Residual Fusion Network Based Attention for Aspect Level Sentiment Analysis

Long Wang¹, Lijiang Chen¹, Linghan Cai¹ and Xia Mao*¹
¹Beihang University, School of Electronic and Information Engineering, Beijing, China.
Email: wang_long@buaa.edu.cn, chenlijiang@buaa.edu.cn, cailh@cau.edu.cn
*Corresponding Author Email: moukyo@buaa.edu.cn

Abstract. Aspect level sentiment analysis is a more fine-grain task compared to document level and sentence level. The aim of the task is to identify the sentiment polarity expressed to the given aspect in the text. In this paper, we propose a residual aspect fusion network with attention for the aspect level sentiment classification. In this network, bidirectional GRU is used to collect semantic information of both sentence words and aspect words, and then attention mechanism is applied to construct aspect vectors for each word in the sentence. In the residual fusion module, the model fuses the aspect information and sentence information together by residual connection structure, in which position weight is applied to utilize the position information of aspect words. At the end of the model, Transformer encode layer is used to extract global feature and fully connected layers are used as the classifier. We train and test the network on SemEval2014 task 4 (Restaurant and Laptop) and Twitter datasets, and the experiments demonstrate that our model performs better than previous methods.

1. Introduction
The main purpose of sentiment analysis is to identify users' opinions on things or people from subjective texts containing emotional information. According to the granularity the task focuses on, sentiment analysis can be generally divided into three sub-tasks: aspect level, sentence level, and document level. The aim of aspect level sentiment classification is to detect the user's sentiment polarity in text (such as comments, tweets, Weibo) for a particular aspect appears in [1]. The following uses an example to explain the content of this task. “The restaurant has great food but its service is slow”. In this sentence, we can know the sentiment polarity for the food is positive, but for the service is negative. Sentiment analysis on document level and sentence level can’t analyse the text contains different sentiment tendency, but on aspect level the shortcoming is overcome to some extent.

In this paper, we propose the model named Residual Fusion Network Based Attention (RFNet-ATT) for the task of aspect level sentiment analysis. The innovation of the network we proposed are as follows:

First, we propose the method of fuse aspect information and sentence information together based on residual fusion module. In this network, aspect vectors are constructed for every word in sentence. The residual fusion module is designed in our network to solve the problem of integrating aspect and sentence information. In this module, residual connection is adopted to reduce loss in the forward process, and position weight is applied to use the position information of aspect words.

Second, Transformer encode layer is applied in the model to extract global features. Transformer structure is first proposed in [2], and has wide application in the field of deep learning owing to its...
superior performance. Experiments in this paper show Transformer encode layer could improve model performance obviously.

2. Related Work

There have been many studies on sentiment analysis at the aspect level because of its great value in E-commerce and social media. In recent years, researchers have achieved many breakthrough results in the field of natural language processing by the method based deep learning. Scholars have begun to apply this technology to the task of aspect level sentiment analysis.

Tang et al. proposed TD-LSTM [3], used two LSTM process the sentence with considering the position of aspect words, obtained good result at that time. Recently, attention mechanism was introduced to aspect level sentiment analysis task after proposing in the field of machine translation [4], in order to capture more relevant information from the text and the aspect. Wang et al. proposed the Attention-Based LSTM with Aspect Embedding (ATAE-LSTM) [5]. The conception of aspect embedding was proposed, and attention mechanism is used to calculate weights which is used to get the weighted sum of the LSTM’s hidden layer as the representation of the sentence for a given aspect. Ma et al. proposed an Interactive Attention Network (IAN), the method generate representation vectors for text and aspect respectively through interactive learning [6]. Huang et al. proposed Attention-over-Attention network for the task, the method jointly learns aspect and sentence representations, and importance of different part of sentence is considered automatically [7].

Some scholars believe that the multi-attention layer model can make model performance improvements. Tang et al. proposed a deep memory network (MemNet) [8]. The method first stacks the word vectors of the context words of aspect words in the sentence to build the memory, multiple computational layers is designed to obtain sentence representation. The first layer regard aspect vector as input, and select important evidences from memory by attention layer, the output of attention layer and linear transformation of aspect vector are summed and the result is considered as the input of next layer. The output of last layer is the sentence representation given target. Chen et al. developed a Recurrent Attention Network on Memory (RAM) [9]. Bi-LSTM and position weight are used to build the memory. In the recurrent attention module, multiple attentions are used to fulfill the first aspect, and then a recurrent network nonlinearly combines the attention results with GRUs.

3. Method

3.1. Word Embedding

In the field of natural language processing, in order for a computer to process a sequence of words, the word need to be mapped to a high-dimensional real vector. This technique is called word embedding.

For the task of sentiment analysis at aspect level, the aspect words and sentence words need to be processed respectively. We need get two sets of word vectors. Give a sentence $s = [w_1, w_2 ... w_k, ... w_n]$ contains aspect words $a = [w_k, ... w_{k+d}]$, where $n$ is the length of the sentence, $k$ and $d$ is the start position and length of aspect words, we can get word vector sequences $V_s = [v_1, v_2, ..., v_k, ... v_n]$ from sentence words and $V_a = [v_k, ... v_{k+d}]$ from aspect words.

3.2. GRU Layer

GRU is a kind of recurrent neural network and a variant of LSTM [10]. In sentiment analysis tasks, bidirectional GRU is often used as an encoder to re-encode the input word vector sequence, so that each element in the hidden states is integrated with the context information.

The model uses bidirectional GRU to encode sentence words and aspect words to get their semantic representation. Given a sequence of word vectors as input, the bidirectional GRU would generate a sequence of hidden states. The computing process is as following:

$$H_r = \overrightarrow{GRU}([v_1, v_2 ... v_k, ... v_n])$$

$$H_l = \overleftarrow{GRU}([v_1, v_2 ... v_k, ... v_n])$$

$$H_s = [h_r, h_s]$$
Given the input \( v = [v_1, v_2, \ldots, v_k, \ldots, v_n] \), the forward GRU will generate the hidden state \( H_f \in \mathbb{R}^{n \times d_h} \) and the backward GRU will generate the hidden state \( H_b \in \mathbb{R}^{n \times d_h} \), \( H_s \in \mathbb{R}^{n \times 2d_h} \) is the concatenation of \( H_f \) and \( H_b \), where \( d_h \) is the dimension of hidden state in the GRU cell.

3.3. The Design of Attention Mechanism

In some previous research, such as ATAE-LSTM, they get the aspect vector by average pooling the representation of aspect words, with not considering the relationship between aspect words and the sentence words. The method of constructing aspect vector for every word in sentence is firstly proposed in TNet [11], the aspect vector is calculated many times in its CPT modules. In our model, the attention mechanism is used to construct a corresponding aspect vector for each word in the sentence like the TNet, but differently, our network calculates aspect vector one time.

Given the semantic representation of the sentence \( H_s = [h_1, h_2, \ldots, h_n] \in \mathbb{R}^{n \times d_h} \), and the semantic representation of the aspect words \( H_a = [h_{10}, h_{11}, \ldots, h_{k+d}] \in \mathbb{R}^{d \times 2d_h} \) have been obtained from the bidirectional GRU. First, the network calculates the correlation matrix \( I \in \mathbb{R}^{n \times d} \) of sentence words and aspect words as the equation (4). Then the weights of each aspect word to construct the aspect vector for each word in sentence can be calculated. Given the \( i \)-th word in sentence, we get the weights assign to each aspect word by equation (5). For the matrix \( I \in \mathbb{R}^{n \times d} \), we get the weight matrix for aspect word with row-wise softmax. By considering the contribution of each aspect word, we calculate the aspect vector \( A_{si} \) for the \( i \)-th sentence word as the equation (6).

\[
I = H_s \cdot H_a^T \tag{4}
\]

\[
w_{ij} = \frac{\exp(i_{ij})}{\sum_j \exp(i_{ij})} \tag{5}
\]

\[
A_{si} = \sum_j w_{ij} H_{aj} \tag{6}
\]

3.4. Residual Fusion Module

The residual structure was first proposed in the field of computer vision to solve the problem of model degradation. He et al. proposed the residual network firstly in the ImageNet competition and won the championship [12]. In our model, we designed a residual fusion module. In this module, residual structure and fully connected layer are used to integrate aspect information and sentence information together [11], and weights are assigned to the sentence words according to the distance from the target.

From the bidirectional GRU and attention module, we have get the semantic representation \( H_s \in \mathbb{R}^{n \times 2d_h} \) for sentence and the aspect vectors \( A_s \in \mathbb{R}^{n \times 2d_h} \). The model concatenate \( H_s \) and \( A_s \) as the matrix \( H_{sa} \in \mathbb{R}^{n \times 4d_h} \) (7), and then fuse the information of aspect and sentence by fully connected layer and residual structure. The calculating process is as follows:

\[
H_{sa} = [H_s, A_s] \tag{7}
\]

\[
H_s = H_s + \text{Relu}(H_{sa}W_f + b_f) \tag{8}
\]

In equation (8), \( W_f \) and \( b_f \) are the weights and bias in the fully connected layer, \( \text{Relu} \) is the activation function [13]. With the residual structure, \( H_s \) contains both sentence information and aspect information.

We mentioned above that a sentence may contain multiple aspects of different sentiment polarity. In practice, the words that are closer to the aspect words tend to have a greater effect on the sentiment tendency to the aspect [14]. So in the residual fusion module, we assign weights to the elements of \( H_s \) we get in the previous step according to the distance from the aspect words, the words that are closer to the aspect words would be assigned higher weights in comparison to those far away with aspect words. For example, “The restaurant has great food but its service is slow”, in this sentence, “great” contributes more to the sentiment polarity on “food” in comparison to “slow”. The weights can be calculated by the trapezoidal function. The process of assigning weights to \( H_{sa} \) is as follows:
In above equations, $n$ is the length of sentence after padding, $k$ is the start location of the aspect words, and $d$ is the length of aspect word, $w_i$ is the weight we assign to the $i$-th element in $H_s$.

In our model, the $H_s$ can be processed by residual fusion module many times to improve model performance. In other word, the residual fusion modules share same parameters can be stacked as Figure 2. The influence of the layer of residual fusion modules on the model would be discussed below.

\[ w_i = \begin{cases} 
1 - \frac{k-i}{n} & 0 \leq i < k \\
1 & k \leq i < k + d \\
1 - \frac{i-k-d+1}{n} & k + d \leq i \leq n 
\end{cases} \]  

(9)

\[ H_{si} = w_i H_{si} \]  

(10)

Figure 1. The architecture of Residual Fusion Module.

3.5. Transformer Encoder Layer

In our model, Transformer encode layer is used to extract more information from the output of residual fusion module. Transformer structure is based on multi-head self-attention mechanism, and has better ability to extract global features. Therefore, Transformer encode layer is widely used in deep model both in natural language processing and image fields as feature extractor. The process of Transformer encode layer is as follows:

\[ H_{t1} = \text{MultiheadAttention}(H_s, H_s, H_s) \]  

(11)

\[ H_{t2} = \text{LayerNorm}(H_s + H_{t1}) \]  

(12)

\[ H_{t3} = \max(0, H_{t2} W_1 + b_1) W_2 + b_2 \]  

(13)

\[ H_{out} = \text{LayerNorm}(H_{t2} + H_{t3}) \]  

(14)

The details of multi-head self-attention mechanism and layer normalization can be seen in [3].

3.6. Final Classification

The calculating process of this section are as follows:

\[ c = \text{maxpool}(H_{out}) \]  

(15)

\[ r = W_h2(\text{Relu}(W_{h1} c + b_{h1})) + b_{h2} \]  

(16)
In above equations, $c$ represents the output of max-pooling layer. $W_{h1}, W_{h2}$ and $b_{h1}, b_{h2}$ are the weights and biases of fully connected layers. With the softmax, we calculate $\hat{y}_i$ represents the probability the text expresses which sentiment polarity on given aspect.

The cross entropy is used in our model as the loss function, and $L_2$ regularization is added to loss function to avoid overfitting. As follows, $y_i$ is the label for each sample in training set $T$, whose value is 1 indicates the correct sentiment polarity, and 0 indicates the wrong, $\lambda$ is the penalty coefficient for model’s parameters $\theta$.

$$
\text{Loss} = -\sum_T \sum_{i=1}^{3} y_i \log \hat{y}_i + \lambda \| \theta \|^2
$$

4. Experiments

4.1. Datasets

We use the SemEval2014 task 4 dataset and the Twitter dataset to train and test the model. The SemEval2014 dataset is a dataset of semantic evaluation competition tasks, including user reviews in two areas: laptop and restaurant. Twitter dataset contains reviews obtained from twitter. The sentiment polarity (positive, negative or neutral) of both training and testing data were labeled. The distribution of experimental data used in this paper are as follows:

| Dataset       | Positive | Negative | Neutral |
|---------------|----------|----------|---------|
| Laptop-train  | 994      | 870      | 464     |
| Laptop-test   | 341      | 128      | 169     |
| Restaurant-train | 2164   | 807      | 637     |
| Restaurant-test | 728     | 196      | 196     |
| Twitter-train | 1567     | 1563     | 3127    |
| Twitter-test  | 174      | 174      | 346     |

4.2. Training Details

In the experiments of this paper, the word vectors we used are Glove word vectors proposed by Pennington et al. [15], where each word vector has a dimension of 300. To prevent overfitting of the model, we use the $L_2$ regularization and dropout methods, where the $L_2$ regular term coefficient is set by 0.001, and the dropout rate is set by 0.1. During the training process, the Adam optimizer is adopted to updates the model parameters [16], and the learning rate is set by 5e-4 and batch size of training data is 32.
4.3. Result Comparisons
The accuracy on test datasets is adopted as the evaluation metric in this paper. The following is results comparison of previous methods and ours:

| Model          | Restaurant | Laptop | Twitter |
|----------------|------------|--------|---------|
| LSTM           | 74.11      | 66.31  | 66.49   |
| TD-LSTM        | 76.25      | 68.97  | 68.06   |
| ATAE-LSTM      | 77.20      | 69.12  | 68.79   |
| IAN            | 78.57      | 70.94  | 69.12   |
| MemNet         | 78.38      | 71.33  | 68.50   |
| RAM            | 79.02      | 72.73  | 69.08   |
| TNet           | 79.22      | 73.26  | 70.09   |
| RFNet-ATT      | **80.27**  | **74.61** | **71.10** |

Table 2. The result comparison with previous models.
(The layer of residual fusion module is set by 3)

From the results in table, we can see that the model we proposed preforms better than those previous model for aspect level sentiment classification in accuracy on three datasets.

5. Discussion

5.1. The Effect of Attention Mechanism
In this section, the paper discusses the effect of attention to improve the performance of our model. In the compared model, attention mechanism is removed, and the aspect vector is calculated by average pooling $H_d$ for all sentence words. The accuracy on 3 datasets are as table 3:

When the attention mechanism removed, there is a significant decline in the test accuracy on three datasets. The reason is that aspect vector is calculated by average pooling without considering the correlation between sentence words and aspect words, which caused loss of semantic information, the contribution of each aspect words to aspect vector is neglected.

| Model       | Restaurant | Laptop | Twitter |
|-------------|------------|--------|---------|
|             | 79.39      | 73.16  | 70.22   |

Table 3. Results when attention is not used.

5.2. The Effect of Position Weight
In this section, the paper discusses the influence of position weights. In compared model, position weight layer was removed from residual fusion module, and the test accuracy is as follows:

| Model       | Restaurant | Laptop | Twitter |
|-------------|------------|--------|---------|
|             | 78.84      | 71.32  | 69.08   |

Table 4. Results when position weight is not used.

As shows in table 4, position weight can effectively use the relationship between aspect words and context. By giving more weight to words that are closer to the aspect word, the network can catch more important information given aspect.

5.3. The Effect of the Number of Residual Fusion Module
In the process of experiment, we found the performance of the model fluctuated with the layers of residual fusion module. We design experiments to verify the influence to the accuracy when different number residual fusion modules stacked in the network. We set the number of residual fusion module from 1 to 5, and the fluctuation of accuracy in three datasets is as table 5.
From the results we obtained, we can see that the test accuracy on the dataset of restaurant and laptop review reach the highest when 3 residual fusion modules stacked in network, and on twitter dataset when 4 residual fusion modules was used.

**Table 5.** Accuracy fluctuation with the layers of residual fusion module on three datasets.

| Layers | Restaurant | Laptop | Twitter |
|--------|------------|--------|---------|
| 1      | 79.64      | 71.47  | 69.08   |
| 2      | 79.12      | 73.04  | 70.66   |
| 3      | **80.27**  | **74.61** | 71.10   |
| 4      | 80.00      | 73.67  | **71.53** |
| 5      | 79.11      | 72.41  | 70.38   |

5.4. *The Effect of Transformer Encoder Layer*

In this experiment, Transformer encoder layer is removed from network, and keep 3 residual fusion modules same. The result of pooling the output of residual fusion module is fed to fully connected layers directly. The follows are the experiment results in 3 datasets.

**Table 6.** Results when Transformer encode layer is not used.

|         | Restaurant | Laptop | Twitter |
|---------|------------|--------|---------|
| Accuracy| 78.38      | 71.69  | 69.36   |

When Transformer encoder layer removed, the accuracy in test datasets decreased obviously. The output of residual fusion layer should not feed to classifier directly, which need Transformer encoder layer to extract more information on the whole sentence.

6. *Conclusion and Future Work*

In this paper, we propose the residual fusion network with attention for the aspect level sentiment analysis. The main idea of our method is that residual fusion module is designed to fuse the sentence and aspect information together, and the Transformer encoder layer is applied to extract global features. In order to verify that the model works well in the aspect level sentiment analysis, we compare the model with the previous classic methods for this task, and experiments show the accuracy is improved. In addition, the factors affecting model performance are discussed.

In future research, we plan to introduce external knowledge, such as sentiment lexicon, to the model. We will design new methods to embed external knowledge in the form of vectors to facilitate deep network training, with expectation to improve the accuracy on this task.

7. *Acknowledgments*

This research was supported in part by the National Science Foundation for Young Scientists of China under grant 61603013, and the Fundamental Research Funds for the Central Universities (No. YWF-20-BJ-J-197).

8. *References*

[1] Yohan Jo, Alice H. Oh 2013 Aspect and sentiment unification model for online review analysis Proc. 4th ACM Int. Conf. on Web Search and Data Mining (New York, NY, USA) pp 815-824.

[2] Ashish Vaswani, Noam Shazeer, et al 2017 Attention is all you need Advances in Neural Information Processing Systems pp 6000–6010.

[3] Tang D, Qin B, Feng X, et al 2016 Effective LSTMs for Target-Dependent Sentiment Classification Proc. COLING 2016, the 26th Int. Conf. on Computational Linguistics (Osaka, Japan) pp 3298–3307.

[4] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio 2014 Neural Machine Translation by Jointly Learning to Align and Translate arXiv: 1409.0473.
[5] Wang Y, Huang M, Zhu X, et al 2016 Attention-based LSTM for Aspect-level Sentiment Classification Proc.2016 Conf. on Empirical Methods in Natural Language Processing (Austin, Texas) pp 606–615.

[6] D. Ma, S. Li, X. Zhang and H. Wang 2017 Interactive attention networks for aspect-level sentiment classification Proc. 26th Int. Joint Conf. on Artificial Intelligence (Melbourne, Australia) pp 4068-4074.

[7] B. Huang, Y. Ou and K. M. Carley 2018 Aspect level sentiment classification with attention-over-attention neural networks arXiv:1804.06536v1.

[8] Duyu Tang, Bing Qin, Ting Liu 2016 Aspect level sentiment classification with deep memory network Proc. 2016 Conf. on Empirical Methods in Natural Language Processing (Austin, Texas) pp 214–224.

[9] Peng Chen, Zhongqian Sun, Lidong Bing, Wei Yang 2017 Recurrent attention network on memory for aspect sentiment analysis Proc. 2017 Conf. on Empirical Methods in Natural Language Processing (Copenhagen, Denmark) pp 452–461.

[10] Cho K, Van Merrienboer B, Gulcehre C, et al 2014 Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation Proc. 2014 Conf. on Empirical Methods in Natural Language Processing (Doha, Qatar) pp 1724–1734

[11] Li X, Bing L, Lam W, et al 2018 Transformation networks for target-oriented sentiment classification Proc. 56th Annual Meeting of the Association for Computational Linguistics pp 946-956.

[12] K. He, X. Zhang, S. Ren and J. Sun 2016 Deep Residual Learning for Image Recognition 2016 IEEE Conf. on Computer Vision and Pattern Recognition (Las Vegas, NV) pp 770-778.

[13] Glorot X, Bordes A, Bengio Y 2011 Deep Sparse Rectifier Neural Networks Journal of Machine Learning Research 15 315-323.

[14] J. Zeng, X. Ma and K. Zhou 2019 Enhancing Attention-Based LSTM With Position Context for Aspect-Level Sentiment Classification IEEE Access 7 20462-20471.

[15] Pennington J, Socher R, Manning C 2014 Glove: global vectors for word representation Proc. 2014 Conf. on Empirical Methods in Natural Language Processing (Doha, Qatar) pp 1532–1543.

[16] Kingma, Diederik P, and J. Ba 2014 Adam: A Method for Stochastic Optimization arXiv: 1412.6980.