Knowledge-Based Systems 227 (2021) 107196

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

A novel network with multiple attention mechanisms for aspect-level sentiment analysis

Xiaodi Wang, Mingwei Tang *, Tian Yang, Zhen Wang

School of Computer and Software Engineering, Xihua University, Chengdu 610039, China

ARTICLE INFO

Article history:
Received 8 August 2020
Received in revised form 1 June 2021
Accepted 2 June 2021
Available online 10 June 2021

Keywords:
Aspect-level sentiment analysis
Attention mechanism
Pre-trained BERT
Natural language processing

ABSTRACT

Aspect-level sentiment analysis aims at identifying the sentiment polarity of specific aspect words in a given sentence. Existing studies mostly use recurrent neural network (RNN) -based models. However, truncated backpropagation, gradient vanishing, and exploration problems often occur during the training process. To address these issues, this paper proposed a novel network with multiple attention mechanisms for aspect-level sentiment analysis. First, we apply the bidirectional encoder representations from transformers (BERT) model to construct word embedding vectors. Second, multiple attention mechanisms, including intra- and inter-level attention mechanisms, are used to generate hidden state representations of a sentence. In the intra-level attention mechanism, multi-head self-attention and point-wise feed-forward structures are designed. In the inter-level attention mechanism, global attention is used to capture the interactive information between context and aspect words. Furthermore, a feature focus attention mechanism is proposed to enhance sentiment identification. Finally, several classic aspect-level sentiment analysis datasets are used to evaluate the performance of our model. Experiments demonstrate that the proposed model can achieve state-of-the-art results compared to baseline models.

1. Introduction

With the development of social networks, more and more customer opinions are represented online. However, due to the complicated structure of those opinions, the performance of traditional sentiment analysis cannot meet our expectations. Therefore, a new task named aspect-level sentiment analysis is proposed in academia and industry [1,2]. It aims to identify the sentiment polarity of specific aspect words in a given sentence or paragraph. Unlike traditional coarse-grained sentiment analysis tasks, such as sentence-level and document-level sentiment analysis, aspect-level sentiment analysis belongs to the fine-grained task of natural language processing (NLP) [3,4]. It fully uses the relationship between context and aspect words, and then judges the sentiment polarity of a given aspect term. An example is shown in Fig. 1. There are three types of aspect words in a sentence: those with positive, negative, and neutral sentiment polarity. There are many research topics in aspect-level sentiment analysis, including recognition [5–8], classification [9,10], and regression [11]. In this paper, we mainly focus on classification issues.

In the early stage of sentiment analysis, statistical methods have been used and obtained good results. For example, support vector machines (SVMs) [12], the bag-of-words model, and sentiment lexicons [13] are classic methods and widely used in this field. However, the semantic relationships between context and aspect words are not considered, and they excessively rely on handcrafted features in classification tasks. Therefore, it is easily suffering from a bottleneck during the training of statistics-based models. To address the above problems, a majority of researchers have devoted themselves to deep learning.

In recent years, scientists have examined the effects of deep learning compared with traditional manual generation methods in NLP tasks [14–16]. Neural network-based models such as the recurrent neural network (RNN) [17], long short-term memory (LSTM) [18], and gate recurrent units (GRU) [19] are used in aspect-level sentiment analysis [20]. They model context and aspect vectors in parallel and learn useful low-dimension representations from context and aspect words automatically. However, they are time-consuming, and parallel operation is inconvenient to implement. Even worse, gradient disappearance or explosion adds some complex issues during model training. Therefore, a gated convolutional neural network (CNN) that boasts a special gate mechanism was proposed to solve the parallel operation issue [21]. CNN-based models enhance the capability of sentiment analysis, and reduce time costs. However, they are not good at dealing with long-distance relations in most situations.

Fortunately, the attention mechanism [22,23], which is widely applied in machine translation [24], image recognition [25], and

* Corresponding author.
E-mail address: tang4415@126.com (M. Tang).

https://doi.org/10.1016/j.knosys.2021.107196
0950-7051/

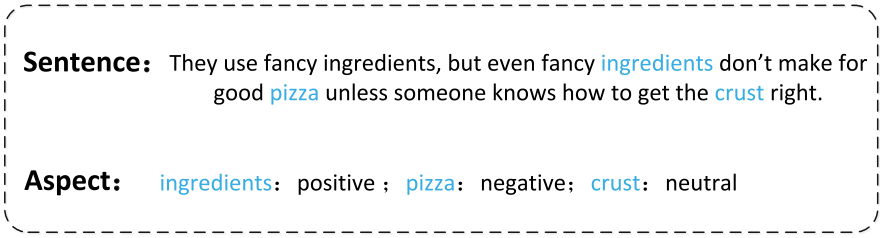


Fig. 1. Example of aspect-level sentiment analysis.

reading comprehension [26,27], can make up for shortcomings that appear in the neural network-based model. It is incorporated to force the model to pay more attention to context words, which have richer semantic relations compared to aspect words. By modeling context and aspect representations interactively, the attention mechanism obtains several weighted scores, and generates new representations of the sentence [28–30]. However, due to the complicated structure of semantic relations in natural language, an attention mechanism cannot learn completely informative features of a sentence. Generally, the single attention mechanism treats an aspect term as a whole while in fact, an aspect term in a sentence is often a phrase. Therefore, each word in the phrase should be given different attention weights. Considering the sentence, "All the money went into the interior decoration, none of it went to the chefs", the aspect term "interior decoration" contains two words. It is obvious that the importance of the former word is less than that of the latter, and thus the word "decoration" should be given a larger weighted score than "interior". Therefore, an effective attention focus structure is vital for fine-grained sentiment analysis tasks.

In conclusion, there are two problems with existing methods. First, machine learning-based or neural network-based methods do not obtain the hidden states of a sentence effectively. Second, the single attention mechanism cannot provide a reasonable weighted score for the model's sentiment judgment.

In this paper, we propose a novel multiple attention mechanism network (MAMN) for aspect-level sentiment analysis. First, we apply the pre-trained BERT [31] to construct word embedding vectors. Second, multiple attention mechanisms, including intra- and inter-level attentions, are used to generate hidden state representations of a sentence. In the intra-level attention mechanism, multi-head self-attention and point-wise feed-forward [32] structures are introduced. In the inter-level attention mechanism, global attention is used to capture the interactive information between context and aspect words. Additionally, a feature focus attention (FFA) [33] is proposed to enhance sentiment identification. Finally, several classic aspect-level sentiment analysis datasets are used to evaluate the performance of our model. The main contributions of our study are summarized as follows:

1. We propose a novel model MAMN in which an intra-level attention mechanism including a multi-head self-attention (MHSA) and point-wise feed-forward (PWFF) structure are designed to generate the hidden state representations of the sentence. In addition, we use the pre-trained BERT to construct word embedding vectors.
2. The MAMN carries out aspect-level sentiment analysis by using multiple attention mechanisms. Moreover, a FFA mechanism is proposed to enhance sentiment identification.
3. We conduct experiments on five public datasets: the SemEval 2014 Task 4 dataset (restaurant and laptop reviews), SemEval 2015 Task 12 dataset (restaurant reviews), SemEval 2016 Task 5 dataset (restaurant reviews), and the Annual Meeting of the Association for Computational Linguistics (ACL) dataset (Twitter posts). Experiment results demonstrate that our proposed model achieves superior performance compared to baseline models.

The rest of this paper is organized as follows. In Section 2, we introduce related work. Section 3 presents a detailed description of our proposed model MAMN. Then, we compare our model with other aspect-level sentiment analysis models in Section 4. Conclusions and directions for future work are presented in Section 5.

2. Related work

In this section, we introduce related work in sentiment analysis. First, we introduce aspect-level sentiment analysis. Afterward, we present neural network methods used in aspect-level sentiment analysis. Finally, we give the development process of attention mechanisms in recent years.

2.1. Aspect-level sentiment analysis

Aspects can be defined as aspect terms and aspect categories [34–36], and we choose the former definition in this paper. Aspect-level sentiment analysis is a vital task in NLP, which aims to extract and identify the sentiment polarity in a sentence or paragraph with regard to specific aspect words [3,37–39]. Compared with document-level or sentence-level sentiment analysis tasks, aspect-level analysis considers implied fine-grained information between aspect and context words. Therefore it is a challenging task in sentiment analysis.

There are two methods that have been used in traditional aspect-level sentiment analysis over the past several years: sentiment dictionary-based methods, and machine learning-based methods. The dictionary-based methods judge the sentiment polarity of a sentence mainly according to the total number of sentiment words; the sentiments may be positive, negative, or neutral [40]. Rao et al. [41] proposed a model utilizing a graph-based semi-supervised learning framework for building sentiment lexicons in a variety of resource availability situations. Relations like synonymy and hypernymy were combined to improve label propagation results. Meanwhile, Federici et al. [42] applied the Stanford NLP Library[1] to identify the polarity of aspect words in a sentence. Then, a syntax tree that reacts to the semantic features was generated to conduct sentiment analysis. In contrast, the machine learning-based methods mainly focus on handcrafted feature engineering to perform sentiment analysis. Pang et al. [43] used SVMs to deal with text representation in sentiment classification tasks. Furthermore, Alsmadi et al. [44] proposed a supervised model using morphological, syntactic, and semantic features to process word embedding in Arabic hotel reviews. Experimental results showed that the SVM classifier gains

[1] The Stanford NLP Library is available at http://stanfordnlp.github.io/CoreNLP/index.html.

better results than other classifiers such as decision trees and the K-nearest neighbor. Jiang et al. [45] proposed a statistical method that combines features engineering to judge sentiment labels. However, the approaches mentioned above are not effective due to the handcrafted features, which result in poor semantic comprehension. Therefore, researchers are forced to focus on deep learning.

2.2. Neural networks for aspect-level sentiment analysis

Compared with traditional classifiers such as the SVM and K-nearest neighbor, neural networks show promising ability in modeling the semantic relations of words. Therefore, neural network-based methods are widely used in aspect-level sentiment analysis [46,47]. RNN and its several variants are commonly applied in all kinds of NLP tasks. Mikolov et al. [48] proposed an RNN model that not only considers the dependency of aspect and context words, but adds a time step into hidden states when extracting features. Tang et al. [49] designed two improved models, TD-LSTM and TC-LSTM, which divided sentences into three parts: the pre-context, aspect word, and post-context. Two LSTMs were used to extract features of the pre-context and post-context. Ma et al. [50] used a new semantic LSTM. This extended LSTM cell consists of a separate output gate that interpolates the token-level memory and the concept-level input. In addition, they created a hybrid of the LSTM and a recurrent additive network that simulates semantic patterns. This is a knowledge-rich solution to aspect-level sentiment analysis. Lastly, Wu et al. [51] proposed a model called MTKFN that fuses textual knowledge from multiple sources for aspect-based sentiment analysis. Structure and sentiment knowledge were extracted and combined. In addition, information on conjunctions were fused to capture the essential context words and judge the sentiment polarity of a specific aspect.

However, LSTM-based models require a significant amount of time during the training process due to the complicated gate mechanism. Therefore, other neural networks have been receiving increasing attention. Gu et al. [52] applied a position-aware bidirectional GRU network for aspect-level sentiment analysis. The GRU simplifies the gate mechanism in LSTM as a reset gate and update gate, thereby reducing the complexity of the neural network. Other effective networks, such as the memory network and CNN, have been used to extract features among words in a sentence. For example, Tang [53] introduced a deep memory network that explicitly captures the important features of each context word with regard to an aspect. The importance degree scores were calculated with multiple deep memory networks. Meanwhile, Xue et al. [54] proposed a model based on the CNN and gate units. A novel gate mechanism called the Gated Tanh ReLu Unit was used to generate sentiment features according to a specific aspect term.

2.3. Attention mechanisms for aspect-level sentiment analysis

The attention mechanism [55–58], which is widely used in neural network-based methods, improves the models' classification capability. The attention mechanism is similar to human visual behavior. When someone is reading a text, some specific words or phrases text will be focused on rather than the whole paragraph or document. Ma et al. [59] introduced an interactive attention neural network called IAN. The IAN model used interactive attention to obtain attention scores in contexts and aspects, and then generated the representations for aspects and contexts separately. Huang et al. [60] proposed an attention over attention (AOA) neural network to perform aspect-level sentiment analysis. They modeled aspect terms and context words in a joint way, and captured essential features between words in a sentence. Furthermore, the AOA network can automatically pay attention to the important parts of the sentence. Song et al. [61] introduced an attention encoder network that replaces the recurrence structure with attention-based encoders for modeling relations between contexts and aspect terms. Xu et al. [62] designed a novel model with multiple attention networks. They used a transformer encoder to construct word embedding vectors to reduce time consumption during the training process. In addition, the inter attention layer including global attention and local attention was used to generate the sentence representations. Lastly, Liu et al. [63] presented a novel end-to-end model termed the recurrent memory neural network (ReMemNN). The ReMemNN utilizes a multi-element attention mechanism to generate powerful attention weights and more precise aspect-dependent sentiment representations.

It can be concluded from the above studies that a reasonable feature modeling structure and attention mechanism can enhance a model's performance in aspect-level sentiment analysis. Therefore, in this paper, the proposed model is based on multiple attention mechanisms. Specifically, we utilize intra-level attention layers and an inter-level attention layer. In the former, we use a stack of t attention layers, including multi-head self-attention and a point-wise feed-forward network, to capture vital features between word embedding vectors. It seems like the structure of the transformer encoder, which contains multi-head self-attention, residual connection, and layer normalization [32]. In the inter-level attention layer, global attention, which influences the aspect term from context words, is employed to capture the interactive information. Moreover, a FFA mechanism is proposed to force the model to focus on those contextual words with close semantic relations toward a given aspect term.

Table 1
Notations of symbols used in MAMN.

Notation	Description
w_j	The jth word of a sentence
e_j^c	Embedding vector of the jth context word of a sentence
e_k^a	Embedding vector of the kth aspect word of a sentence
q_i	The ith vector in the query sequence
k_j	The jth vector in the key sequence
v_k	The kth vector in the value sequence
h_i^{cp}	The context hidden representation calculated by PWFF
h_k^{ap}	The aspect hidden representation calculated by PWFF
h^{cm}	Masked output representations of context words
h^{cw}	Weighted down output representations of context words

3. Model description

This section introduces the structure of our proposed model MAMN. The overall architecture of the MAMN is shown in Fig. 2. It is composed of four parts: a word embedding layer, intra-level attention layers, an inter-level attention layer, and an output layer.

3.1. Problem definition

Given a sentence with n context words and m aspect terms, we denote the contexts and aspects as $[w_1, w_2, \ldots, w_n]$ and $[w_i, w_{i+1}, \ldots, w_{i+m-1}]$, respectively. The aspect is a subset sequence of context words and may be a single word or a phrase. The purpose of our research is to identify the sentiment polarity of the context words over a given aspect term. The sentiment polarity is one of the elements in {positive, negative, and neutral}. Notations employed in our model are described in Table 1.

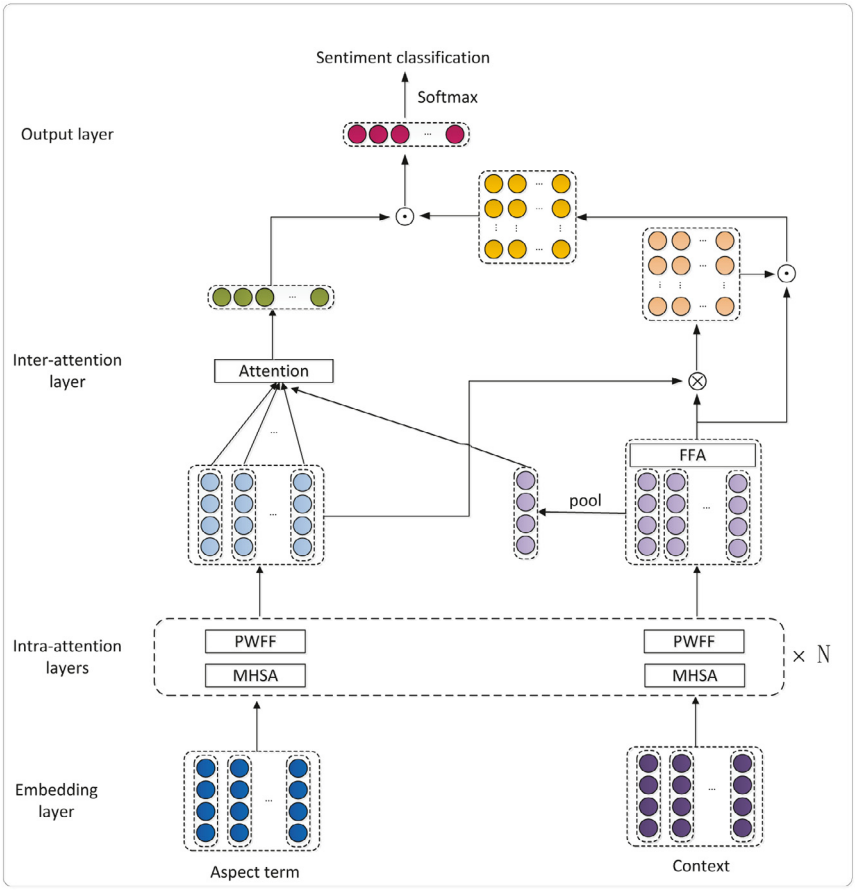


Fig. 2. Architecture of MAMN.

3.2. Word embedding layer

Word embedding maps each word w_i in a given sentence into a low-dimension real-value vector space. We use the pre-trained language model $\mathbf{BERT_{base}}$ [31] to obtain a fixed word embedding for each context word and aspect word. BERT has been widely applied in unsupervised predictive tasks in recent years. It boasts an excellent feature representation capability for word embedding due to pre-training on a massive corpus. The structure of BERT is shown in Fig. 3.

For the input of context and aspect words w_j $(1 \leqslant j \leqslant n)$ and w_k $(i \leqslant k \leqslant i+m-1)$, we obtain two sets of word embedding vectors $\mathbf{e_j^c} \in \mathbb{R}^{d_w}$ and $\mathbf{e_k^a} \in \mathbb{R}^{d_w}$ by using Eqs. (1) and (2), where d_w is the dimension of word embedding.

$$e_j^c = BERT(w_j) \tag{1}$$

$$e_k^a = BERT(w_k) \tag{2}$$

3.3. Intra-level attention layers

After obtaining the embedding vector of each word in Section 3.2, we use intra-level attention layers to generate hidden state representations. Inspired by the transformer encoder introduced by Google [32], these layers mainly consist of two components: multi-head self-attention and a point-wise feed-forward network.

3.3.1. Multi-head self-attention

Multi-head attention utilizes multiple attention mechanisms to obtain weighted attention scores for each word in a sentence.

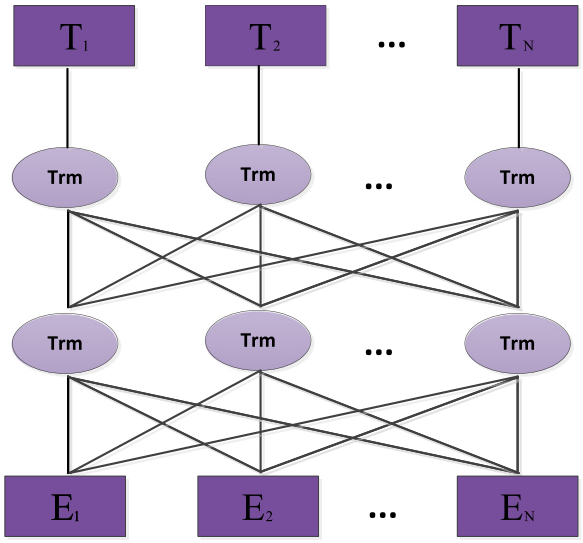


Fig. 3. Structure of BERT.

In this paper, the "head" in "multi-head attention" refers to a special attention mechanism that consists of multiple parallel scaled dot-product attention mechanisms. Therefore, before introducing multi-head attention, we first discuss the scaled dot-product attention.

An attention mechanism can be defined as a mapping function (MF). The input of MF is a query sequence $\mathbf{Q} = \{\mathbf{q_1}, \mathbf{q_2}, \ldots, \mathbf{q_n}\}$ and a set of key–value pairs, and the output of MF is a weighted

sum of the values. We denote key and value sequences as $\mathbf{K} = \{\mathbf{k_1}, \mathbf{k_2}, \ldots, \mathbf{k_n}\}$ and $\mathbf{V} = \{\mathbf{v_1}, \mathbf{v_2}, \ldots, \mathbf{v_n}\}$, respectively. The $\mathbf{q_i} \in \mathbb{R}^{d_h}$ in the query sequence, $\mathbf{k_i} \in \mathbb{R}^{d_h}$ in the key sequence, and $\mathbf{v_i} \in \mathbb{R}^{d_h}$ in the value sequence are all vectors, where d_h is the dimension of the hidden state. Then, the scaled dot-product attention can be defined based on the query $\mathbf{Q}$, the key $\mathbf{K}$, and the value $\mathbf{V}$ as follows:

$$Attention(Q, K, V) = softmax(\frac{f\left(Q, K^T\right)}{\sqrt{d}})V \tag{3}$$

where $\sqrt{d}$ is a scaling factor that aims to prevent the above inner product from becoming too large. A large inner product may affect the performance of the softmax function. The multi-head attention allows the model to learn semantic information from different representation subspaces. The results obtained from different heads are concatenated together, and the values calculated by linear transformations are presented as the output of multi-head attention:

$$head_i = Attention(W_i^Q Q, W_i^K K, W_i^V V) \tag{4}$$

$$MultiHead = Concate(head_1, head_2, \ldots, head_h) \cdot W^O \tag{5}$$

where $\mathbf{W_i^Q} \in \mathbb{R}^{d_q \times d_h}$, $\mathbf{W_i^K} \in \mathbb{R}^{d_k \times d_h}$ and $\mathbf{W_i^V} \in \mathbb{R}^{d_v \times d_h}$ are linear transformation parameter matrices. The dimensions d_q, d_k, d_v are all equal to $\frac{d_h}{h}$, and h refers to the total number of heads. The $\mathbf{W^O} \in \mathbb{R}^{hd_v \times d_h}$ is another linear transformation matrix.

Multi-head self-attention is a special subset of multi-head attention that sets query sequence $\mathbf{Q}$ equal to key sequence $\mathbf{K}$. When the embedding vectors $\mathbf{e_j^c}$ and $\mathbf{e_k^a}$ are put into the multi-head self-attention mechanism, the context and aspect representations $\mathbf{h_j^c} \in \mathbb{R}^{d_h}$ and $\mathbf{h_k^a} \in \mathbb{R}^{d_h}$ can be calculated by Eqs. (6) and (7), respectively.

$$h_j^c = MultiHead(e_j^c, e_j^c, e_j^c) \tag{6}$$

$$h_k^a = MultiHead(e_k^a, e_k^a, e_k^a) \tag{7}$$

3.3.2. Point-wise feed-forward network

A point-wise feed-forward (PWFF) network transforms the information of hidden states generated by MHSA. The PWFF is defined as

$$PWFF(h) = relu(h * W_p^1 + b_p^1) * W_p^2 + b_p^2 \tag{8}$$

where $*$ represents the convolution operator, and $relu(\cdot)$ stands for the activation function. $\mathbf{W_p^1} \in \mathbb{R}^{d_h \times d_h}$ and $\mathbf{W_p^2} \in \mathbb{R}^{d_h \times d_h}$ are weight matrices of the two convolutional kernels and are updated during the training process. $\mathbf{b_p^1} \in \mathbb{R}^{d_h}$ and $\mathbf{b_p^2} \in \mathbb{R}^{d_h}$ are biases of the two convolutional kernels. The kernel size for the convolution operation is 1 because we focus on the point-wise level of the hidden representation. The same transformation is utilized for each token belonging to the input. Then, we can obtain the output hidden representations $\mathbf{h_j^{cp}} \in \mathbb{R}^{d_h}$ and $\mathbf{h_k^{ap}} \in \mathbb{R}^{d_h}$ of a intra-level attention layer by

$$h_j^{cp} = PWFF(h_j^c) \tag{9}$$

$$h_k^{ap} = PWFF(h_k^a) \tag{10}$$

We remind the reader that it is impossible for a single intra-level attention layer to model the completely semantic features between words in a sentence. Several studies have proven that multiple attention layers can extract high-dimension abstract relations of hidden representations [62,64]. Therefore, we apply a stack of t layers to obtain more abstract features of context and aspect representations.

3.4. Inter-level attention layer

The inter-level attention mechanism is proposed to focus on the semantic relations between aspect and context words. Unlike intra-level attention, which computes weighted scores based on aspects or contexts themselves, inter-level attention mainly focuses on the semantic information between words. On top of that, a special mechanism named the FFA mechanism is introduced to enhance the model's capability to identify sentiment polarity.

3.4.1. Feature focus attention mechanism

The FFA mechanism aims to focus on the vital local context words that have close semantic relations with a given aspect. It has two working modes: the masked mechanism of context (MMC), and the weighted down mechanism of context (WDMC). The structure of the FFA mechanism is shown in Fig. 4.

First, the word position index of an aspect term in a sentence is marked as the cardinal point "0". Discrete spacing from the ith word in a sentence to the cardinal point is called the relative distance, and is denoted by d_i. It can be calculated by Formula (11).

$$d_i = \begin{cases} a_s - i, & i < a_s \\ 0, & a_s \leqslant i \leqslant a_e \\ i - a_e, & i > a_e \end{cases} \tag{11}$$

Here, i refers to the position index of the ith contextual word, a_s and a_e are the start position index and end position index, respectively. a_s equals a_e when the aspect term is a single word. Next, we introduce a new concept, the preserve window, which is shown in blue in Fig. 4. The size of the preserve window is equal to the total number of words before and after a given aspect. The size of the preserve window should consider sentence length and semantic information. If it is too large, the role of the MMC and WDMC will be weakened. In contrast, if it is too small, the semantic relation around an aspect term cannot be learned effectively. The hidden embedding values of contextual words in the preserve window will be completely retained, while the words outside the preserve window will be masked or weighted down. The algorithms of MMC and WDMC are presented in algorithms 1 and 2, respectively.

Algorithm 1 Masked mechanism of context

Input: The output representation $h^{cp} = [h_1^{cp}, h_2^{cp}, ..., h_n^{cp}]$ of context words calculated by MHSA and PWFF.
Output: Masked output representations h^{cm} of contextual words.

```
Initial h^cp = [h_1^cp, h_2^cp, ..., h_n^cp];
Calculate the relative distance d_i for each context word with regard to a given aspect;
for each relative distance d_i do
    if d_i ⩽ (preserve window size θ)/2 then
        r_i = e, where e refers to the ones vector;
    else
        r_i = 0, where 0 is the zeros vector;
    end if
end for
Calculate a mask matrix M = concate(r_1, r_2, ..., r_n);
Conduct a matrix element-wise product operation h^cm = h^cp · M;
return h^cm;
```

Finally, the new output hidden representations of context can be obtained by utilizing MMC or WDMC; these carry richer semantic relations and informative features than before.

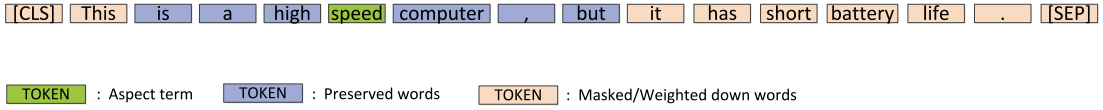


Fig. 4. Structure of FFA.

Algorithm 2 Weighted down mechanism of context

Input: The output representation $h^{cp} = [h_1^{cp}, h_2^{cp}, ..., h_n^{cp}]$ of context words calculated by MHSA and PWFF.
Output: Weighted down output representations h^{cw} of contextual words.

```
Initial h^cp = [h_1^cp, h_2^cp, ..., h_n^cp];
Calculate the relative distance d_i for each context word with regard to a given aspect;
for each relative distance d_i do
    if d_i ⩽ (preserve window size θ)/2 then
        r_i = e, where e refers to the ones vector;
    else
        r_i = (1 − (d_i − θ/2)/n) · e;
    end if
end for
Calculate a weight matrix W = concate(r_1, r_2, ..., r_n);
Conduct a matrix element-wise product operation h^cw = h^cp · W;
return h^cw;
```

3.4.2. Global attention mechanism

After obtaining the new hidden representations of context and aspect words, we assign weights to words using the global attention mechanism. We consider the contribution of contextual words corresponding to each word in a given aspect term. When the FFA is MMC, it can be calculated using Eqs. (12) to (14).

$$I_{kj} = \frac{\exp\left(f\left(h_j^{cm}, h_k^{ap}\right)\right)}{\sum_{i=1}^{n} \exp\left(f\left(h_i^{cm}, h_k^{ap}\right)\right)} \tag{12}$$

$$g_k = \sum_{j=1}^{n} I_{kj} \cdot h_j^{cm} \tag{13}$$

$$f\left(h_j^{cm}, h_k^{ap}\right) = tanh(h_j^{cm^T} \cdot W \cdot h_k^{ap} + b) \tag{14}$$

Here, $\mathbf{I_{kj}} \in \mathbb{R}^{m \times n}$ is the attention weight scores from the word vector $\mathbf{h_k^{ap}}$ to the jth context word vector. $\mathbf{g_k} \in \mathbb{R}^{m \times d_h}$ indicates the weighted sum of the hidden representations. $tanh(\cdot)$ is a nonlinear activation function. $\mathbf{W} \in \mathbb{R}^{d_h \times d_h}$ and $\mathbf{b} \in \mathbb{R}^{d_h}$ are the weight matrix and bias, respectively. When the FFA is WDMC, we replace $\mathbf{h_j^{cm}}$ in Eqs. (12) to (14) with $\mathbf{h_j^{cw}}$ and apply the same method as in MMC to obtain hidden representations.

We also consider that each word in a given aspect term may make different contributions to the identification of sentiment polarity. When the FFA is MMC, we calculate attention weights α_k for each aspect word as follows:

$$h_{avg}^{cm} = \frac{1}{n} \sum_{j=1}^{n} h_j^{cm} \tag{15}$$

$$\alpha_k = \frac{\exp\left(f\left(h_{avg}^{cm}, h_k^{ap}\right)\right)}{\sum_{i=1}^{m} \exp\left(f\left(h_{avg}^{cm}, h_i^{ap}\right)\right)} \tag{16}$$

$$f\left(h_{avg}^{cm}, h_k^{ap}\right) = tanh(h_{avg}^{cm\ T} \cdot W \cdot h_k^{ap} + b) \tag{17}$$

where $\mathbf{h_{avg}^{cm}} \in \mathbb{R}^{d_h}$ is the average pooled hidden representation of context words. $\alpha_k \in \mathbb{R}^m$ represents the attention weights from context words to each word in the aspect term. For WDMC, the model performs a similar operation and we only need to replace $\mathbf{h_j^{cm}}$ with $\mathbf{h_j^{cw}}$.

3.5. Output layer

After getting α_k and g_k from Section 3.4.2, the weighted sum hidden representations $\mathbf{O} \in \mathbb{R}^{d_h}$ can be obtained with Eq. (18).

$$O = \sum_{k=1}^{m} \alpha_k \cdot g_k \tag{18}$$

Then, a non-linear activation function is used to calculate the final sentence representation:

$$z = tanh(O \cdot W + b) \tag{19}$$

where $\mathbf{W} \in \mathbb{R}^{d_h \times d_c}$ and $\mathbf{b} \in \mathbb{R}^{d_c}$ are the weight matrix and bias, respectively. d_c refers to the number of classifications. Finally, we apply a softmax layer to calculate the sentiment polarity probability P_c of a specific aspect term.

$$P_c = \frac{\exp(z_c)}{\sum_{i=1}^{c} \exp(z_i)} \tag{20}$$

3.6. Model training

During the model's training, we apply a loss function to optimize parameters. The aim of the optimizer is to minimize the training loss as much as possible for all parameters. The loss function consists of two components, cross-entropy loss and $L2$ regulation, and is defined as follows:

$$Loss = -\sum_{i \in D} y_i \log\left(\hat{y}_i\right) + \lambda \|\Theta\|^2 \tag{21}$$

where D indicates the training dataset. y_i is the true sentiment polarity, and $\hat{y}_i$ is the predicted sentiment polarity for an aspect term. λ is a coefficient of $L2$ regulation, and Θ represents all used parameters. Furthermore, the dropout strategy [65] is used in the training process to avoid overfitting.

4. Experiments

4.1. Datasets

We conduct experiments on five publicly available datasets: SemEval 2014[2] [34], which consists of restaurant and laptop reviews; and SemEval 2015[3] [35] and SemEval 2016[4] [36], both in the restaurant domain. The last one is a collection of Twitter review comments gathered by Dong et al. [66]. Each piece of data in the datasets is a single sentence consisting of comments, aspect terms, and sentiment labels. However, several sentences may express a conflicting sentiment polarity, such as both positive and negative sentiments at the same time. Therefore, we remove the label "conflict" in case they affect the model's performance. The statistics of the datasets mentioned above are shown in Table 2.

[2] http://alt.qcri.org/semeval2014/task4/.
[3] http://alt.qcri.org/semeval2015/task12/.
[4] http://alt.qcri.org/semeval2016/task5/.

Table 2
Statistics of the employed datasets.

Dataset	Positive		Neutral		Negative		Total	
	Train	Test	Train	Test	Train	Test	Train	Test
Restaurant2014	2164	728	637	196	807	196	3608	1120
Laptop2014	994	341	464	169	870	128	2330	638
Restaurant2015	1178	439	50	35	382	328	1610	802
Restaurant2016	1620	597	88	38	709	190	2417	825
Twitter	1561	173	3127	346	1560	173	6248	692

4.2. Experimental settings

In our experiments, we use the pre-trained BERT to initialize the word embedding vectors. The dimensions of context word embedding and aspect word embedding are set to 768. All weight matrices are initialized by sampling from uniform distribution $U(-0.01, 0.01)$, and all biases are set to zero. The batch size is set at 16, and the max length of a sentence is set to 80. The preserve window size is 6. We use the Adam optimizer to optimize all parameters during the training process. The learning rate and the coefficient of $L2$ regulation are 2e-5 and 1e-5, respectively. Furthermore, the dropout rate is set to 0.5 to avoid overfitting. We apply accuracy and macro-f1 metrics to evaluate the performance of our model.

4.3. Model comparison

In this section, we compare the proposed model MAMN with several baseline models, including non-BERT-based models and BERT-based models. Many models have been proposed for the aspect-level sentiment analysis. Because our model mainly focuses on multiple attention mechanisms, we select only some of the published models that are related to our study as baseline models. For non-BERT-based baselines, the dimensions of word embedding vectors and hidden states are initialized to 300. Meanwhile, for BERT-based baselines, the dimensions of word embedding vectors and hidden states are 768. The learning rate and coefficient of $L2$ regulation for all baselines are set to 2e-5 and 1e-5, respectively. Other hyper-parameters not mentioned above adopt the same settings as the model's original paper. Details of the baseline models are presented below.

Non-BERT-based baseline models:

• **ATAE-LSTM** [29] is based on LSTM and the attention mechanism. Unlike the traditional LSTM-based model, ATAE-LSTM can make full use of aspect term information. The aspect embedding and context word embedding are concatenated and put into the LSTM neural network. Then, the weighted sum of hidden representations, which is obtained using an attention mechanism, is applied for prediction.

• **IAN** [59] learns the hidden representation of context and aspect words using two LSTMs. Then, the model uses an interactive attention network to obtain semantic information between the context and aspect. Finally, the aspect attention weight and context attention weight are fed into a softmax function to perform sentiment classification.

• **MemNet** [53] utilizes a deep memory network with several computational hops to generate sentence representations for aspect-level sentiment analysis. The attention mechanism is used multiple times, and can extract high-quality abstract features. The number of computational layers is set to 9 in the original paper.

• **RAM** [28] is a recurrent attention network that uses a bidirectional LSTM and multiple recurrent attention layers to obtain the hidden representation of a sentence. A special location-weighted memory module is designed to capture long-distance information. In addition, it uses a GRU network to combine the semantic features of the attention layers.

• **TNet-LF** [9] is a transformation attention network. It employs bidirectional LSTM, CNN, and transformation architecture to model hidden representations. A special CPT module, which contains a context-preserving and target-specific representations mechanism, is used many times in the transformation architecture.

• **MAN** [62] utilizes multiple attention mechanisms to judge the sentiment polarity of different aspect terms. It uses a transformer encoder instead of the RNN-based neural network to reduce the model's complexity. Furthermore, an inter-attention layer including global attention and local attention is introduced to capture differently grained interactive information between the aspect and context.

BERT-based baseline models:

• **BERT-SPC** [61] is a basic bidirectional encoder representation adapted from the transformers language model. The original sentence is converted into a sentence pair so that the basic BERT model can be fine-tuned for aspect-level sentiment analysis.

• **AEN-BERT** [61] overcomes the shortcomings of traditional recurrent neural networks, and employs the pre-trained BERT to construct the word embedding vector. This model uses an attention encoder mechanism for the modeling between context and aspect. Furthermore, a label smoothing regularization term is input into the loss function during model training.

• **MGMD** [67] is a multi-source data fusion model based on BERT. It learns sentiment knowledge from different types of resources. Moreover, a unified framework is designed to integrate data from multi-level sentiment lexicons.

4.4. Results comparisons

Table 3 presents the experimental results of MAMN and baseline models. The proposed model MAMN achieves the best results, whether MAMN_M or MAMN_W. Specifically, the $f1-score$ increased by approximately 4% and 2% in MAMN_W for SemEval 2014 restaurant reviews and the Twitter dataset, respectively. One of the reasons for this phenomenon is that the MAMN model benefits from the pre-trained BERT when initializing the word embedding vector. Furthermore, the MHSA and PWFF are used many times to ensure that the model can learn more abstract semantic information. Meanwhile, a special FFA mechanism forces the model to focus on short-distance words according to a specific aspect term. The performance of MAMN_W is superior to that of MAMN_M in most cases because the weights of hidden representations of context are just decreased instead of totally masked. In MMC, the weight values of words far away from the aspect term are masked as zero. The model only focuses on a few words in the preserve window, which does not provide sufficient semantic information for model learning.

For baseline models, the performance of ATAE-LSTM is not good since it only uses a simple LSTM structure. IAN is slightly better than ATAE-LSTM since the interactive attention mechanism adds two directions of information for final sentence representations: from context to aspect, and from aspect to context. Meanwhile, MemNet performs better than IAN owing to multiple computational layers being used, and the effectiveness of

Table 3
Comparison of different models on public datasets. The results with '†' are retrieved from published papers, and '-' indicates not reported. MAMN_M and MAMN_W refer to MMC and WDMC FFA mechanisms, respectively (see Section 3.4.1).

Model	Restaurant2014		Laptop2014		Restaurant2015		Restaurant2016		Twitter	
	acc	macro-f1	acc	macro-f1	acc	macro-f1	acc	macro-f1	acc	macro-f1
ATAE-LSTM†	77.20	–	68.70	–	–	–	–	–	–	–
IAN†	78.60	–	72.10	–	–	–	–	–	–	–
MemNet†	80.95	–	72.71	–	–	–	–	–	–	–
RAM	80.61	69.79	73.51	70.35	80.79	68.91	84.37	70.48	70.94	69.89
TNet-LF	80.72	70.52	76.25	71.51	81.43	68.78	85.03	71.49	74.59	73.41
MAN	84.29	71.36	78.21	72.98	82.59	69.35	85.69	73.16	76.70	72.41
BERT-SPC	84.32	77.15	78.54	75.26	83.27	68.93	86.58	74.62	73.55	72.14
AEN-BERT	83.50	74.28	79.46	76.15	83.79	69.17	88.61	76.73	74.89	73.36
MGMD	85.92	78.46	79.39	76.20	84.02	69.28	89.35	77.46	74.62	73.34
MAMN_M	86.25	80.97	81.05	77.77	84.85	**70.10**	90.22	78.75	76.57	74.97
MAMN_W	**86.52**	**81.57**	**81.35**	**77.83**	**84.97**	68.49	**90.34**	**79.21**	**76.59**	**75.27**

multiple attention layers is superior to that of a single attention layer. RAM and TNet-LF outperform these baselines. For TNet-LF, the *accuracy* and $f1-score$ on laptop and Twitter reviews improved sharply. This is because TNet-LF adds several CPT modules and a CNN layer, which can extract long-distance and non-sequential information. MAN applies multiple attention mechanisms to model sentence representations. Unlike recurrent neural network-based models, it uses three transform encoder layers to obtain hidden representations of context and aspect words. Therefore, the performance of MAN is better than that of RNN-based models. BERT-SPC outperforms MAN but cannot perform as effectively as AEN-BERT due to the basic BERT-based model used. AEN-BERT not only uses the pre-trained BERT for initial word embedding, but also multiple attention mechanisms. Therefore, AEN-BERT is superior to BERT-SPC. Lastly, since MGMD learns sentiment information from different types of resources, it obtains the best results among the baseline models.

4.5. Model analysis

In this section, we analyze the experimental results of our model MAMN. First, we focus on the influence of the number of layers in intra-level attention. Then, we investigate the rationality of each component in the MAMN model. Finally, we analyze the effect of the number of epochs on the training process.

4.5.1. Analysis of the number of layers

As mentioned in Section 3.3, it is impossible for a single attention layer to extract all sentiment relations in a sentence. Complex semantic information between words must be processed with multiple layers. Therefore, we conduct experiments to observe the effect of the number of layers in intra-level attention. The results are shown in Table 4.

Overall, the *accuracy* and $f1-score$ increase with the number of attention layers added. This proves that a stack of t attention layers indeed provides the model with more abstract sentiment relations between words. It should be noted that there are several special situations that are exceptions. For example, the *accuracy* for SemEval 2014 restaurant reviews and the $f1-score$ for Twitter reviews are best results with MAMN_W(2) and MAMN_W(4), respectively. However, the most superior *accuracy* and $f1-score$ occurred with MAMN_M(3) or MAMN_W(3), which indicates a three-layer structure is best. When the number of layers is equal to four, the performance of the proposed model (both MAMN_M and MAMN_W) declines due to poor generalization ability caused by complex calculations in multiple layers.

4.5.2. Analysis of several variants of the proposed model

To verify the rationality of each component in the MAMN, we design some variants based on the MAMN model and conduct ablation experiments. As shown in Table 5, MAMN(M) and MAMN(P) refer to the proposed model using only MHSA and PWFF in intra-level attention layers, respectively. MAMN(L) denotes the model where we replace the MHSA and PWFF with bidirectional LSTM, just like in some of the baseline models mentioned in Section 4.3. MAMN_M and MAMN_W are the complete versions of the proposed model.

It can be seen that the complete versions of the proposed model obtain the best performance on five public datasets except in rare cases. The MAMN without the FFA mechanism performs poorly, illustrating the importance of the masked mechanism of context and weighted down mechanism of context. The $f1-score$ of the MAMN(M)_W is highest on the Twitter dataset. This may be because the data format is more suitable for the model to learn semantic features of a sentence. In addition, we notice that the MAMN(L)_W obtains exciting results on the first three datasets. For restaurant reviews of SemEval 2014 and SemEval 2015, MAMN(L)_W obtains the highest *accuracy* and $f1-score$, respectively. However, the shortcomings of bidirectional LSTM used in the MAMN are obvious, such as the computational complexity and time consumption. Thus, we did not consider this structure in the final proposed MAMN model.

4.5.3. Analysis of the number of epochs

To evaluate the stability of the MAMN model, we analyze the change in *accuracy* and $f1-score$ on SemEval 2014 and the Twitter dataset. The curves are presented in Figs. 5–7.

As shown in Fig. 5(a), the *accuracy* on all datasets increased stably over the first epochs. When the training epochs increased to 25, the best result was achieved. However, the $f1-score$ illustrates rising volatility in Fig. 5(b). The reason for this phenomenon is that the $f1-score$ belongs to a comprehensive evaluation standard. It evaluates the performance of the model from two perspectives, precision and recall, and thus is sensitive to many factors. When the number of training epochs is more than 25, the *accuracy* and $f1-score$ remain stable. The curves in Figs. 6 and 7 are similar to those in Fig. 5, therefore we do not discuss them in detail again. Overall, the performance of WDMC is superior to that of MMC, for both *accuracy* and $f1-score$; This proves that reducing the weights of hidden representations is more reasonable than totally masking them.

4.6. Case study

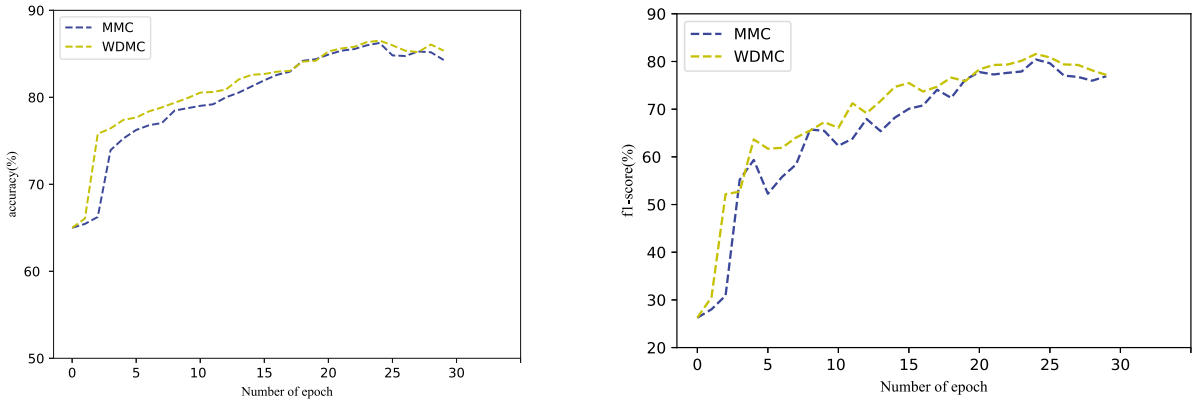
In order to provide an intuitive understanding of the attention mechanism used in the MAMN model, we visualize the attention weights of two sentences in Fig. 8. The deeper color of the word, the greater the effect on the sentence.

Table 4
Effect of the number of layers. t in MAMN_M(t) or MAMN_W(t) refers to the number of intra-level attention layers.

Model	Restaurant2014		Laptop2014		Restaurant2015		Restaurant2016		Twitter	
	acc	macro-f1	acc	macro-f1	acc	macro-f1	acc	macro-f1	acc	macro-f1
MAMN_M(1)	85.18	78.41	79.47	76.40	84.02	67.05	88.94	76.98	74.92	73.69
MAMN_W(1)	85.89	79.50	79.94	76.43	84.50	67.11	89.52	77.04	75.00	73.94
MAMN_M(2)	85.98	79.90	80.41	76.87	84.52	67.96	89.58	77.65	75.14	74.05
MAMN_W(2)	**86.70**	80.43	80.72	77.58	84.73	68.43	89.99	77.83	75.43	74.74
MAMN_M(3)	86.25	80.97	81.05	77.77	84.85	**70.10**	90.22	78.75	76.57	74.97
MAMN_W(3)	86.52	**81.57**	**81.35**	**77.83**	**84.97**	68.49	**90.34**	**79.21**	**76.59**	75.27
MAMN_M(4)	85.36	78.91	79.47	76.35	84.26	67.02	88.82	78.79	76.30	75.25
MAMN_W(4)	85.62	79.48	80.56	77.16	84.62	68.36	89.02	78.85	76.45	**75.58**

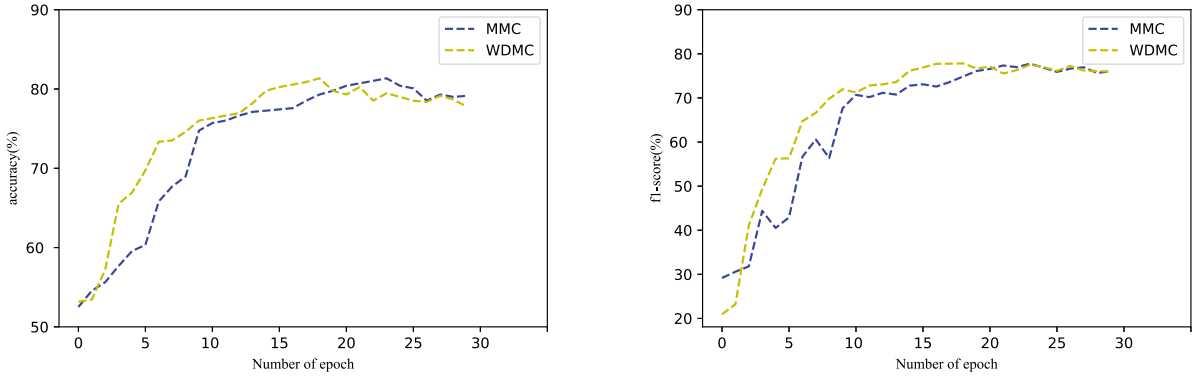
Table 5
Effect of different components. "w/o" indicates "without.".

Model	Restaurant2014		Laptop2014		Restaurant2015		Restaurant2016		Twitter	
	acc	macro-f1	acc	macro-f1	acc	macro-f1	acc	macro-f1	acc	macro-f1
MAMN(M)_M	85.69	79.12	79.77	75.49	84.09	67.65	88.82	76.67	75.14	73.35
MAMN(M)_W	85.77	79.70	79.91	75.87	84.62	67.88	89.17	77.76	76.45	**75.35**
MAMN(P)_M	85.71	79.75	78.68	75.64	84.26	66.15	88.71	76.11	75.29	74.17
MAMN(P)_W	86.34	79.78	79.31	75.49	84.38	67.71	89.99	78.35	75.43	74.36
MAMN w/o FFA	85.27	78.53	79.47	75.87	83.09	67.29	88.21	77.78	76.16	74.55
MAMN(L)_M	86.92	80.50	80.39	76.88	83.43	68.21	88.35	74.57	74.86	73.67
MAMN(L)_W	**87.09**	80.72	81.06	77.47	84.62	**70.84**	88.94	76.13	75.00	73.73
MAMN_M	86.25	80.97	81.05	77.77	84.85	70.10	90.22	78.75	76.57	74.97
MAMN_W	86.52	**81.57**	**81.35**	**77.83**	**84.97**	68.49	**90.34**	**79.21**	**76.59**	75.27



(a) (b)

Fig. 5. Change in (a) *accuracy* with the number of epochs and (b) $F1-score$ with the number of epochs. Experiments conducted on the restaurant dataset.



(a) (b)

Fig. 6. Change in (a) *accuracy* with the number of epochs and (b) $F1-score$ with the number of epochs. Experiments conducted on the laptop dataset.

As can be seen from Fig. 8, the sentences "excellent sashimi, and the millennium roll is beyond delicious" and "small screen somewhat limiting but great for travel" are taken from the restaurant reviews and laptop reviews, respectively. The aspects and predicted sentiment polarity of the model are presented before and after the given sentence. In the first sentence, the context word "excellent" is important in judging the sentiment polarity of aspect "sashimi" since it has a closer semantic relation than other

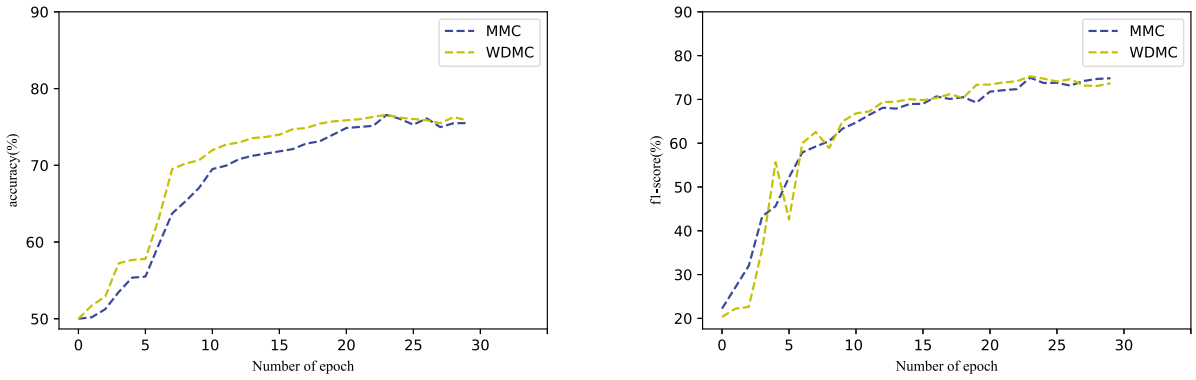


Fig. 7. Change in (a) *accuracy* with the number of epochs and (b) $F1-score$ with the number of epochs. Experiments conducted on the Twitter dataset.

Aspects	Sentence	Polarity
sashimi	Excellent sashimi . and the millennium roll is beyond delicious .	positive
millennium roll	Excellent sashimi . and the millennium roll is beyond delicious .	positive
screen	Small screen somewhat limiting but great for travel .	negative
travel	Small screen somewhat limiting but great for travel .	positive

Fig. 8. Visualized attention weights of two sentences.

contexts such as "and" or "roll". Additionally, some common words that do not carry sentiment polarity, like "and", "is", and "the" are given small attention weights in the MAMN. In the second sentence, the context words "small" and "limiting" are more vital in classifying the sentiment polarity of the aspect word "screen". The context word "great" is not allocated more attention because the word "but" occurred in front of it. This indicates that the model has learned the long-distance semantic features correctly. Therefore, we can conclude that the multiple attentions supplemented by the FFA mechanism enable the model to focus on the important words associated with a given aspect and learn semantic information between words automatically.

5. Conclusions and directions for future work

In this study, we propose a novel model MAMN to improve aspect-level sentiment analysis. The MAMN adopts the pre-trained BERT model to initialize word embeddings, which makes it better than common methods such as Word2vec or Glove. Multiple attention mechanisms including intra-level attention and inter-level attention are used to generate the hidden representations of a sentence. In intra-level attention, MHSA and PWFF instead of RNN-based structure are applied to process word embeddings. In inter-level attention, a special FFA module is designed to force the model to focus on those vital words that are closer to a given aspect. Experimental results demonstrate that the proposed model obtains the best performance in several aspect-level sentiment analysis tasks. In the future, we will try to add some auxiliary information, such as syntactic dependency and knowledge graphs, to improve the model's performance on aspect-level sentiment analysis.

CRediT authorship contribution statement

Xiaodi Wang: Conceptualization, Methodology, Formal analysis, Writing - original draft. **Mingwei Tang:** Resources, Writing - review & editing, Supervision, Project administration, Funding acquisition. **Tian Yang:** Software, Validation, Visualization. **Zhen Wang:** Investigation, Data curation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work is supported by the Foundation of Cyberspace Security Key Laboratory of Sichuan Higher Education Institutions, China (No. sjzz2016-73), "Chun Hui" Research Funds for Educational Department of China, China under Grant (No. Z2016151), the key of Scientific Research Funds Project of Xihua University, China (No. Z17133), the Scientific Research Funds Project of Science and Technology Department of Sichuan Province, China (No. 2016JY0244, 2017JQ0059, 2019GFW131, 2020JY, 2020GFW), Funds Project of Chengdu Science and Technology Bureau, China (No. 2017-RK00-00026-ZF) and the National Natural Science Foundation of China (No. 61902324), the Fund of Sichuan Educational Committee, China (17ZA0360), Sichuan Youth Science and technology innovation research team, China (2021**).

References

[1] Xianghua Fu, Wangwang Liu, Yingying Xu, Laizhong Cui, Combine hownet lexicon to train phrase recursive autoencoder for sentence-level sentiment analysis, Neurocomputing 241 (JUN.7) (2017) 18–27.

[2] Jun Zhao, Kang Liu, Liheng Xu, Sentiment analysis: Mining opinions, sentiments, and emotions, Comput. Linguist. 42 (3) (2016) 595–598.

[3] Kim Schouten, Flavius Frasincar, Survey on aspect-level sentiment analysis, IEEE Trans. Knowl. Data Eng. 28 (3) (2016) 813–830.

[4] Jialong Tang, Ziyao Lu, Jinsong Su, Yubin Ge, Linfeng Song, Le Sun, Jiebo Luo, Progressive self-supervised attention learning for aspect-level sentiment analysis, in: ACL 2019 : The 57th Annual Meeting of the Association for Computational Linguistics, 2019, pp. 557–566.

[5] Lei Shu, Hu Xu, Bing Liu, Lifelong learning CRF for supervised aspect extraction, 2017.

[6] Mauro Dragoni, Marco Federici, Andi Rexha, ReUS: a real-time unsupervised system for monitoring opinion streams, Cogn. Comput. (2019).

[7] Toqir A. Rana, Yu-N Cheah, Aspect extraction in sentiment analysis: comparative analysis and survey, Artif. Intell. Rev. (2016).

[8] Soujanya Poria, Erik Cambria, Alexander Gelbukh, Aspect extraction for opinion mining with a deep convolutional neural network, Knowl.-Based Syst. 108 (sep.15) (2016) 42–49.

[9] Xin Li, Lidong Bing, Wai Lam, Bei Shi, Transformation networks for target-oriented sentiment classification, 2018.

[10] Ruidan He, Wee Sun Lee, Hwee Tou Ng, Daniel Dahlmeier, Exploiting document knowledge for aspect-level sentiment classification, 2018.

[11] Hongning Wang, Yue Lu, Chengxiang Zhai, Latent aspect rating analysis on review text data: a rating regression approach, in: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, 2010.

[12] T. Joachims, Transductive inference for text classification using support vector machines, in: Sixteenth International Conference on Machine Learning, 1999.

[13] Albert Weichselbraun, Stefan Gindl, Arno Scharl, Extracting and grounding contextualized sentiment lexicons, IEEE Intell. Syst. 28 (2) (2013) 39–46.

[14] Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, Yoshua Bengio, Learning phrase representations using RNN encoder-decoder for statistical machine translation, Comput. Sci. (2014).

[15] M. Schuster, K.K. Paliwal, Bidirectional recurrent neural networks, IEEE Trans. Signal Process. 45 (11) (2002) 2673–2681.

[16] Fuhai Chen, Rongrong Ji, Jinsong Su, Donglin Cao, Yue Gao, Predicting microblog sentiments via weakly supervised multimodal deep learning, IEEE Trans. Multimed. 20 (4) (2018) 997–1007.

[17] Garen Arevian, Recurrent neural networks for robust real-world text classification, in: IEEE/WIC/ACM International Conference on Web Intelligence, 2007.

[18] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (8) (1997) 1735–1780.

[19] Junyoung Chung, Caglar Gulcehre, Kyung Hyun Cho, Yoshua Bengio, Empirical evaluation of gated recurrent neural networks on sequence modeling, 2014, Eprint Arxiv.

[20] Duyu Tang, Bing Qin, Xiaocheng Feng, Ting Liu, Effective LSTMs for target-dependent sentiment classification, Comput. Ence (2015).

[21] Sun Cheng-Ai, Zhao Rui, Tian Gang, Gated convolution network and emotion analysis based on aspect with CNN attention mechanism, Comput. Eng. Softw. (2019).

[22] Yi Cai, Qingbao Huang, Zejun Lin, Jingyun Xu, Qing Li, Recurrent neural network with pooling operation and attention mechanism for sentiment analysis: A multi-task learning approach, Knowl.-Based Syst. 203 (2020) 105856.

[23] Jiyao Wei A, Jian Liao A, Zhenfei Yang A, Suge Wang A B, Qiang Zhao A, BiLSTM with multi-polarity orthogonal attention for implicit sentiment analysis, Neurocomputing 383 (2020) 165–173.

[24] Dzmitry Bahdanau, Kyunghyun Cho, Yoshua Bengio, Neural machine translation by jointly learning to align and translate, Comput. Ence (2014).

[25] Volodymyr Mnih, Nicolas Heess, Alex Graves, koray kavukcuoglu, Recurrent models of visual attention, in: Z. Ghahramani, M. Welling, C. Cortes, N.D. Lawrence, K.Q. Weinberger (Eds.), Advances in Neural Information Processing Systems 27, Curran Associates, Inc., 2014, pp. 2204–2212.

[26] Yiming Cui, Zhipeng Chen, Si Wei, Shijin Wang, Ting Liu, Guoping Hu, Attention-over-attention neural networks for reading comprehension, in: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2017, pp. 593–602.

[27] Karl Moritz Hermann, Tomá Koisk, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, Phil Blunsom, Teaching machines to read and comprehend, 2015.

[28] Peng Chen, Zhongqian Sun, Lidong Bing, Wei Yang, Recurrent attention network on memory for aspect sentiment analysis, in: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, 2017.

[29] Yequan Wang, Minlie Huang, Xiaoyan Zhu, Li Zhao, Attention-based LSTM for aspect-level sentiment classification, in: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, 2016.

[30] Xiaodi Wang, Tian Yang, Xiaoliang Chen, Zhen Wang, Mingwei Tang, Aspect-level sentiment analysis based on position features using multilevel interactive bidirectional GRU and attention mechanism, Discrete Dyn. Nat. Soc. 2020 (2020).

[31] Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: NAACL-HLT 2019: Annual Conference of the North American Chapter of the Association for Computational Linguistics, 2019, pp. 4171–4186.

[32] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin, Attention is all you need, in: Proceedings of the 31st International Conference on Neural Information Processing Systems, 2017, pp. 5998–6008.

[33] Biqing Zeng, Heng Yang, Ruyang Xu, Wu Zhou, Xuli Han, LCF: A local context focus mechanism for aspect-based sentiment classification, Appl. Ences 9 (16) (2019) 3389.

[34] Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Semeval-2014 task 4: aspect based sentiment analysis, in: Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), 2014, pp. 27–35.

[35] Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Suresh Manandhar, Ion Androutsopoulos, SemEval-2015 task 12: aspect based sentiment analysis, in: Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), 2015, pp. 486–495.

[36] Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammad AL-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphee De Clercq, Veronique Hoste, Marianna Apidianaki, Xavier Tannier, Natalia Loukachevitch, Evgeniy Kotelnikov, Núria Bel, Salud María Jiménez-Zafra, GlÅen Eryiit, Semeval-2016 task 5 : aspect based sentiment analysis, in: Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016), 2016, pp. 19–30.

[37] E. Cambria, Affective computing and sentiment analysis, IEEE Intell. Syst. 31 (2) (2016) 102–107.

[38] Qinjuan Yang, Yanghui Rao, Haoran Xie, Jiahai Wang, Fu Lee Wang, Wai Hong Chan, Segment-level joint topic-sentiment model for online review analysis, IEEE Intell. Syst. 34 (1) (2019) 43–50.

[39] Erik Cambria, Soujanya Poria, Amir Hussain, Bing Liu, Computational intelligence for affective computing and sentiment analysis [guest editorial], IEEE Comput. Intell. Magaz. 14 (2) (2019) 16–17.

[40] Nidhika Yadav, Niladri Chatterjee, Text summarization using sentiment analysis for DUC data, in: International Conference on Information Technology, 2017.

[41] Delip Rao, Deepak Ravichandran, Semi-supervised polarity lexicon induction, in: Eacl, Conference of the European Chapter of the Association for Computational Linguistics, Conference, Athens, Greece, 2009.

[42] Marco Federici, Mauro Dragoni, A knowledge-based approach for aspect-based opinion mining, in: Semantic Web Evaluation Challenge, 2016.

[43] Bo Pang, Lillian Lee, Shivakumar Vaithyanathan, Thumbs up? sentiment classification using machine learning techniques, in: Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing, 2002, pp. 79–86.

[44] Mohammad Al-Smadi, Mahmoud Al-Ayyoub, Yaser Jararweh, Omar Qawasmeh, Enhancing aspect-based sentiment analysis of arabic hotels' reviews using morphological, syntactic and semantic features, Inf. Process. Manage. 55 (2) (2018) 308–319.

[45] Long Jiang, Mo Yu, Ming Zhou, Xiaohua Liu, Tiejun Zhao, Target-dependent Twitter sentiment classification, in: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Vol. 1, 2011, pp. 151–160.

[46] Chao Yang, Hefeng Zhang, Bin Jiang, Keqin Li, Aspect-based sentiment analysis with alternating coattention networks, Inf. Process. Manage. 55 (3) (2019) 463–478.

[47] Hai Ha Dohaiha, Pwc Prasad, Angelika Maag, Abeer Alsadoon, Deep learning for aspect-based sentiment analysis: A comparative review, Expert Syst. Appl. 118 (2018).

[48] Tomas Mikolov, Martin Karafiát, Lukas Burget, Jan Cernock, Sanjeev Khudanpur, Recurrent neural network based language model, in: Interspeech, Conference of the International Speech Communication Association, Makuhari, Chiba, Japan, 2015.

[49] Duyu Tang, Bing Qin, Xiaocheng Feng, Ting Liu, Target-dependent sentiment classification with long short term memory, Comput. Sci. (2015).

[50] Yukun Ma, Haiyun Peng, Tahir Khan, Erik Cambria, Amir Hussain, Sentic LSTM: a hybrid network for targeted aspect-based sentiment analysis, Cogn. Comput. 10 (4) (2018) 639–650.

[51] Sixing Wu, Yuanfan Xu, Fangzhao Wu, Zhigang Yuan, Xing Li, Aspect-based sentiment analysis via fusing multiple sources of textual knowledge, Knowl.-Based Syst. 183 (2019) 104868.

[52] Shuqin Gu, Lipeng Zhang, Yuexian Hou, Yin Song, A position-aware bidirectional attention network for aspect-level sentiment analysis, in: COLING 2018: 27th International Conference on Computational Linguistics, 2018, pp. 774–784.

[53] Duyu Tang, Bing Qin, Ting Liu, Aspect level sentiment classification with deep memory network, in: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, 2016, pp. 214–224.
[54] Wei Xue, Tao Li, Aspect based sentiment analysis with gated convolutional networks, 2018.
[55] Huy Thanh Nguyen, Minh Le Nguyen, Effective attention networks for aspect-level sentiment classification, in: 2018 10th International Conference on Knowledge and Systems Engineering (KSE), 2018.
[56] Xinyi Wang, Guangluan Xu, Jingyuan Zhang, Xian Sun, Lei Wang, Tinglei Huang, Syntax-directed hybrid attention network for aspect-level sentiment analysis, IEEE Access 7 (2019) 5014–5025.
[57] Yi Cai, Qingbao Huang, Zejun Lin, Jingyun Xu, Qing Li, Recurrent neural network with pooling operation and attention mechanism for sentiment analysis: A multi-task learning approach, Knowl.-Based Syst. 203 (2020) 105856.
[58] Feiyang Ren, Liangming Feng, Ding Xiao, Ming Cai, Sheng Cheng, Dnet: A lightweight and efficient model for aspect based sentiment analysis, Expert Syst. Appl. (2020) 113393.
[59] Dehong Ma, Sujian Li, Xiaodong Zhang, Houfeng Wang, Interactive attention networks for aspect-level sentiment classification, in: Twenty-Sixth International Joint Conference on Artificial Intelligence, 2017.
[60] Binxuan Huang, Yanglan Ou, Kathleen M. Carley, Aspect level sentiment classification with attention-over-attention neural networks, 2018.
[61] Youwei Song, Jiahai Wang, Tao Jiang, Zhiyue Liu, Yanghui Rao, Attentional encoder network for targeted sentiment classification, 2019.
[62] Qiannan Xu, Li Zhu, Tao Dai, Chengbing Yan, Aspect-based sentiment classification with multi-attention network, Neurocomputing 388 (2020).
[63] Ning Liu A. B, Bo Shen A. B, Rememnn: A novel memory neural network for powerful interaction in aspect-based sentiment analysis, Neurocomputing 395 (2020) 66–77.
[64] Y. Lecun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (7553) (2015) 436.
[65] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, Ruslan Salakhutdinov, Dropout: a simple way to prevent neural networks from overfitting, J. Mach. Learn. Res. 15 (1) (2014) 1929–1958.
[66] Li Dong, Furu Wei, Chuanqi Tan, Duyu Tang, Ke Xu, Adaptive recursive neural network for target-dependent Twitter sentiment classification, in: Meeting of the Association for Computational Linguistics, 2014.
[67] Fang Chen, Zhigang Yuan, Yongfeng Huang, Multi-source data fusion for aspect-level sentiment classification, Knowl.-Based Syst. 187 (2020).