An Aspect Level Sentiment Analysis Based on Embedding Tuning & Attention Neural Network

Jiachen Ding[1, a], Gongshen Liu[1, b, *], Kui Meng[1, c], and Weidong Qiu[1, d]

[1]School of Electronic Information and Electrical Engineering, Shanghai Jiaotong University, Shanghai 200240, China

[a] ding222@sjtu.edu.cn, [b] lgshen@sjtu.edu.cn, [c] mengkui@sjtu.edu.cn, [d] qiuwd@sjtu.edu.cn

**corresponding author*

Keywords: sentiment analysis; local attention; neural network; fine-tuning

Abstract: A novel framework based on neural network is proposed to predict sentiment polarity of aspects in a sentence. The proposed model finely tunes pre-trained word embedding in order to get more accurate embedding for aspect-based sentiment analysis task. Aspect attention based on its location and the whole sentence is calculated to generate importance of context words. The merits of our model are tested on three datasets: reviews of restaurants, laptops from SemEval2014, and review collected from twitter for testing robust of our model on language form. It is shown by experiments that the proposed model outperforms the state-of-the-art methods on these datasets.

1. Introduction

Aspect based sentiment analysis (ABSA) aims to predict sentiment polarity (i.e., negative, neutral, or positive) based on a sentence and a specific aspect. It is very useful in commerce reviews, because customers usually have complex comments on products. They may like the aspect of a product, while thinking the other aspect not good enough. For example, in “Drivers updated ok but the BIOS update froze the system up and the computer shut down”, which is a laptop review, there are three aspects, “drivers”, “BIOS update” and “system”. The customer has positive sentiment on “drivers”, but has negative sentiment on “BIOS update” and “system”. Traditional sentence-oriented sentiment analysis methods (Socher et al., 2011; Appel et al., 2017) [1,2] are not able to classify sentiments via different aspects.

In order to well identify the sentiment polarity of a sentence based on specific aspect, one major task is to generate good context features for different aspects. In simple cases, there are explicit sentiment words beside the aspect, such as “Amazing Quality”. The sentiment word for “quality” is “amazing”, which expressing the sentiment obviously. However, in most cases, sentences are more complicated, so simple syn-tactically nearby rule does not work well. For example, “The quality is not amazing enough” may express a neutral sentiment, while “The quality is not amazing at all” may express a negative one. However, in both sentence, “amazing” is quite close to “quality”. What’s more, some sentences are more complex that it only conveys implicit meaning. For example, in “It weighed like seven pounds or something like that”, there are not explicit sentiment words which can show the sentiment polarity. And in this case, semantic analysis is needed. So, more powerful model

Published by CSP © 2018 the Authors

is needed to solve the above problems.

Thus, deep learning model appears. Target dependent LSTM (TD-LSTM) (Tang et al., 2015) [3] is able to represent phrase-like word sequence like "is not amazing enough". However, if the aspect is at the begin or end part of a sentence, forward LSTM or back forward LSTM cannot perform well, and features will be lost. As attention mechanism is successfully used in machine translation (Bahdanau et al., 2014) [4], more attention models become used in ABSA tasks (Wang et al., 2016; Tang et al.,2016; Ma et al., 2017) [5-7]. However, if a sentence contains both positive and negative sentiment polarity, it is still hard to differentiate the polarity via different aspects.

In this paper, we propose a novel model to solve above problems in ABSA tasks. First, in embedding layer, we fine-tuned the pre-trained embedding, hoping that tuned embedding can better separate antonyms. Antonyms are usually similar to each other, because they are both adjectives. However, in sentiment classification task, we want them to be as distinct as possible. Then, a bidirectional Gated Recurrent Unit (Bi-GRU) is used to produce memory, as GRU has less parameters than LSTM and our dataset is not very large. The memories are globally weighted, and locally weighted according to their distance to the aspect and whether it represents a punctuation mark or not. Both attentions are combined together with some learnt weight. Finally, softmax function is used to predict the probability of different sentiments. The detailed algorithm is described in section 3.

The main contributions of this model are followings: 1) both global and local attention techniques are used, so that the model can capture both long and short distance dependences; 2) embeddings are tuned during training to increase accuracy; 3) features are dynamically learnt during training.

2. Related Work

Aspect based sentiment classification is a fine-grained classification task. Traditional methods include rule-based methods and statistic methods. Lu et. al, (2003) [8] used SVM for classification and segmentation of an audio clip. Krishnlal G et al., (2013) [9] the intelligent news classifier was developed and experimented with online news from net for the class sports, finance and politics. The novel approach combined two powerful algorithms, hidden mathematician Model and Support vector machine, within the online news classification domain, providing very sensible result compared to existing methodologies.

Nowadays, since neural architecture is a hit in natural language processing, many neural networks appear in solving ABSA problem. LSTM has achieved many great successes in other NLP tasks such as machine translation, dependency parsing and sentence-level sentiment classification. However, single LSTM model cannot distinguish different sentiment polarity based on different aspects, for the feature vector of sentence with different aspects is the same. TD-LSTM and TC-LSTM (Tang et al., 2015) [3] achieved state-of-the-art performance in target-dependent sentiment classification. However, only the last output of LSTM was used in representing the sentence, which might loss some previous important features. ATAE-LSTM (Wang et al., 2016) [10] added attention mechanism to determine the weight of each hidden unit. IAN (Ma et al., 2017) [11] published a network using interactive attention between contexts and aspects. Thus, contexts and aspects can influence each other during learning.

3. Model

The overall architecture of our model is shown in Fig.1. It contains five parts: embedding tuning model, bi-GRU memory model, global attention model, local attention model and output model. Details will be discussed in the following separately. Suppose that the input sentence is $\mathrm{s} = \{s_1, \ldots, s_{\tau-1}, s_\tau, s_{\tau+1}, \ldots, s_T\}$, where T represents the length of the sentence and s_τ represents the aspect word. Our goal is to predict the sentiment polarity of the sentence based on the aspect word s_τ. In some cases, aspect word may be phrase-like words which is consist of several words, e.g. “Toshiba Warranty”. For simplicity, we just use the average of these words’ vectors as the aspect word vector.

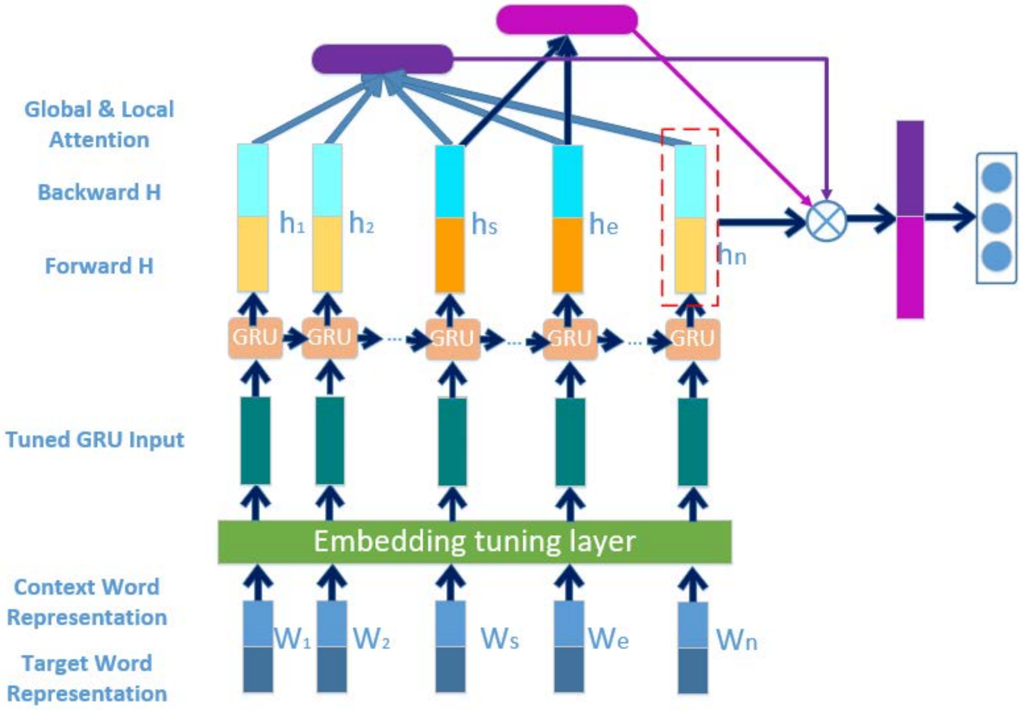


Fig. 1. The overall architecture of ETAN

3.1 Embedding Tuning

Let $\mathbb{L} \in \mathbb{R}^{d\times|V|}$ be a pre-trained embedding lookup table, where d is the dimension of word vectors and $|V|$ is the vocabulary size. GloVe (Pennington et al., 2014) [12] is used to pre-train. Since we want to capture more intrinsic information that is especially useful for ABSA task and expect that aspect word can influence context words as much as possible, aspect word and context word are concatenated together as the input of fine-tuning layer. Fine-tuning method is:

$$v_t = \tanh(W_{emb}(c_t, v_a)) \tag{1}$$

Where $c_t, v_a \in \mathbb{R}^d$ and $v_t \in \mathbb{R}^{2d}$. c_t is context word vector and v_a is the aspect word vector.

3.2 Bi-GRU for Memory Building

We use Multi-layer Bidirectional GRU to build the memory. At each time step t, both forward and backward GRU maintain a hidden state h_t^l, and a memory $\tilde{h}_t^l$. The update process as each time step t is as follows:

$$\vec{z}_t^l = \sigma(\vec{W}_z \vec{h}_t^{l-1} + \vec{U}_z \vec{h}_{t-1}^l) \tag{2}$$

$$\vec{r}_t^l = \sigma(\vec{W}_r \vec{h}_t^{l-1} + \vec{U}_r \vec{h}_{t-1}^l) \tag{3}$$

$$\tilde{h}_t^l = \tanh(\vec{W}_h \vec{h}_t^{l-1} + \vec{r}_t^l \circ \vec{U}_h \vec{h}_{t-1}^l) \tag{4}$$

$$h_t^l = \vec{z}_t^l \circ h_{t-1}^l + (1 - \vec{z}_t)\tilde{h}_t^l \tag{5}$$

Where σ and $tanh$are sigmoid and hyperbolic tangent functions, $\vec{W}_z$, $\vec{W}_r$, $\vec{W}_h \in \mathbb{R}^{\vec{d}_l \times \vec{d}_{l-1}}$, $\vec{U}_z$, $\vec{U}_r$,

$\vec{U}_h \in \mathbb{R}^{\vec{d}_l \times \vec{d}_l}$, $\vec{d}_l$ is the number of hidden cells at layer l of forward GRU, and $\circ$ is element-wise multiply. The gate $r, z \in \mathbb{R}^{\vec{d}_l}$ simultaneously control whether the memory will forget the past information and how much information of the memory matters now. The backward GRU is the same as forward one, except that the input sequence is reversed. Suppose that there are L layers stacked in the Bi-GRU, the final hidden state is $H^* = \{h_1^*, \dots, h_t^*, \dots, h_T^*\}$, where $h_t^* = (\vec{h}_t^L, \overleftarrow{h}_t^L) \in \mathbb{R}^{\vec{d}_L + \overleftarrow{d}_L}$.

3.3 Global Attention

h_T^* is the final hidden state of a sentence, and it can be used to present a sentence. However, only using h_T^* losses many information in the past hidden states. So, attention mechanism is implemented in our model. The attention mechanism will produce an attention weight vector α and weighted hidden representation r.

$$M = \tanh(\begin{bmatrix} W_h H \\ W_v[v_a, \dots, v_a] \end{bmatrix}) \tag{6}$$

$$\alpha = softmax(w^T M) \tag{7}$$

$$r = H\alpha^T \tag{8}$$

where $M \in \mathbb{R}^{(\vec{d}_L + \overleftarrow{d}_L + d) \times T} W_h \in \mathbb{R}^{(\vec{d}_L + \overleftarrow{d}_L) \times (\vec{d}_L + \overleftarrow{d}_L)}$, $H \in \mathbb{R}^{(\vec{d}_L + \overleftarrow{d}_L) \times T}$, $W_v \in \mathbb{R}^{d \times d}$, $[v_a, \dots, v_a] \in \mathbb{R}^{d \times T}$ and $w \in \mathbb{R}^{\vec{d}_L + \overleftarrow{d}_L + d}$. $[v_a, \dots, v_a]$ is a matrix repeatedly concatenating v_a for T times a. α is an attention vector and r is the weighted sentence representation.

3.4 Location Based Local Attention

In section Global Attention, global attention model is used to produce weighted sentence representation. However, since there are more than one aspects in the sentence, attention vector may have high weight on more than one aspects or several sentiment words. If a sentence has two aspects and they have opposite sentiment polarity (e.g. “drivers” and “BIOS update” in “Drivers updated ok but the BIOS update froze the system up and the computer shut down”), it is more likely that one of them will be predicted wrong. So, we add a location based local attention layer in our model. Specifically, we suppose that the closer a word is to the aspect, the higher its memory should be weighted. For each sentence with specific aspect, our model only focuses on serval words around the aspect. A window with size $\mathbb{s}$ is defined and length of local hidden state is $2\mathbb{s} + 1$. If there are punctuations in chosen hidden state, window will be shifted. We add a special symbol ——“.” to avoid out of index when shifting. Detailed algorithm is as follows:

```
        Input: window size $\mathbb{s}$,
               sentence representation
               $\{s_1, \dots, s_{\tau-1}, s_\tau, s_{\tau+1}, \dots, s_T\}$,
               aspect location $\tau$,
        Output: window start position $ws$,
                window end position $we$
--------------------------------------------------------------
  $ws = \tau, we = \tau$
        for i in $\mathbb{s}$
        if $s_{ws-1}$ is not punctuation, $s_0$ or EOS
              $ws = ws - 1$
        else break
        r = $\mathbb{s} - (\tau - ws)$
        for i in $\mathbb{s} + $r
        if $s_{we+1}$ is not punctuation, $s_0$ or EOS
              $we = we + 1$
           else break
        if $we - ws < 2\,\mathbb{s}$
        $a = (2\,\mathbb{s} - (we - ws))/2$
           $ws = ws - \lfloor a \rfloor, we = we + \lceil a \rceil$
```

Attention vector calculation is almost the same as global one.

$$M_l = \tanh(\begin{bmatrix} W_{hl}H_l \\ W_{vl}[v_a,\dots,v_a]_l \end{bmatrix}) \tag{9}$$

$$\alpha_l = softmax(w_l^T M_l) \tag{10}$$

$$r_l = H_l \alpha_l^T \tag{11}$$

In experiment, model with both global and local attention performs best. So, combining global and local attention mechanism, we concatenate sentence representation generated by global and local attention. Hope that our model can automatically learn the importance of global and local representation. The final sentence representation of specific aspect is given by:

$$h^* = \begin{bmatrix} \tanh(W_{pg}r + W_{xg}h_T^*) \\ \tanh(W_{pl}r_l + W_{xl}h_T^*) \end{bmatrix} \tag{12}$$

where $h^* \in \mathbb{R}^{2(\vec{d}_L + \overleftarrow{d}_L)}$, W_{pg}, W_{pl}, W_{xg} and W_{xl} are projection parameters to be learned during training. Inspired by Rocktäschel et al.(2015) [13], we find that performance will be better if h_T^* is added in final sentence representation.

3.5 Output and Training

In output layer, h^* is fed into a softmax layer and probability distribution is generated. The model is trained by minimizing cross entropy error with L_2 regularization term:

$$\mathcal{L} = \sum_{(x,y)\in D} \sum_{c\in C} y^c \log f^c(x;\theta) + \lambda \|\theta\|^2 \tag{13}$$

where $C = \{\text{positive}, \text{negative}, \text{nutural}\}$ is sentiment category set, D is the collection of training data, $y \in \mathbb{R}^C$ is a one-hot vector where the element of true sentiment is 1 and else elements are 0, $f^c(x;\theta)$ is predicted probability of sentiment c, λ is L_2 regularization weight and θ represents all parameters. Dropout is also used to prevent overfitting.

4. Experiment

4.1 Dataset and Experimental Setting

Our experiments are conducted on four datasets, as shown in Table 1. Restaurant and laptop reviews are from SemEval2014 (Pontiki et al., 2014) [14], which are widely used in previous works. Tweet dataset is a collection of twitters, which is collected by (Dong et al., 2014) [15]. BDCI dataset is a Chinese dataset in BDCI competition host by CCF. For data balance, we remove a few examples with "conflict" sentiment polarity, e.g., "Certainly not the best sushi in New York, however, it is always fresh" (Pontiki et al., 2014) [14].

Table 1. Statistics of all Datasets.

Dataset	Positive		Negative		Natural	
	Train	Test	Train	Test	Train	Test
Restaurant	2164	728	898	196	637	196
Laptop	987	340	866	128	460	169
Tweet	1411	173	1411	173	2826	346
BDCI	6715	2889	4033	1711	669	266

300-dimensional pre-trained word vector by GloVe (Pennington et al., 2014) [12] (whose vocabulary size is 1.9M^2) is used as original word representation, which is also widely used in other papers (Wang et al., 2016; Tang et al.,2016; Ma et al., 2017) [5-7]. For Chinese embedding, 300-dimensional pre-trained embedding by fastText (Grave et al., 2018) [16] is used. Table 1 shows the training and test instance numbers in each category. Thus, we can better conclude the capability of our model and compare our model with others at the same starting point.

4.2 Comparison to Other Methods

We compared our embedding tuning & attention model (ETAN) with our baseline and the following models, including LSTM, TD-LSTM, ATAE-LSTM, IAN:

- Baseline model: Baseline model only uses global attention, but not local attention and fine-tuning technique.
- LSTM: Standard LSTM cannot capture aspect information in sentence, so the sentence representation is always the same one, which means it always predict the same sentiment polarity given different aspects. Predictably, the performance of LSTM is not very good.
- TD-LSTM: TD-LSTM improves the performance by letting aspect word be the last input word, so that the hidden state contains the most information of aspect word. However, it does not use attention mechanism, so it cannot capture the importance of a word to the aspect word.
- ATAE-LSTM: ATAE-LSTM added attention mechanism into model and used aspect attention. Thus, the importance of each context word to aspect word can be calculated. However, only aspect attention is generated in the model, so ATAE-LSTM performs a little better than TD-LSTM.
- IAN: IAN increased prediction accuracy by fully using aspect words. Unlike previous models that just think of phrase-like aspect words as a single word, it used interactive attention between aspect words and context words, which better present aspect words and the relationship between two.

None of above models fine-tuned word embedding. However, sentiment analysis task is quite different from other NLP tasks, in which we want the embedding of opposite sentiment words to be as far away as possible. So, our model finely tunes word embedding based on pre-trained GloVe. Table 2 shows that the increase of performance of baseline+fine-tuning model has some contribution to the final result.

What's more, location information is also proved to be quite important in ABSA. Baseline+local_attention model shows significant improvements than baseline model in Table 2. So, global and location based local attention are both applied in our model.

Table 2. Comparison with other models. Accuracy and F1 score on 3-class prediction about SemEval 2014 Task 4, Tweet and BDCI. Best performances are in **bold.**

Dataset	Restaurant		Laptop		Tweet		BDCI	
	Acc	F1	Acc	F1	Acc	F1	Acc	F1
LSTM	74.2	70.2	66.4	62.8	62.2	58.4	75.5	70.3
TD-LSTM	75.6	74.0	68.1	66.9	66.4	63.3	80.4	75.6
ATAE-LSTM	76.9	75.5	68.8	67.8	66.8	64.2	86.8	84.2
IAN	77.2	76.8	**70.0**	69.2	67.5	66.8	87.2	84.9
Baseline	76.6	75.5	68.6	67.6	66.3	64.3	86.2	84.5
Baseline+local_attention	77.2	76.8	69.3	68.6	69.0	68.5	87.9	87.6
Baseline+fine-tuning	76.9	76.0	69.0	68.0	66.9	65.0	86.8	84.7
ETAN	**77.9**	**77.0**	69.9	**69.2**	**69.5**	**68.7**	**88.5**	**88.2**

Table 2 shows the performance comparison of ETAN and other baselines. TD-LSTM performs better than LSTM because it takes aspect into account, and ATAE-LSTM performs further better because attention mechanism is used. IAN outperforms ATAE-LSTM because it emphasizes the importance of aspects through learning aspect and context representation interactively.

ETAN model performs the best among all models in restaurant and tweet data, because we use the fine-tuned embedding and add location-based attention. Further, ETAN model improves the accuracy about 3.9% and 2.9% on tweet data compared with ATAE-LSTM and IAN respectively. As we know, it is difficult to boost 1 percent of accuracy on sentiment classification. The main reason may be that ETAN model has fine-tuning layer. Pre-trained embedding model GloVe is generated from Wiki, whose sentences are more formal and restrict than twitter sentences. So, GloVe embedding may not be a very good one for sentiment analysis on twitter data.

Main hyper-parameters of ETAN model are window size s and layer number L. For windows, we made experiments on 3, 4 and 5. Finally, 4 is chosen because it performs best in most datasets. Experiments are also done to determine layer number L. Since the more layers, the more accurate the aspect is expressed but the harder it is to find global optima, we made experiment on 1, 2 and 3, and 2 is chosen. Table 3 and table 4 show the details.

Table 3. Results on different window size. Best performances are in **bold.**

Window size	Restaurant	Laptop	Tweet	BDCI
3	77.3	69.1	69.1	87.8
4	**77.9**	**69.9**	69.5	**88.5**
5	77.7	69.0	**69.8**	86.8

Table 4. Results on different layer number. Best performances are in **bold.**

No. of GRU Layer	Restaurant	Laptop	Tweet	BDCI
ETAN-1L	77.6	69.3	69.4	87.6
ETAN-2L	**77.9**	69.9	**69.8**	**88.5**
ETAN-3L	77.8	**70.1**	69.2	88.0

4.3 Embedding Tuning Case Study

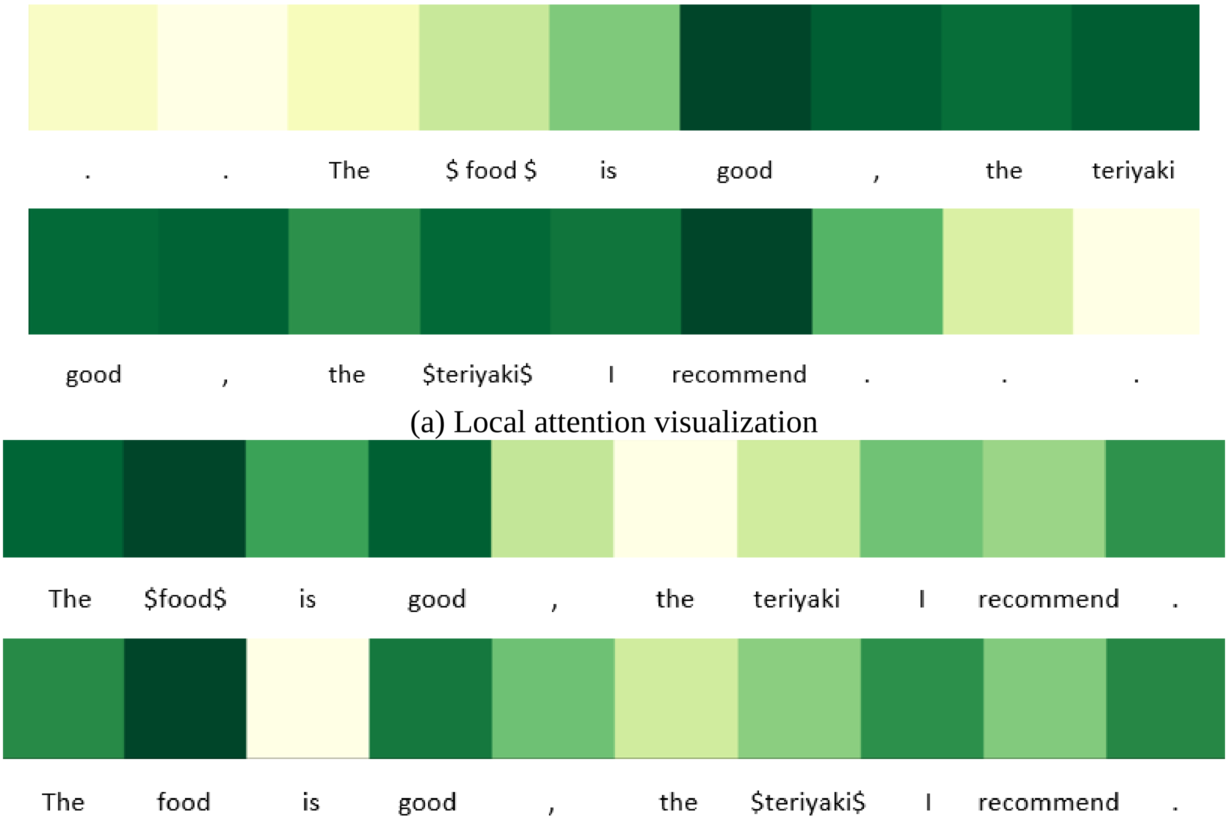
Table 5. Cosine similarity based on word “good”. Better performance after tuning are in **bold** in column 4.

Word	cos-s before tuning	cos-s after tuning	Similarity increasing percentage
bad	0.367488	0.131898	**-64.11%**
noisy	0.163984	-0.295815	**-280.39%**
crowded	0.149404	-0.299225	**-300.28%**
overpriced	0.163889	-0.231246	**-241.10%**
uncomfortable	0.182744	-0.240384	**-231.54%**
tasting	0.164978	0.231671	**40.43%**
enjoy	0.273898	0.233731	-14.66%
perfectly	0.265494	0.252856	-4.76%
lovely	0.262591	0.234845	-10.57%
wonderfully	0.226514	0.202873	-10.44%

In sentiment analysis tasks, we want the word vectors of the same polarity to be more similar and the word vectors of the opposite polarity more different. However, similarity between antonyms is always high, may be because they share the same POS (part of speech) and quite co occurs. Table 5 shows the cosine similarity between some opposite-sentiment-polarity words (the first five one) and same-sentiment-polarity words (the second five one). It shows that similarity of opposite-sentiment-polarity words decrease a lot after tuning, usually 200% to 300%, while similarity of same-sentiment-polarity words also decrease a little bit generally. Compared with so significant decrease of similarity on opposite-sentiment-polarity words, we assume that these little decreases on same-sentiment-polarity words will not influence results. All these comparisons are based on the word "good".

4.4 Attention Case Study

In this section, we first use "The food is good, the teriyaki I recommend" as a review context and two aspects "food" and "teriyaki" from restaurant category as case study. We apply ETAN to predict the sentiment polarity and get two correct results: positive and positive for two aspects respectively. Figure 2 (a) visualizes the local attention on "food" and "teriyaki" and Figure 2 (b) does global attention on "food" and "teriyaki". Since local attention cuts the sentence into paragraphs, interference information is generally decreased, (e.g. on aspect "food", "recommend" is not considered, which describes "teriyaki"). In most cases, global attention can weigh the words well, e.g. the sentence with "food" aspect. However, sometimes it does not generate weights well enough, e.g. the sentence with "teriyaki" aspect. Although "teriyaki" and "recommend" weigh more than other irrelevant words, "food" and "good" are still the most important. So, combining local and global attention, our model can maximum accuracy.

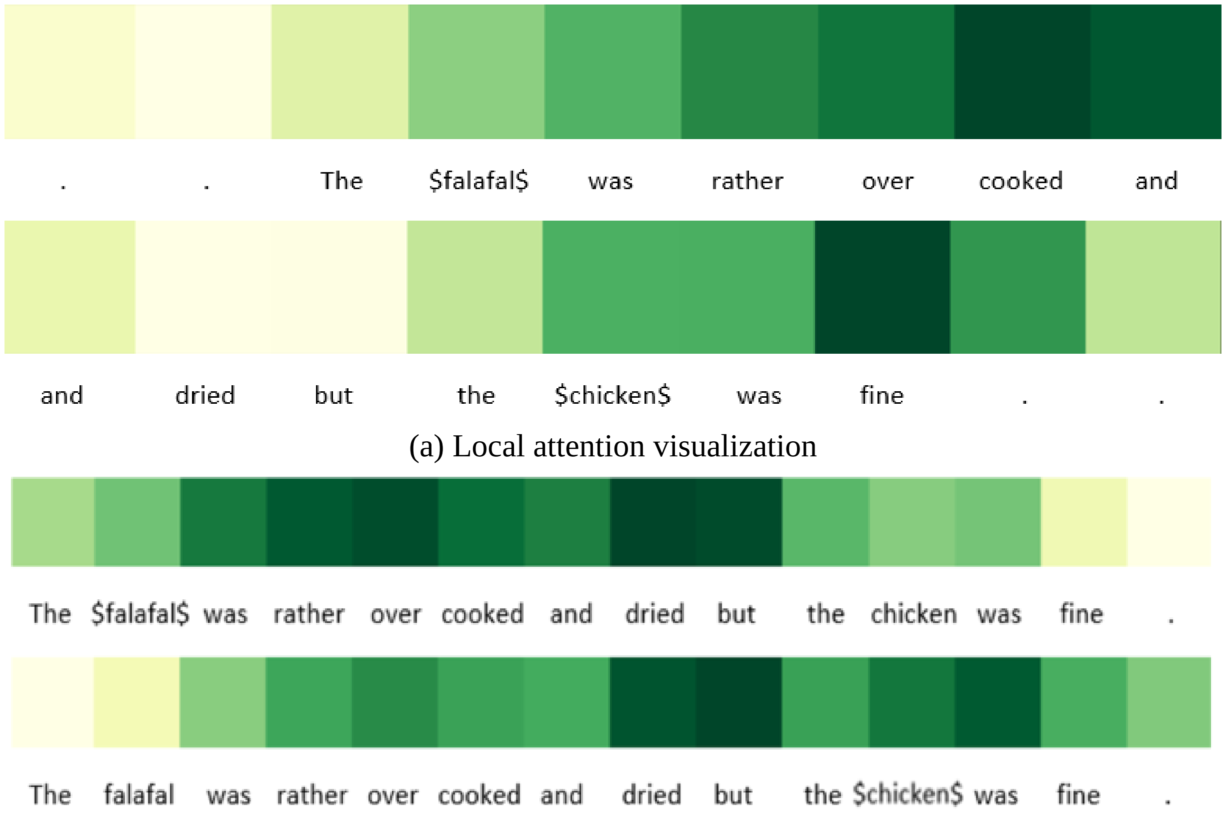


(a) Local attention visualization

(b) Global attention visualization

Fig. 2. Attention visualization of sentence "The food is good, the teriyaki I recommend". Aspect word is symbolled as xxx.

Further, we test a sentence with opposite sentiment polarity with different aspects, which is "The falafal was rather over cooked and dried but the chicken was fine" on "falafal" and "chicken". ETAN successfully predicts the correct results: negative and positive for two aspects respectively. It is noteworthy that "falafal" is not pre-trained in GloVe, so we randomized it. The correct prediction further supports the effect of embedding tuning. Figure 3 (a) and (b) visualizes the local and global attention respectively. Local attention model weighs the words quite well on both aspects, while global attention put more focus on "dried" than on "fine" when "chicken" is the aspect. Thus, through ETAN, we can well present a sentence with specific aspects.



(a) Local attention visualization

(b) Global attention visualization

Fig. 3. Attention visualization of sentence "The falafal was rather over cooked and dried but the chicken was fine". Aspect word is symbolled as xxx.

5. Conclusion

In this paper, we propose a neural network combining embedding tuning and attention for ABSA. The key contributions of the model are tuning embedding so that they fit more for ABSA task and adding location attention in the model to highlight aspect-related information. Experiments show that the proposed model obtains superior performance over the baseline models.

Acknowledgments

This research work has been funded by the National Natural Science Foundation of China (Grant No.61772337, U1736207 and 61472248), the SJTU-Shanghai Songheng Content Analysis Joint Lab, and program of Shanghai Technology Research Leader (Grant No.16XD1424400).

References

[1] Appel, O., Chiclana, F., Carter, J., & Fujita, H. (2017). Successes and challenges in developing a hybrid approach to sentiment analysis. Applied Intelligence (6), 1-13.

[2] Richard Socher, Jeffrey Pennington, Eric H. Huang, Andrew Y. Ng, and Christopher D. Manning. 2011. Semi-supervised recursive autoencoders for predicting sentiment distribu-tions. Proceedings of the 2011 Conference on Empirical Methods in Natural Language Pro-cessing, EMNLP 2011, pages 151–161.

[3] Duyu Tang, Bing Qin, Xiaocheng Feng, and Ting Liu. 2015. Target-dependent sentiment classification with long short-term memory. CoRR, abs/1512.01100.

[4] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. CoRR, abs/1409.0473.

[5] Wang, Y.; Huang, M.; Zhu, X.; and Zhao, L. 2016. Attention-based LSTM for aspect-level sentiment classification. Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016, 606–615.

[6] Duyu Tang, Bing Qin, and Ting Liu. 2016. Aspect level sentiment classification with deep memory network. Proceedings of the 2016 Conference on Empirical Methods in Natural Lan-guage Processing.

[7] Ma, D., Li, S., Zhang, X., Wang, H., Ma, D., & Li, S., et al. (2017). Interactive Attention Networks for Aspect-Level Sentiment Classification. Twenty-Sixth International Joint Con-ference on Artificial Intelligence (pp.4068-4074).

[8] Tetsuya Nasukawa and Jeonghee Yi. 2003. Sentiment analysis: Capturing favorability using natural language processing. In Proceedings of the 2nd international conference on Knowledge capture, pages 70–77. ACM.

[9] Saif M Mohammad, Svetlana Kiritchenko, and Xiaodan Zhu. 2013. Nrc-canada: Building the state-of-the-art in sentiment analysis of tweets. arXiv preprint arXiv:1308.6242.

[10] Wang, Y.; Huang, M.; Zhu, X.; and Zhao, L. 2016. Attention-based LSTM for aspect-level sentiment classification. Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016, 606–615.

[11] Ma, D., Li, S., Zhang, X., Wang, H., Ma, D., & Li, S., et al. (2017). Interactive Attention Networks for Aspect-Level Sentiment Classification. Twenty-Sixth International Joint Con-ference on Artificial Intelligence (pp.4068-4074).

[12] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vec-tors for word representation. Empirical Methods in Natural Language Processing (EMNLP), pages 1532–1543.

[13] Xiaodan Zhu, Parinaz Sobhani, and Hongyu Guo. 2015. Long short-term memory over tree structures. ICML.G. Eason, B. Noble, and I. N. Sneddon, "On certain integrals of Lipschitz-Hankel type involving products of Bessel functions," Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955. (References)

[14] Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopou-los, and Suresh Manandhar. 2014. Semeval-2014 task 4: Aspect based sentiment analysis. Proceedings of the 8th international workshop on semantic evaluation (SemEval 2014), pages 27–35.

[15] Li Dong, Furu Wei, Chuanqi Tan, Duyu Tang, Ming Zhou, and Ke Xu. 2014. Adaptive recur-sive neural network for target-dependent twitter sentiment classification. Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Pa-pers).

[16] E. Grave, P. Bojanowski*, P. Gupta, A. Joulin, T. Mikolov, Learning Word Vectors for 157 Languages*