Hierarchical Attention Based Position-aware Network for Aspect-level Sentiment Analysis

Lishuang Li, Yang Liu and **AnQiao Zhou**
School of Computer Science and Technology, Dalian University of Technology
lilishuang314@163.com

Abstract

Aspect-level sentiment analysis aims to identify the sentiment of a specific target in its context. Previous works have proved that the interactions between aspects and the contexts are important. On this basis, we also propose a succinct hierarchical attention based mechanism to fuse the information of targets and the contextual words. In addition, most existing methods ignore the position information of the aspect when encoding the sentence. In this paper, we argue that the position-aware representations are beneficial to this task. Therefore, we propose a hierarchical attention based position-aware network (HAPN), which introduces position embeddings to learn the position-aware representations of sentences and further generate the target-specific representations of contextual words. The experimental results on SemEval 2014 dataset show that our approach outperforms the state-of-the-art methods.

1 Introduction

Aspect-level sentiment analysis is a fine-grained task in sentiment analysis, which aims to identify the sentiment polarity (i.e., negative, neutral, or positive) of a specific opinion target expressed in a comment/review by a reviewer. For example, given a sentence "The *price* is reasonable although the *service* is poor", the sentiment polarity for aspects *"price"* and *"service"* are positive and negative respectively.

Traditional methods for aspect-level sentiment analysis mainly focus on designing a set of features (such as bag-of-words, sentiment lexicons, and linguistic features) to train a classifier for sentiment classification (Kiritchenko et al., 2014; Wagner et al., 2014; Vo and Zhang, 2015). However, such kind of feature engineering work often relies on human ingenuity, which is a time-consuming process and lacks generalization. In recent years, more and more neural network based models have been proposed and obtained the state-of-the-art results (Wang et al., 2016; Tang et al., 2016a;2016b; Chen et al,. 2017; Ma et al., 2017; Tay et al., 2017; Zheng et al., 2018; Huang et al., 2018).

As previous research (Jiang et al., 2011) reveals that 40% of sentiment classification errors are caused by not considering targets in sentiment classification, recent works tend to focus on fusing the information of the targets and the contexts. Wang et al. (2016) and Tang et al. (2016a) both concatenated the aspect embeddings and embeddings of each word as inputs to a LSTM based model so as to introduce the information of the target into the model. Tay et al. (2017) adopted circular convolution and circular correlation to model the similarity between aspect and contextual words. Ma et al. (2017) and Zheng et al. (2018) both employed a bidirectional attention operation to achieve the representations of targets and contextual words determined by each other. Huang et al. (2018) introduced an attention-over-attention based network to model the aspects and contexts in a joint way and explicitly capture the interaction between aspects and the context.

As described above, the existing studies show that the interactions between aspects and the context are important to the aspect-level sentiment analysis. Leveraging this idea, we also propose a succinct hierarchical attention based mechanism to fuse the information of targets and the contextual words, which aims to generate the target-specific representations of each word.

In addition, most of the above methods ignore the position information of the aspect when

Proceedings of the 22nd Conference on Computational Natural Language Learning (CoNLL 2018), pages 181–189
Brussels, Belgium, October 31 - November 1, 2018.

encoding the sentence. We argue that the position of a candidate aspect is important for the sentence modelling. For instance, consider the sentence "I bought a mobile phone, its *camera* is wonderful but the *battery life* is a bit short". For the candidate aspect *"battery life"*, *"wonderful"* and *"short"* are both likely to be considered as its adjunct word. In this case, if we encode the position information into the representation of each word effectively, we would have more confidence in concluding that the *"short"* is the adjunct word of *"battery life"* and predict the sentiment as negative. Then, the next problem is how to introduce the position information. In some previous works (Tang et al., 2016b; Chen et al,. 2017), they weighted the representation of each word according to the position, and the words close to the aspect could be paid more attention. However, this operation is not always reasonable and sometimes the adjunct word may be far away from the target word. Thus, we introduce position embeddings when modelling the sentence and further generate the position-aware representations. In other words, the position information is considered as a kind of features and embedded into position embeddings. The model will learn to exploit both of the semantic information and the position clues.

Based on the analysis above, in this paper, we propose a hierarchical attention based position-aware network (HAPN) for aspect-level sentiment classification. A position-aware encoding layer is introduced for modelling the sentence to achieve the position-aware abstract representation of each word. On this basis, a succinct fusion mechanism is further proposed to fuse the information of aspects and the contexts, achieving the final sentence representation. Finally, we feed the achieved sentence representation into a softmax layer to predict the sentiment polarity.

We evaluate our approach on SemEval 2014 dataset (Pontiki et al., 2014), containing reviews of restaurant domain and laptop domain. The experimental results demonstrate that the proposed approach is effective for aspect-level sentiment classification, and it outperforms state-of-the-art approaches with remarkable gains. We make our source code public at `https://github.com/DUT-LiuYang/Aspect-Sentiment-Analysis`.

2 Related Work

Many approaches have been proposed to address the problem of aspect-level sentiment analysis. Traditional approaches to this task normally exploited a diverse set of strategies to convert classification clues (i.e., sentiment lexicons, bag-of-word) into feature vectors (Kiritchenko et al., 2014; Wagner et al., 2014; Vo and Zhang, 2015). Although these methods have achieved comparable performance, their models highly depend on the effectiveness of the handcraft features which are labor intensive and lack generalization.

Therefore, many neural network based models have been proposed in recent years. And most current state-of-the-art works in aspect-based sentiment analysis pay more attention to fusing the information of the targets and contextual words. Wang et al., (2016) proposed an attention based LSTM which introduced the aspect clues by concatenating the aspect embeddings and the word representations. Tang et al. (2016a) developed two target-dependent LSTM to model the left and right contexts with target, where the target information was automatically taken into account. Tay et al. (2017) proposed an attention based LSTM which learned to attend based on associative relationships between sentence words and aspect by adopting circular convolution and circular correlation. Ma et al. (2017) proposed an interactive attention network which interactively learned attentions in the contexts and targets. Similarly, Zheng et al. (2018) introduced a rotatory attention mechanism to achieve the representations of the targets, the left context and the right context, which were determined by each other. Huang et al. (2018) introduced an attention-over-attention network modeled the aspects and sentences in a joint way, which jointly learned the representations for aspects and sentences and automatically focused on the important parts in sentences. In addition, other current researches focus on capturing more accurate information by adopting multiple attentions. Tang et al. (2016b) designed a deep memory network which consisted of multiple computational layers, each of which was an attention model over an external memory. Chen et al. (2017) proposed a recurrent attention based network which introduced multiple attention mechanisms.

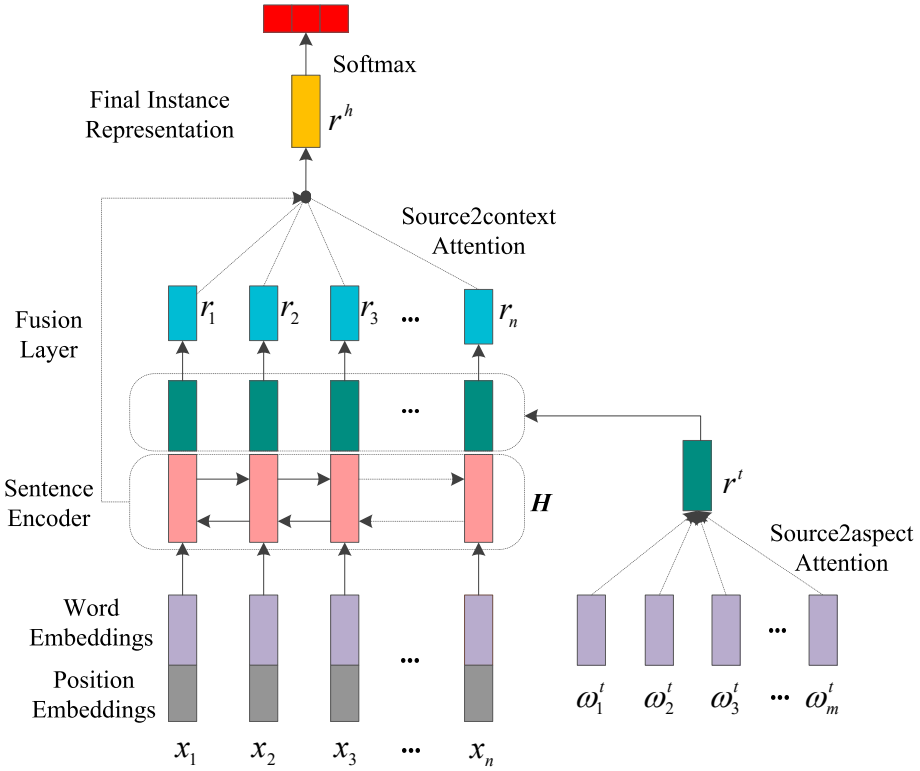

Figure 1: The architecture of the proposed model.

Compared with the above models, we introduce position embeddings when modelling the sentence to generate position-aware representations; on this basis, we propose a hierarchical attention based fusion mechanism to fuse the clues of aspects and the contexts.

3 Model

In our approach, each target along with the sentence where the target is located constitutes an instance. We suppose that a sentence consists of n words $\boldsymbol{w} = \{w_1, w_2, \cdots, w_n\}$ and a target has m words $\boldsymbol{w^t} = \{w_1^t, w_2^t, \cdots, w_m^t\}$. $\boldsymbol{w^t}$ is a sub-sequence of $\boldsymbol{w}$. The goal of our model is to predict the sentiment polarity of the sentence over the target.

As shown in Figure 1, our model primarily includes four parts: input embeddings, Bi-GRU based encoding layer, hierarchical attention based fusion layer and the output layer.

3.1 Input Embedding

The embedding layer has two parts: the word embeddings and the position embeddings. Let $W_w \in \mathbb{R}^{d_w \times v_w}$ be a word embedding lookup table generated by an unsupervised method such as GloVe (Pennington et al., 2014) or CBOW (Mikolov et al., 2013), where d_w is the dimension of the word embeddings and v_w is the size of word vocabulary. As described in Section 1, we also introduce position embeddings, which have been widely used in CNN based models, as a part of the inputs to the model. Similar as the word embedding layer, the position embedding layer is a $W_p \in \mathbb{R}^{d_p \times v_p}$, where d_p is the dimension of the position embeddings and v_p is the number of possible relevant positions between each word and the target. The position embedding lookup table is initialized randomly and tuned in the training phase.

3.2 Bi-GRU Based Sentence Encoder

In this paper, we apply a Bi-GRU (Cho et al., 2014) to learn a more abstract representation of the sentence. In the following, we describe our encoding layer in detail.

In the encoding phase, we first transform each token w_i in the sentence into a real-valued vector x_i using the concatenation of the following vectors:

- The pre-trained word embeddings ω_i of w_i.
- The position embeddings of w_i: the relevant position between the i-th word and the target is defined as the relative offset with respect

to the target and calculated by the follow equation:

$$\begin{cases} i-k & i<k \\ i-k-m & n \geq i > k+m \\ 0 & k+m \geq i \geq k \end{cases} \quad (1)$$

where k is the index of the first word of target, m is the length of the target, n is the length of the sentence, and *none* is the special marks assigned to the token padded. The position embedding vector is obtained by looking up the randomly initialized embeddings table according to the relevant position.

Hence a sequence of words can be represented as $X = \{x_1, x_2, \dots, x_n\}$. We then run two parallel GRU layers: forward GRU layer and backward GRU layer. We run the forward GRU to generate the hidden representation $(\vec{h}_1, \vec{h}_2, \dots, \vec{h}_n)$ and run the backward GRU to get the hidden representation $(\overleftarrow{h}_1, \overleftarrow{h}_2, \dots, \overleftarrow{h}_n)$. Eventually, we obtain the new representation $\boldsymbol{H} = (h_1, h_2, \dots, h_n)$ by concatenating the hidden vectors in $(\vec{h}_1, \vec{h}_2, \dots, \vec{h}_n)$ and $(\overleftarrow{h}_1, \overleftarrow{h}_2, \dots, \overleftarrow{h}_n)$: $h_i = [\vec{h}_i, \overleftarrow{h}_i]$. Note that $h_i \in \mathbb{R}^{2d_h}$ essentially encapsulates the context information over the whole sentence (from 1 to n) with a greater focus on position i, where d_h is the dimension of hidden states. Due to the introduction of the position embeddings, h_i is considered to be position-aware.

3.3 Hierarchical Attention Based Fusion Layer

In this subsection, we illustrate the proposed succinct mechanism to fuse the information of targets and the contextual words. In detail, a source2aspect attention is first employed to capture the most important clues in the target words and the representation of the aspect is obtained. Subsequently, an aspect-specific representation of each word is generated based on the aspect representation and the encoded position-aware representation. A source2context attention is then used to capture the most indicative sentiment words in the context and generate the weighted sum embeddings as the final sentence representation.

Source2aspect Attention: Due to the fact that substantial numbers of aspects contain at least two words (Zheng et al., 2018), we introduce a source2aspect mechanism to generate the representation of the aspect. The source2* attention is inspired by the related research of self-attention network (Shen et al., 2017). First, we introduce a score function by taking the word embeddings of each word in target as inputs.

$$f_t(\omega_i^t) = tanh(W_t \cdot \omega_i^t) \quad (2)$$

where $W_t \in \mathbb{R}^{d_w}$ is a weight vector and *tanh* is a non-linear function. The score f_t is then used as a weight denoting the importance of a word in the target. On this basis, the normalized importance weight of i-th word in the target α_i^t is computed as follows:

$$\alpha_i^t = \frac{exp\left(f_t(\omega_i^t)\right)}{\sum_{j=1}^{m} exp\left(f_t\left(\omega_j^t\right)\right)} \quad (3)$$

At last, a weighted combination of word embeddings is considered as the representation for the target:

$$r^t = \sum_{i=1}^{m} \alpha_i^t \cdot \omega_i^t \quad (4)$$

Information Fusion: After achieving the target representation, we then further make use of the achieved representation to construct the target-specific representation of each word in the sentence by the following equation:

$$r_i = W_s \cdot [h_i, r^t] \quad (5)$$

where $W_s \in \mathbb{R}^{(2d_h+d_w)\times d_w}$ is a weight matrix. r_i denotes the target-specific representation of the i-th word w_i in the sentence.

Source2context Attention: Then, the target-specific representation of each word is used to learn attentions and further generate the final sentence representation. The attention is defined as the following equations:

$$f_s([r_i, h_i]) = tanh(W_c \cdot [r_i, h_i]) \quad (6)$$

$$\alpha_i = \frac{exp(f_s([r_i,h_i]))}{\sum_{j=1}^{n} exp(f_s([r_i,h_i]))} \quad (7)$$

where $W_c \in \mathbb{R}^{2d_h+d_w}$ is a weight vector and α_i denotes the importance of the i-th word in the sentence.

At last, a weighted combination of position-aware hidden states is computed:

$$r^h = \sum_{i=1}^{n} \alpha_i \cdot h_i \quad (8)$$

which is considered as the final representation of the current instance.

Dataset	Positive		Neutral		Negative	
	Train	Test	Train	Test	Train	Test
Restaurant	2164	728	633	196	805	196
Laptop	987	341	460	169	866	128

Table 1: Statistics of SemEval 2014 Dataset.

3.4 Output and Model Training

Hence, we can get the final representation r^h of the current instance after the last three subsections. Then we feed it into a softmax layer to predict the target sentiment.

Given all of our (suppose N) training samples $(x^{(i)}; y^{(i)})$, we can then define the loss function as the negative log-likelihood:

$$\mathcal{L}(\theta) = -\sum_{j=1}^{N} log\, p\left(y^{(j)} \mid x^{(j)}, \theta\right) \quad (9)$$

In order to compute the network parameter θ, we minimize the average negative log-likelihood $\mathcal{L}(\theta)$ via RMSprop proposed by Tieleman and Hinton (2012) over shuffled mini-batches. We also adopt the dropout regularization (Zaremba et al., 2014) and early stopping to ease overfitting.

4 Experiments

4.1 Experiment Settings

We conduct experiments on SemEval 2014 Task 4 to validate the effectiveness of our model, as shown in Table 1. The SemEval 2014 dataset contains reviews of restaurant and laptop domains, which are widely used in previous works. The evaluation metric is classification accuracy.

We use 300-dimension word vectors pre-trained by GloVe (Pennington et al., 2014) (whose vocabulary size is 1.9M) for our experiments, as previous works did (Tang et al., 2016b; Chen et al., 2017; Zheng et al., 2018). All out-of-vocabulary words are initialized as zero vectors, and all biases are set to zero. The dimensions of hidden states and fused embeddings are set to 300. The dimension of position embeddings is set to 50. Keras is used for implementing our neural network model. In model training, we set the learning rate to 0.001, the batch size to 64, and dropout rate to 0.5. The paired *t*-test is used for the significance testing.

4.2 Compared Methods

In order to evaluate the performance of proposed model, we select the following state-of-the-art methods for comparison:

- **Majority** assigns the sentiment polarity with most frequent occurrences in the training set to each sample in test set.
- **Bi-LSTM** and **Bi-GRU** adopt a Bi-LSTM and a Bi-GRU network to model the sentence and use the hidden state of the final word for prediction respectively.
- **TD-LSTM** adopts two LSTMs to model the left context with target and the right context with target respectively (Tang et al., 2016a); It takes the hidden states of LSTM at last time-step to represent the sentence for prediction.
- **MemNet** (Tang et al., 2016b) applies attention multiple times on the word embeddings, and the output of last attention is fed to softmax for prediction.
- **IAN** (Ma et al., 2017) interactively learns attentions in the contexts and targets, and generates the representations for targets and contexts separately.
- **RAM** (Chen et al., 2017) is a multilayer architecture where each layer consists of attention-based aggregation of word features and a GRU cell to learn the sentence representation.
- **LCR-Rot** (Zheng et al., 2018) employs three Bi-LSTMs to model the left context, the target and the right context. Then they propose a rotatory attention mechanism which models the relation between target and left/right contexts.
- **AOA-LSTM** (Huang et al., 2018) introduces an attention-over-attention (AOA) based network to model aspects and sentences in a joint way and explicitly capture the interaction between aspects and context sentences.

4.3 System Performance Comparision

Table 2 shows the performance comparison of our method with the state-of-the-art methods on the same test dataset. From the table, we make the following observations:

(1) As shown in the table, we can clearly observe that the **Majority** method is the worst, which means the majority sentiment polarity occupies 65.0% and 53.45% of all samples on the *Restaurant* and *Laptop* corpus respectively. In addition to MemNet, all the other models are RNN based models and better than the **Majority** method. This indicates that RNN based models can obtain better representations of sentence automatically

Dataset	Restaurant (%)	Laptop (%)
Majority	65.00	53.45
Bi-LSTM	78.57	70.53
Bi-GRU	80.27	73.35
TD-LSTM	75.63*	68.13*
MemNet	79.98	70.33
IAN	78.60	72.10
RAM	80.23	74.49
LCR-Rot	81.34	75.24
AOA-LSTM	81.20	74.50
HAPN	***82.23***	***77.27***

Table 2: Comparison with baselines on SemEval 2014 dataset. The results with * are retrieved from MemNet paper.

without manual feature engineering and improve the performance in this task.

(2) The **TD-LSTM** model, which has been shown to be better than LSTM (Tang et al., 2016a), gets the worst performance of all RNN based models and the accuracy achieved by **TD-LSTM** is 2.94% and 2.4% lower than those by **Bi-LSTM** on the two datasets respectively. This results show that introducing target clues only by splitting the sentence according to the position of target is inadequate and bidirectional RNN based model can achieve better performance than unidirectional model in this task. Another noticeable observation is that **Bi-GRU** achieves 80.27% and 73.35% accuracies which are 1.7% and 2.82% higher than those of **Bi-LSTM** on the *Restaurant* and *Laptop* dataset respectively. It indicates that Bi-GRU is more suitable to this task than Bi-LSTM.

(3) Compared with the state-of-the-art methods, our model achieves the best performance, which illustrates the effectiveness of the proposed approach. Our method achieves accuracies of 82.23% as well as 77.27% on the *Restaurant* and *Laptop* dataset respectively, which are 0.89% and 2.03% higher than the current best method. We will give a detailed analysis in the following subsections.

4.4 Effects of Position Embeddings

In order to verify the efficiency and advantage of position embeddings, we design the following models:

Bi-GRU employs the standard Bi-GRU to encode the sentence and predict the sentiment polarity.

Dataset	Restaurant (%)	Laptop (%)
Bi-GRU	80.27	73.35
Bi-GRU-PW	79.55	71.94
Bi-GRU-PE	***80.89***	***76.02***

Table 3: The performance of models with different strategies to introduce position information.

Dataset	Restaurant (%)	Laptop (%)
HAPN	***82.23***	***77.27***
No-fusion	81.88	76.49

Table 4: The performance of models with or without fusion operation.

Bi-GRU-PW first weights the word embeddings of each word in the sentence based on the distance from the target, as did in (Tang et al., 2016b; Chen et al,. 2017). Then the weighted representations are fed into the Bi-GRU.

Bi-GRU-PE concatenates the word embeddings and the position embeddings of each word as inputs to the Bi-GRU when modelling the sentence.

In Table 3, we report the performance of the three models. It can be observed that **Bi-GRU-PE** performs better than **Bi-GRU** significantly. After introducing the position embeddings, the accuracy has an increase of 0.62% and 2.67% on two datasets. This indicates that exploiting the position clues effectively can improve the performance of models in this task. In addition, another observation is that **Bi-GRU-PW** performs even worse than **Bi-GRU**. The accuracy achieved by **Bi-GRU-PW** is 0.72% as well as 1.41% lower than that by **Bi-GRU** on the *Restaurant* and *Laptop* dataset respectively. To an extent, the results verify that weighting the word representations according to the distance to the aspect is ineffective in this task.

4.5 Effects of the Information Fusion

To verify the efficiency of the information fusion, we further design the following model for comparison:

No-fusion is a simplified version of **HAPN**, where we directly concatenate the target representation and the position-aware representation of each word as the inputs to the source2context attention.

In Table 4, we report the performance comparison of **HAPN** and **No-fusion**. From the Table, we can observe that **HAPN** performs better

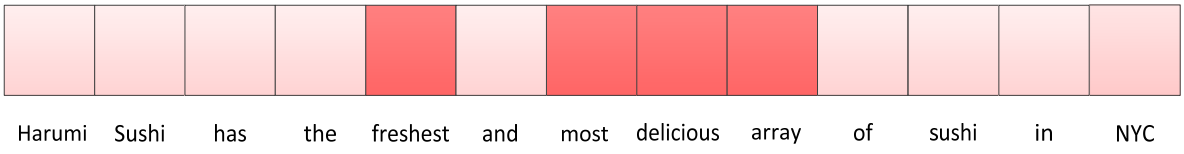

Figure 2: Attention visualizations of an example sentence.

than **No-fusion**. **HAPN** achieves improvement of 0.35% and 0.78% on accuracy respectively on the two dataset. It indicates that the fusion operation we propose has potentials in automatically generating target-specific representations and improves the performance.

4.6 Effects of The Hierarchical Attention

This subsection evalutes the effectiveness of the hierarchical attention mechanism. To achieve this goal, we deactivate the two attention respectively from the proposed model.

Firstly, to verify the efficiency of the Source2aspect attention, we design the following model for comparison:

No-S2A-attention is a simplified version of **HAPN**, where the Source2aspect attention is replaced with averaging the initial word embeddings to represent the target phrase.

Table 5 presents the performance comparison of **HAPN** and **No-S2A-attention**. From Table 5, we can see that **No-S2A-attention** achieve the accuracies of 81.34% as well as 76.49% on the Restaurant and Laptop dataset respectively, which are 0.89% and 0.78% lower than the proposed model. This indicates that the Source2aspect attention in our model is effective to this task.

Secondly, as described in the privious sections, the Source2context attention in the paper aims at weighted summing the position-aware hidden states based on the target-specific representations generated by the information fusion operation. From Figure 1, it could be observed that: (1) The information fusion operation is only used to calculate the Source2context attention value. (2) The output of Source2aspect attention is only used for information fusion.

Dataset	Restaurant (%)	Laptop (%)
HAPN	***82.23***	***77.27***
No- S2A-Attention	81.34	76.49

Table 5: The performance of models with or without Source2aspect attention.

Therefore, we remove the fusion operation and Source2aspect attention while removing the Source2context attention. And the achieved model is "Bi-GRU-PE" reported in the Table 3, achieving the accuracies of 80.89% and 76.02% on the two datasets respectively, which are 1.34% and 1.25% lower than the proposed model. This indicates that the Source2context attention is necessary in the proposed model.

4.7 Case Study

In this section, we use a review sentence "Harumi Sushi has the freshest and most delicious *array of sushi* in NYC" and the target *"array of sushi"* from the Restaurant dataset as a case study. We apply our HAPN to model the sentence and the target, and obtain the correct sentiment polarity: *positive*. In Figure 2, we give the visualization of the attention weights (Source2context) on this sentence computed by HAPN.

The meaning of the example sentence in the case study is that the *"array of sushi"* is good. Obviously, the words *"freshest"* and *"most delicious"* play an important role in judging the sentiment polarity of *"array of sushi"*. From Figure 2, we can observe that those words are paid much attention as we expect. And it is worth noting that the word *"freshest"* obtains as much attention as *"delicious"*, although *"freshest"* is much farther from the target than *"delicious"*. This shows that our model doesn't reduce a word's weight only according to the long distance from the target. This may be because that our HAPN embeds the position information and can consider the influence of position in combination with semantic information instead of simply weighting.

5 Discussion

Experimental results show that our proposed method has better performance than state-of-the-art approaches. The detailed analysis for improvement is as follows:

(1) Position embeddings

As discussed in Section 1, position information is important when modelling the sentence. When there are several aspects in a sentence, it is easy to pay attention to the adjectives of another aspect by error. In this case, the relevant position between a word and the target can help to understand the structure of sentences. We introduce position embeddings as a part of inputs when modelling the sentence. Therefore, we can achieve the position-aware representations of each word and the model will learn to exploit both the semantic information and the position clues of each word. As shown in Table 3, the introduction of position embeddings bring a performance improvement of 0.62% and 2.67% on two datasets respectively, which illustrates the effectiveness of the position embeddings.

(2) Hierarchical attention based fusion operation

Compared with the traditional sentiment analysis task, the aspect-level sentiment analysis is more fine-grained and need the information of specific target. As described in Section 3.3, we introduce a hierarchical attention based fusion layer to generate the target-specific representation of each word. By exploiting the specific representations to further compute the attention value and generate the final sentence representation, the model can obtain more target clues. As shown in Table 4 and Table 5, the experimental results show that the hierarchical attention based information fusion operation can bring performance improvement to this task.

(3) Bi-GRU based encoder

In this paper, we employ a Bi-GRU based encoder to model the sentence. GRU has been shown to achieve comparable performance with less parameters than LSTM (Chung et al., 2014; Jozefowicz et al., 2015). And we run two parallel GRUs to obtain richer semantic information and position clues. The experimental results in Table 2 also show that Bi-GRU can achieve better performance in this task.

6 Conclusions

In this paper, we propose a hierarchical attention based position-aware network for aspect-level sentiment analysis. This architecture introduces position embeddings as a part of inputs to further generate position-aware representations. Furthermore, we propose a succinct hierarchical attention based mechanism to fuse the information of targets and the contextual words, and achieve the final sentence representation. Experimental results show that our approach achieves state-of-the-art performance on the Semeval 2014 dataset.

Acknowledgments

This paper is supported by the National Natural Science Foundation of China under NO.61672126. We thank anonymous reviewers for their valuable comments.

References

Peng Chen, Zhongqian Sun, Lidong Bing, and Wei Yang. 2017. Recurrent attention network on memory for aspect sentiment analysis. In *Proceedings of EMNLP*, pages 463-472.

Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *Computer Science*.

Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*.

Binxuan Huang, Yanglan Ou and Kathleen M. Carley. 2018. Aspect Level Sentiment Classification with Attention-over-Attention Neural Networks. *arxiv preprint arXiv:1804.06536*.

Long Jiang, Mo Yu, Ming Zhou, Xiaohua Liu, and Tiejun Zhao. 2011. Target-dependent twitter sentiment classification. In *Proceedings of ACL: Human Language Technologies-Volume1*, pages 151-160.

Rafal Jozefowicz, Wojciech Zaremba, and Ilya Sutskever. 2015. An empirical exploration of recurrent network architectures. In *ICML*, pages 2342-2350.

Svetlana Kiritchenko, Xiaodan Zhu, Colin Cherry, and Saif Mohammad. 2014. Nrc-canada-2014: Detecting aspects and sentiment in customer reviews. In *Proceedings of SemEval*, pages 437-442

Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. 2017. Interactive attention networks for aspect-level sentiment classification. In *Proceedings of IJCAI*, pages 4068-4074.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111-3119.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word

representation. In *Proceedings of EMNLP*, pages 1532-1543.

Maria Pontiki, Dimitrios Galanis, John Pavlopoulos, Haris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. Semeval-2014 task 4: Aspect based sentiment analysis. In *SemEval@COLING*, pages 27-35, Stroudsburg, PA, USA. Association for Computational Linguistics.

Tao Shen, Tianyi Zhou, Guodong Long, Jing Jiang, Shirui Pan, and Chengqi Zhang. 2017. DiSAN: Directional Self-Attention Network for RNN/CNN-Free Language Understanding. *arXiv preprint arXiv: 1709.04696*.

Duyu Tang, Bing Qin, Xiaocheng Feng, and Ting Liu. 2016a. Effective lstms for target-dependent sentiment classification. In *Proceedings of COLING*, pages 3298-3307.

Duyu Tang, Bing Qin, and Ting Liu. 2016b. Aspect level sentiment classification with deep memory network. In *Proceedings of EMNLP*, pages 214-224.

Yi Tay, Anh Tuan Luu, and Siu Cheung Hui. 2017. Learning to attend via word-aspect associative fusion for aspect-based sentiment analysis. *arXiv preprint arXiv:1712.05403*.

Tijmen Tieleman and Geoffrey Hinton. 2012. Lecture 6.5-rmsprop. In COURSERA: *Neural networks for machine learning.*

Duy-Tin Vo and Yue Zhang. 2015. Target-dependent twitter sentiment classification with rich automatic features. In *Proceedings of IJCAI*, pages 1347-1353.

Joachim Wagner, Piyush Arora, Santiago Cortes, Utsab Barman, Dasha Bogdanova, Jennifer Foster, and Lamia Tounsi. 2014. Dcu: Aspect-based polarity classification for semeval task 4. In *Proceedings of SemEval*, pages 223-229.

Yequan Wang, Minlie Huang, xiaoyan zhu, and Li Zhao. 2016. Attention-based lstm for aspect-level sentiment classification. In *Proceedings of EMNLP*, pages 606-615.

Wojciech Zaremba, Ilya Sutskever, and Oriol Vinyals. 2014. Recurrent neural network regularization. *preprint arXiv: 1409.2329*.

Shiliang Zheng and Rui Xia. 2018. Left-Center-Right Separated Neural Network for Aspect-based Sentiment Analysis with Rotatory Attention. *arxiv preprint arXiv:1802.00892*.