA, but each sentence contains the customer's evaluation of the specific terms. Lap-ATSA is the evaluation of specific terms of the laptops by customers. Some existing work [9] on three datasets in SemEval-2014 removed "conflict" labels. Tweet-ACSA is the user's sentiment expression on five topics of "feminist movement", "hillary clinton", "climate change is a real concern", "legalization of abortion" and "atheis". We divide the sentiment of the four datasets into three categories: "positive", "negative" and "neutral".

TABLE I
STATISTICS OF THE DATASETS

Dataset	Train			Test		
	Pos	Neg	Neu	Pos	Neg	Neu
Res-ACSA	2179	839	500	657	222	94
Tweet-ACSA	731	1342	741	304	715	230
Res-ATSA	2164	805	633	728	196	196
Lap-ATSA	987	866	460	341	128	169

B. Experimental Setting

In our experiments, we use 300-dimension word embedding vectors pre-trained by GloVe[3] [22] which is trained from web data where the vocabulary size is 1.9 m[2]. Word embedding vectors are not fine-tuned during training. Position embedding vectors are randomly initialized. The number of convolution filters is 300. We set the learning rate as 7×10^{-5} and L2 regularization coefficient as 1×10^{-5}. We set dropout to be 0.2. We will discuss the window size and hops in detail later. In order to learn semantic information from easy to difficult and reduce the padding to 0, we sort the training data by sentence length, and let the network learn short sentences before long ones. The batch size is 20 instances and the maximal epochs is 40. We randomly sampled 0.2 training data as dev set, and saved the best performance model parameters on the dev set, then calculated evaluation on the test set.

C. Baselines

In experiments, we compare our proposed model with the following models:

1) Feature + SVM: Feature-based SVM shows good performance on aspect sentiment classification. The system uses n-gram, parse and lexicon features [12].

2) LSTM: A standard LSTM [23] encodes a sentence from the starting to the final word, and the average value of all the hidden states is regarded as the final representation. For different aspects in a sentence, the model will give the same sentiment polarity.

3) TD-LSTM: It uses two LSTMs start from left and right to term words respectively [13]. Then it takes the hidden states of LSTM at the last time step to represent the features for prediction.

4) ATAE-LSTM: An aspect sentiment classification method using attention-based LSTM [7]. The model concatenates aspect embedding and the embedding of each word and feeds

[1] http://alt.qcri.org/semeval2014/task4/

[2] http://alt.qcri.org/semeval2016/task6/

[3] http://nlp.stanford.edu/projects/glove/

them to LSTM, and then passes through an attention layer.

5) IAN: Interactive attention network (IAN) [24] uses two LSTM on aspect embedding and word embedding, and regards the result of average-pooling as the query vector of other party's attention.

6) MemNet: This applies attention multiple times on word embedding, and feeds the last attention's output to softmax for prediction [14].

7) GCAE: Gated convolutional network [9] is an efficient model based on CNN. It uses two convolution with different activation functions on embedding, and uses the result of convolution to structure Gated Tanh-Relu Units.

D. Main Result

For model comparability, we evaluate our model's accuracy [9], [14], [24] and macro-averaged F-score. CMA-MemNet achieves the best performance compared with baselines on 4 datasets. Conv-MemNet only uses convolutions while MA-MemNet only uses multi-head self-attention on embedding. The result of ATSA task is shown in Table II, and ACSA is shown in Table III.

TABLE II
EXPERIMENTAL RESULTS FOR ATSA. THE MODELS WITH "1" ARE PROVIDED BY [18], "2" ARE PROVIDED BY [15], "3" ARE PROVIDED BY [9], "4" ARE PROVIDED BY [25]

Model	Res-ATSA		Lap-ATSA	
	Accuracy	F-score	Accuracy	F-score
Feature + SVM_2	80.16	NA	70.49	NA
$LSTM_1$	74.28	62.21	66.46	61.72
TD-$LSTM_1$	75.63	64.16	68.18	62.28
ATAE-$LSTM_1$	77.23	64.95	68.65	62.45
IAN_4	78.60	NA	72.10	NA
$MemNet_2$	78.16	65.83	70.33	64.09
$GCAE_3$	77.28	NA	69.14	NA
Conv-MemNet	80.85	67.92	72.73	68.14
MA-MemNet	81.02	67.75	72.83	67.82
CMA-MemNet	**81.26**	**68.64**	**73.24**	**68.94**

Note: "NA" indicates F-score was not calculated.

TABLE III
EXPERIMENTAL RESULTS FOR ACSA WITHOUT TD-LSTM AND IAN. THE MEANING OF MARKUP IS THE SAME AS IN TABLE II

Model	Res-ACSA		Tweet-ACSA	
	Accuracy	F-score	Accuracy	F-score
Feature + SVM_3	82.93	NA	–	–
$LSTM_1$	82.01	70.20	66.86	54.32
ATAE-$LSTM_1$	83.98	71.76	69.58	56.72
$MemNet_1$	84.28	72.38	70.14	58.62
$GCAE_3$	79.35	NA	–	–
Conv-MemNet	84.67	72.58	70.39	59.09
MA-MemNet	85.42	72.73	71.22	60.12
CMA-MemNet	**86.11**	**73.87**	**72.16**	**60.35**

Note: "—" indicates no experiment on tweet-ACSA.

As can be seen from Tables II and III, SVM provides a relatively strong machine learning baseline, which has outstanding performance in ABSA tasks. However, its performance depends strongly on feature engineering and effective vocabulary, and its effect is not as good as those of neural networks when there are not enough features. LSTM networks have more advantages than most networks in sequence modeling, and do not need to manually extract features to generate effective feature representation. Among all LSTM based methods, standard LSTM is the worst, mainly because it ignores aspect information. ATAE-LSTM pays close attention to the expression of sentiment in the sequence of aspect, and has made a significant improvement, especially in the Res-ATSA dataset, where the accuracy has been improved by 2.95%. IAN is the best LSTM based method for ATSA tasks, mainly because it utilizes the strong sequence modeling ability of LSTM, and combines the information of aspect influencing sequence and sequence influencing aspect. It is 5.64% more accurate than LSTM on Lap-ATSA dataset.

MemNet is an excellent network for ACSA tasks. It wins all baselines on Res-ACSA and Tweet-ACSA datasets, and its accuracy on Res-ATSA dataset is only 0.44% lower than that on IAN. Compared with MemNet, Conv-MemNet collects context information and MA-MemNet collects the semantic relevance of sequence itself, which is improved. This proves that this part of semantic information is effective in improving performance. We can draw the conclusion that MemNet has a strong aspect-sequence modeling capability, but lacks context information and sequence information, which limits its performance. CMA-MemNet can also combine this information well, while retaining the original information.

E. Effects of Window Size and Hops

As shown in Table IV, we take the Lap-ATSA dataset as an example to illustrate the effect of convolution window size and the number of memory network hops on the performance of the model. Window size affects the length of the context semantic information extracted from the network. The number of hops is the layer amount of aspect attention, which affects the abstraction of semantic information captured by the network. Experimental results show that the impact of window size and the number of hops on network performance is not a monotonous trend. The value of optimum performance on different datasets is often different.

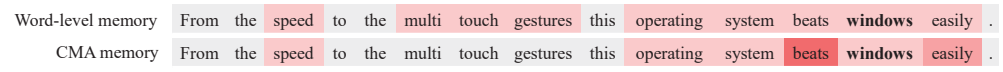
TABLE IV
EFFECT OF CONVOLUTION'S WINDOW SIZE AND NUMBER OF HOPS ON NETWORK ACCURACY FOR LAP-ATSA

Window	hops = 1	hops = 2	hops = 3	hops = 4	hops = 5
1	68.57	69.03	69.94	71.63	67.96
2	70.25	69.33	71.17	69.94	68.87
3	72.23	**73.24**	71.94	70.86	69.94
4	72.08	71.17	69.33	69.48	68.87
5	71.78	69.03	70.40	69.33	68.87

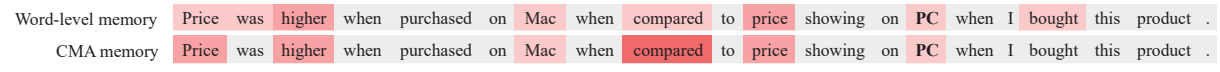
We find that the accuracy rate is the highest on the Lap-ATSA datasets when the window size is 3 and the number of hops is 2. For when more than 1 hop is needed, [14] explains

TABLE V
CASES IN THE LAP-ATSA DATASET. THE BOLD IS ASPECT AND THE SUBSCRIPT IS LABEL

Category	Example
Implicit expression (Case 1)	Within a few hours I was using the [**gestures**]$_{pos}$ unconsciously.
Multipoint dispersion (Case 2)	From the [**speed**]$_{pos}$ to the multi touch gestures this [**operating system**]$_{pos}$ beats [**windows**]$_{neg}$ easily.
Context expression (Case 3)	Price was higher when purchased on [**Mac**]$_{neg}$ when compared to price showing on [**PC**]$_{pos}$ when I bought this product.



(a) Attention score display of Case 2. The target is "windows"



(b) Attention score display of Case 3. The target is "PC"

Fig. 4. Comparison of attention on word-level memory and CMA memory. Attention score by (9) is used as color-coding. Deep color means a high attention score.

that it is necessary to extract deeper semantic information. In the experiment of MemNet, when hops is 7, the model works best. Our network is not based on word embedding. The model has extracted deep semantic information through convolutional multi-head self-attention, so fewer hops are needed. When window size is 1, it is equivalent to paying attention to word level information. When window size is too large, the network is easily affected by some non-related information noise in the same window.

We use the same method to get the best value of the model on other datasets. On Res-ATSA dataset, window size is 3 and the number of hops is 5. On Res-ACSA dataset, window size is 2 and the number of hops is 2. On Tweet-ACSA dataset, window size is 2 and the number of hops is 2.

F. Case Study

In this section, we analyze some cases in the Lap-ATSA dataset, as shown in Table V and Fig. 4, to illustrate the effectiveness of the mechanism.

There are three types of examples that most methods find difficult to identify. The first is implicit sentiment expression. In Case 1, it uses "gestures" unconsciously to explain it likes them. However, there are no obvious sentiment words, and our system can recognize such examples completely correctly. This is another important research direction in SA. The second is the complex expression of important information. The aspect-sequence attention in MemNet can capture useful information for aspect, but often not accurately enough, and does not recognize all aspects correctly such as in Case 2. As shown in Fig. 4(a), "beats windows easily" in "speed", shows a negative the polarity for "windows". But it is hard for word-level mechanisms to capture information such as "A beats B". Convolution can combine some related and important features, and self-attention pays attention to the semantics of the sequence itself, while networks can better understand the relationship between important words. The third is context expression such as negation, comparison and condition. As with the comparative expression in Case 3, this is a difficult problem for word-level mechanisms. As shown in Fig. 4(b), if a model lacks a sequence semantic, it may only see "price" and "higher" in the sentence when analyzing "PC". The model is likely to give negative judgment to both "PC" and "Mac". Convolution and self-attention can better understand this kind of contextual information, and enable the model to focus on the word "compared".

IV. CONCLUSION

In this paper, we propose a highly parallel convolutional multi-head self-attention based memory network. Compared with an embedding based memory network, CMA-MemNet can capture complex semantic information of the context better and give more attention to the semantic relations between the words in the sequence itself. We show the performance of the model on four datasets for ATSA and ACSA tasks and prove its effectiveness. In the future, we would like to consider more types of memory modules in semantic information representation, and synthetically analyze their aspects according to the scores outputted by different memory modules.

REFERENCES

[1] Liu B and Zhang L, *A Survey of Opinion Mining and Sentiment Analysis*. Boston, MA: Springer US, 2012, pp. 415–463.

[2] Pang B and Lee L, "Opinion mining and sentiment analysis," *Foundations and Trends. in Information Retrieval*, vol. 2, no. 1–2, pp. 1–135, 2008.

[3] 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 on Semantic Evaluation*, *SemEval@COLING*, Dublin, Ireland, Aug. 2014, pp. 27–35.

[4] M. Pontiki, D. Galanis, H. Papageorgiou, S. Manandhar, and I. Androutsopoulos, "Semeval-2015 task 12: Aspect based sentiment analysis," in *Proc. 9th Int. Workshop on Semantic Evaluation, SemEval@NAACL-HLT*, Denver, Colorado, USA, Jun. 2015, pp. 486–495.

[5] M. Pontiki, D. Galanis, H. Papageorgiou, I. Androutsopoulos, S. Manandhar, M. Al-Smadi, M. Al-Ayyoub, Y. Y. Zhao, B. Qin, O. D. Clercq, V. Hoste, M. Apidianaki, X. Tannier, N. V. Loukachevitch, E. V. Kotelnikov, N. Bel, S. M. J. Zafra, and G. C. Eryigit, "Semeval-2016 task 5: Aspect based sentiment analysis," in *Proc. 10th Int. Workshop on Semantic Evaluation*, *SemEval@NAACLHLT*, San Diego, CA, USA, Jun. 2016, pp. 19–30.

[6] L. Shu, H. Xu, and B. Liu, "Lifelong learning CRF for supervised

aspect extraction," in *Proc. 55th Annual Meeting of the Association for Computational Linguistics*, *ACL*, Vancouver, Canada: vol. 2, 2017, pp. 148–154.

[7] Y. Q. Wang, M. Huang, X. Y. Zhu, and L. Zhao, "Attention-based LSTM for aspect-level sentiment classification," in *Proc. Conf. Empirical Methods in Natural Language Processing*, *EMNLP*, Austin, Texas, USA, Nov. 2016, pp. 606–615.

[8] H. H. Do, P. W. C. Prasad, A. Maag, and A. Alsadoon, "Deep learning for aspect-based sentiment analysis: A comparative review," *Expert Syst. Appl.*, vol. 118, pp. 272–299, 2019.

[9] W. Xue and T. Li, "Aspect based sentiment analysis with gated convolutional networks," in *Proc. 56th Annual Meeting of the Association for Computational Linguistics*, *ACL*, Melbourne, Australia: Association for Computational Linguistics, Jul. 2018, pp. 2514–2523.

[10] X. W. Ding, B. Liu, and P. S. Yu, "A holistic lexicon-based approach to opinion mining," in *Proc. Int. Conf. Web Search and Web Data Mining*, *WSDM*, Palo Alto, California, USA, Feb. 2008, pp. 231–240.

[11] W. X. Zhao, J. Jiang, H. F. Yan, and X. M. Li, "Jointly modeling aspects and opinions with a maxent-lda hybrid," in *Proc. Conf. Empirical Methods in Natural Language Processing*, *EMNLP*, MIT Stata Center, Massachusetts, USA, 2010, pp. 56–65.

[12] S. Kiritchenko, X. D. Zhu, C. Cherry, and S. Mohammad, "NRC-Canada-2014: Detecting aspects and sentiment in customer reviews," in *Proc. 8th Int. Workshop on Semantic Evaluation*, *SemEval@COLING*, Dublin, Ireland, Aug. 2014, pp. 437–442.

[13] D. Y. Tang, B. Qin, X. C. Feng, and T. Liu, "Target-dependent sentiment classification with long short term memory," *CoRR*, vol. abs/1512.01100, 2015.

[14] D. Y. Tang, B. Qin, and T. Liu, "Aspect level sentiment classification with deep memory network," in *Proc. Conf. Empirical Methods in Natural Language Processing*, *EMNLP*, Austin, Texas, USA, Nov. 2016, pp. 214–224.

[15] P. Chen, Z. Q. Sun, L. D. Bing, and W. Yang, "Recurrent attention network on memory for aspect sentiment analysis," in *Proc. Conf. Empirical Methods in Natural Language Processing*, *EMNLP*, Copenhagen, Denmark, Sep. 2017, pp. 452–461.

[16] B. T. Do, "Aspect-based sentiment analysis using bitmask bidirectional long short term memory networks," in *Proc. 31st Int. Florida Artificial Intelligence Research Society Conf.*, *FLAIRS*, Melbourne, Florida, USA. May 2018, pp. 259–264.

[17] N. Majumder, S. Poria, A. F. Gelbukh, M. S. Akhtar, E. Cambria, and A. Ekbal, "IARM: inter-aspect relation modeling with memory networks in aspect-based sentiment analysis," in *Proc. Conf. Empirical Methods in Natural Language Processing*, Brussels, Belgium, Oct. – Nov. 2018, pp. 3402–3411.

[18] P. S. Zhu and T. Y. Qian, "Enhanced aspect level sentiment classification with auxiliary memory," in *Proc. 27th Int. Conf. Computational Linguistics*, *COLING*, Santa Fe, New Mexico, USA, Aug. 2018, pp. 1077–1087.

[19] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Advances in Neural Information Processing Systems 30: Annual Conf. Neural Information Processing Systems*, Long Beach, CA, USA, 2017, pp. 6000–6010.

[20] S. Gao, A. Ramanathan, and G. D. Tourassi, "Hierarchical convolutional attention networks for text classification," in *Proc. 3rd Workshop on Representation Learning for NLP*, *Rep4NLP@ACL*, Melbourne, Australia, Jul. 2018, pp. 11–23.

[21] S. Mohammad, S. Kiritchenko, P. Sobhani, X. D. Zhu, and C. Cherry, "Semeval-2016 task 6: Detecting stance in tweets," in *Proc. 10th Int. Workshop on Semantic Evaluation*, *SemEval@NAACL-HLT*, San Diego, CA, USA, Jun. 2016, pp. 31–41.

[22] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proc. Conf. Empirical Methods in Natural Language Processing*, *EMNLP*, Doha, Qatar, 2014, pp. 1532–1543.

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

[24] D. H. Ma, S. J. Li, X. D. Zhang, and H. F. Wang, "Interactive attention networks for aspect-level sentiment classification," in *Proc. 26th Int. Joint Conf. Artificial Intelligence*, *IJCAI*, Melbourne, Australia, Aug. 2017, pp. 4068–4074.

[25] X. Y. Wang, G. L. Xu, J. Y. Zhang, X. W. Sun, L. Wang, and T. L. Huang, "Syntaxdirected hybrid attention network for aspect-level sentiment analysis," *IEEE Access*, vol. 7, pp. 5014–5025, 2019.

Yaojie Zhang received the bachelor degree in computer science and technology from Harbin Institute of Technology in 2018. He is studying for M.Sc. degree at Harbin Institute of Technology. His research interests include sentiment analysis in natural language processing, including fine-grained sentiment analysis, opinion target and opinion word extraction, and multitask learning.

Bing Xu received the Ph.D. degree in computer science and technology from Harbin Institute of Technology in 2012. She is currently an Associate Professor with the School of Computer Science and Technology, Harbin Institute of Technology. Her research interests include natural language processing and sentiment analysis.

Tiejun Zhao is a Professor of the School of Computer Science and Technology, Harbin Institute of Technology. He is the Director of HIT-MSRA Natural Language and Speech Joint Laboratory. His research interests include natural language understanding, and applied artificial intelligence. As project PI, he accomplished 5 projects from national science foundation and national high-tech plan. He published 3 academic books and over 30 papers on journals and conferences in recent 5 years. Dr. Zhao has been a PC member on ACL, COLING in current 5 years and was also assigned a Co-chair on NLPCC 2017.