



# Transformer Based Memory Network for Sentiment Analysis of Chinese Weibo Texts

Junlei Wu<sup>(✉)</sup>, Jiang Ming, and Min Zhang

Institute of Software and Intelligent Technology, Hangzhou Dianzi University,  
Hangzhou 310018, China  
171050047@hdu.edu.cn, jmzju@163.com, hz\_andy@163.com

**Abstract.** Weibo has already become the main platform of mobile social and information exchange. Therefore, the sentiment feature extraction of Weibo texts is of great significance, and aspect-based sentiment analysis (ABSA) is useful to retrieval the sentiment feature from Weibo texts. Now, context-dependent sentiment feature is obtained by widely using long short-term memory (LSTM) or Gated Recurrent Unit (GRU) network, and target vector is usually replaced by average target vector. However, Weibo texts has become increasingly complex and feature extraction with LSTM or GRU might cause the loss of key sentiment information. Meanwhile, average target vector might be wrong target feature. To correct drawbacks of the old method, a new Transformer (a new neural network architecture based on self-attention mechanism) based memory network (TF-MN), is introduced. In TF-MN, the task is migrated into question answering process in which context, question and memory module is modified optimally. The text is encoded by Transformer in context module, question module transfer target into sentiment question, memory module eliminates the effect of unrelated words by several extractions. The result of the experiment proves that our model reaches better accuracy than the state-of-the-art model.

**Keywords:** ABSA · Transformer · Memory network · Weibo texts

## 1 Introduction

Recently, rapid development has been witnessed in mobile social which has fully infiltrated into the global user communities. As one of the most important mobile social applications, Weibo contains entertainment, social, marketing and so on [1]. It has gradually evolved from a social demand that satisfies people's "weak relationship" to a popular public opinion platform, becoming one of the most important realtime information sources and the center of spreading public opinion. Viewpoints and proposals in Weibo are universal and adaptive due to large amounts of users along with the differences of their standpoints

and knowledge they have. Consumers or businesses can understand most users' emotional attitudes toward the relative products by means of sentiment analysis technique which can provide policy-making references for consumers and the evidence for enterprises to improve product quality. The recognition of the subjective information in Weibo is the main purpose of Weibo sentiment analysis, which means to analysis users' viewpoints and proposals toward products, news and hotspots.

Various aspects are included in viewpoints expressed in Weibo. For example, the text, "Nice Service but the food was too bad!", expresses the emotions of two different goals. With the approach of aspect level sentiment analysis, we can analyze the opinions and emotions expressed by Weibo in a more fine-grained manner. As to the above instance, the polarity is positive when target is "service", but it turns negative if "food" is seen as target.

Traditional methods focus on characteristic rule such as sentiment vocabulary and word bag feature for aspect-level sentiment analysis. These features are utilized to train a classifier [2]. However, the manual features are concentrated labor force and are highly relied by these methods. Different from previous methods, neural network models extract text feature in a labor-saving and scalable way.

Learning text feature is mainly by means of sequence transduction models in neural network models. The mainstream of sequence transduction models are based on complicated recurrent neural network (RNN) or convolutional neural networks (CNN) which consist of an encoder and a decoder. The models that connect the encoder and decoder through attention cells give the best combined properties [3]. Long short-term memory (LSTM) [4] and gated recurrent (GRU) [5] neural networks have been attained the best result in sentiment analysis domain. However, Weibo texts has become increasingly complex and it is quite difficult to extract context-dependent text feature. LSTM and GRU are the best sequence transduction models at present, which can't process long text. Reference [6] puts forward that text could be separated into four parts to ensure no key information lost. But, context-dependent text feature may be unable to be extracted.

Besides, target may consist of several words, and target feature can be learned directly by sequence transduction models. Experiment shows that average target vector is the best method to get target feature [7], however this method has certain weaknesses. For example, in text "Nice macarons in France are not good at all.", the target is "Nice macarons", consists of two words. Due to the limited number of words, vocabulary only contains "Nice", and thereby "macarons" will be assigned to a set of minimal random feature. Then, the average target feature of "Nice macarons" equals approximately the feature of "Nice", which leads to wrong results of classification.

Based on the two problems analyzed above, we propose a memory network model that combines Transformer. Transformer is a novel network architecture, based solely on attention mechanisms without recurrent and convolutional structure entirely. Its basic unit is self-attention mechanism which can obtain better

context-dependent text representation using interactive calculation in each part of sequence whether the size of sequence.

And we utilize the memory network to capture sentiment information of given target. It contains four modules: the context module for encoding Weibo texts, the question module for storing knowledge from previous steps, the question module for conversion target and encoding questions, and the answer module for evaluating sentiment polarity by data from memory module.

The memory network is proposed for the question answering task, however, the aspect-based sentiment classification doesn't have an exact question. The original MN handles this by initializing the problem vector generated by the problem module with a zero or offset vector, while we argue that every target in the text could be converted into a question. We propose the Transformer based memory network (TF-MN) to realize our ideas, the question module of TF-MN treats each target in the text as implicitly asking a question "What is the emotion tendency of target in the text?". Figure 1 is the overview of TF-MN architecture.

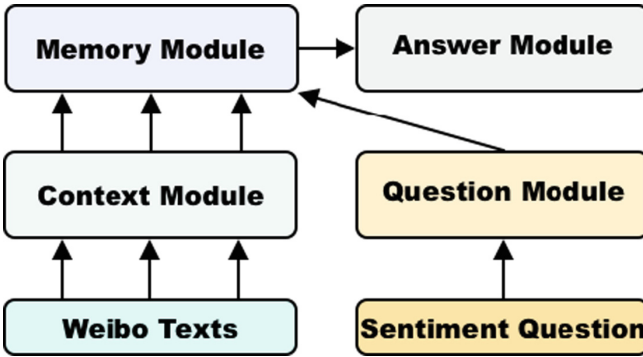


Fig. 1. The architecture diagram of TF-MN.

The following is a summary of our work:

- In sentiment analysis task, we extract long text feature by using Transformer for the first time. Our model effectively solves the problem of inaccurate long text feature extraction by using LSTM and GRU.
- The problem module does not average the target word vector, but is responsible for designing the corresponding target problem to handle the case where the target consists of multiple words.
- On our dataset, our model achieved the best accuracy, and the experimental results further show that using Transformer and adding implicit question can actually improve the performance of the model.

## 2 Related Work

ABSA is a subdomain in sentiment analysis which focuses on fine-grained sentiment information [8]. There are two main approaches to solve ABSA problem.

The first one is the traditional method using lexicons and rules. Reference [9] compute sentiment word score using the weight sum method. Reference [10] proposed a holistic lexicon-based method involving both explicit and implicit opinions. The method improves performance to identify the aspect level relations by multiple kernels [11].

The second one is the machine learning method. Reference [12] employed hinge-loss Markov random fields to tackle ABSA in MOOC. Reference [13] put forward a emotion-aligned model for to predict aspect rate. Reference [14] combined vocabulary-based method and characteristics-based Support Vector Machine (SVM), and detect sentiment towards aspect words in SemEval 14 competition at first time. Reference [15] constructed binary phrase dependency tree of target to build the feature of aspect words. Reference [16] solved the problem using recurrent neural network, and suggested two approaches TD-LSTM and TC-LSTM. Reference [17] proposed an attention-based LSTM method. It is the best method that deal with abstractive context memory information. Reference [18] introduced a deep memory network method to solve ABSA task. It used hierarchical structure model in which the text was fully connected with target for final classification by using attention cell [19]. Reference [20] processed average target and context-dependent vector from LSTM with attention method. Reference [6] introduced a method of dividing each sentence into three parts, and the context-dependent feature was extracted using bidirectional GRU. LSTM and GRU are widely used by the models mentioned above without considering how to extract the long text feature. Meanwhile, they never consider the situation of lacking target in vocabulary when getting target by average target vector.

Different from above models, we are enlightened by self-attention mechanism. We resolve the problem of long text by Transformer. In addition, The way of solving the wrong target aims to convert target into the form of sentiment question. At last, we eliminate the influence of sentiment-irrelevant words by multiple extractions in memory module.

## 3 The Proposed Model

We presents TF-MN model for ABSA in this section. The task of ABSA concluded as follows: given a text consisting of  $n$  words  $C = \{w_1^C, w_2^C, \dots, w_{n-1}^C, w_n^C\}$  that is named context, a target  $T = \{w_1^T, w_2^T, \dots, w_{i-1}^T, w_i^T\}$  in which several adjacent words appear in the context, and the aim is predicting the sentiment polarity of the specified target in the given context.

Figure 2 shows the TF-MN architecture, which converts context into a consecutive low dimensional sequence with pretrained word embeddings in the context module. And Transformer processes context sequence to preserve sequential information in memory module. In question module, target is converted into sentiment question. The question is in the form of ‘‘What is the emotion tendency

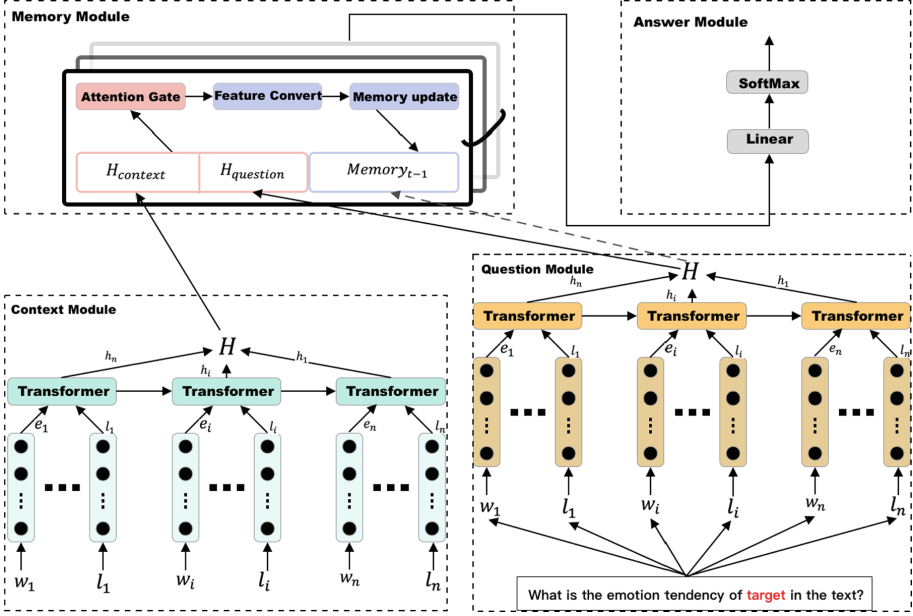


Fig. 2. The architecture of TF-MN model.

of target in the text?”. Then, Transformer is also operated upon the question. In memory module, we eliminates the effect of unrelated words by several extractions. Finally, softmax layer outputs sentiment polarity. Each step of our model are showed as follows.

### 3.1 Context Module

The context module includes the following layers: the context encoder, the location encoder and the fusion layer. The context and the location encoder layer encode each context and location information into a vector separately, while the fusion layer exchange information between these encoded vectors using Transformer.

**Context Encoder Layer.** Specified a context  $C = \{w_1^C, w_2^C, \dots, w_{n-1}^C, w_n^C\}$ , every word in  $C$  is converted into a  $k$ -dimensional vector  $e_i^C \in \mathbb{R}^k$  with a pre-trained word embedding matrix  $E \in \mathbb{R}^{k*|V|}$ , such as Tencent AI Lab Embedding [21]:

$$e_i^C = E(w_i^C) \tag{1}$$

where “ $|V|, k$ ” are the size of vocabulary and word vector respectively.

**Location Encoder Layer.** We establish a context location word embedding matrix  $L \in \mathbb{R}^{k \times n}$ , which maps word location into a  $k$ -dimensional vector  $l_i^C \in \mathbb{R}^k$ :

$$l_i^C = L(w_i^C) \quad (2)$$

where  $n$  is the dimension of row vector in  $L$ . To provide rich location information for context, row vector of  $L$  is a  $k$ -dimensional vector consisting of  $k$  location information. Matrix  $L$  is a set of parameters to be trained, in which every row vector will be assigned a sequence of location information with random normal  $U(-0.02, 0.02)$ .

**Fusion Layer.** The fusion layer processes the context vector  $E_C$  and context location vector  $L_C$  which contain exchanged information among vectors. We generate context representation  $H_C \in \mathbb{R}^{k \times nc}$ :

$$H_c = Transformer(E_C, L_C) \quad (3)$$

where  $nc$  denotes the max size of context (If the length is not enough, fill it with 0).

### 3.2 Question Module

The question module further also contains these layers: the question encoder layer, the location encoder layer and the fusion layer. The question encoder layer converts target into sentiment question firstly, and then encodes question into vector. The location encoder layer encodes location information into a vector. The fusion layer fuses these vectors into more specific features through the Transformer.

**Question Encoder Layer.** Given a question  $T = \{w_1^T, w_2^T, \dots, w_i^T\}$ ,  $T$  consists of one or more words, which can be included in  $C$  or doesn't appear in  $C$ . If  $T$  is embedded in sentiment question, you will get the question as "What is the emotion tendency of target in the text?". Every word in question is converted into a  $k$ -dimensional vector  $e_i^Q \in \mathbb{R}^k$  with a pretrained word embedding matrix  $E$ :

$$e_i^Q = E(w_i^Q) \quad (4)$$

**Location Encoder Layer.** We also establish a question location word embedding matrix  $L$ , which maps word location into a  $k$ -dimensional vector  $l_i^Q \in \mathbb{R}^k$ :

$$l_i^Q = L(w_i^Q) \quad (5)$$

where the location information matrix of  $Q$  is the same as context module.

**Fusion Layer.** The fusion layer generates the sentiment question representation  $H_Q \in \mathbb{R}^{k*nq}$ :

$$H_Q = Transformer(E_Q, L_Q) \quad (6)$$

where  $nq$  denotes the max size of question.

### 3.3 Memory Module

The memory module has three components: the attention gate, feature conversion and the memory update gate, which is used to combine information from context with target and purify of the target vector from the given context.

The output  $F$  from context module, the question  $q^*$  from question module and the acquired knowledge stored in the memory vector  $m_{t-1}$  from the previous step.

The three inputs are transformed by:

$$u = [F * q^*; |F - q^*|; F * m_{t-1}; |F - m_{t-1}|] \quad (7)$$

where “;” is concatenation. “ $*$ ”, “ $-$ ”, “ $||$ ” are element-wise product, subtraction and absolute value respectively.  $F$  is a matrix of size  $(1, H_C)$ , while  $q^*$  and  $m_{t-1}$  are vectors of size  $(1, H_Q)$  and  $(1, H_m)$ , where  $H_m$  is the output size of the memory update gate. To allow element-wise operation,  $H_C$ ,  $H_Q$  and  $H_m$  are set to the same shape. In Eq. (7), the first two terms measure the similarity and difference between facts and the question. The last two terms have the same functionality for context and the last memory state.

Let the  $i$ -th element in  $\alpha$  to be the attention weight for  $w_i^C$ .  $\alpha$  is obtained by transforming  $u$  using a two-layer perceptron:

$$\alpha = softmax(tanh(u \cdot W_{m1}) \cdot W_{m2}) \quad (8)$$

where  $W_{m1}$  and  $W_{m2}$  are parameters of the perceptron and we omit bias terms.

The feature conversion takes  $F$  and  $\alpha$  as input and then get the updated  $F$ :

$$F = F \cdot \alpha \quad (9)$$

The memory update gate outputs the updated memory  $m_t$  using question  $q^*$ , previous memory state  $m_{t-1}$  and the updated  $F$ :

$$m_t = relu([q^*; m_{t-1}; F] \cdot W_u) \quad (10)$$

where  $W_u$  is the parameter of the linear layer.

The memory module could be iterated several times with a new  $\alpha$  generated for each time. This allows the model to attend to different parts of the facts in different iterations, which enables the model to perform complicated reasoning across sentences. The memory module produces  $m_t$  as the output at the last iteration.

### 3.4 Answer Module

In answer module, we regard the memory module outputs as the final representation, and put it into a softmax layer for aspect-based sentiment analysis task. To minimize the cross entropy error of sentiment classification, we train the model in a supervised method in which loss function is described as follows:

$$loss = - \sum_{(c,q) \in T} \sum_{lb \in LB} P_{lb}^g(c,q) \cdot \log(P_{lb}(c,q)) \quad (11)$$

where  $T$  is all training items,  $LB$  is the set of sentiment polarities,  $(c, q)$  is a context-question pair. Our system outputs the probability of class  $lb$  by computing the item  $(c, q)$ .  $P_{lb}^g(c, q)$  means zero or one, expressing whether the item is positive or not. In the module, We calculate the gradients of the overall parameters by using back propagating and update them in a stochastic gradient descent manner.

## 4 Experiment

### 4.1 Dataset and Experiment Setup

**Dataset.** In most recent ten years, a lot of Chinese Weibo competitions have been held, and many excellent datasets have been produced such as NLPCC 2013 and 2014 training dataset. Unfortunately, Most datasets only analyze the overall sentiment polarity of the entire sentence. So we try to construct a new dataset for ABSA. We collect weibo data from Weibo, and each the weibo may contain multiple target entities. Each target can be an entity that appears in weibos or an abstracted entity in weibos. The emotional polarity of the goals we mark includes positive, negative, and neutral. If the target has a divergence between these three polarities, we will ignore this target entity. We mainly build dataset in the fields of restaurant which contains four aspects: traffic, service, price and environment. Then, we randomly selected 18480 Weibo items (a total of 22821 Weibo items) as training set, and the rest 4341 items as test set. Table 1 is the detail of the dataset. Finally, it is noted that the text length of our dataset is generally more than 200 words, which belong long texts.

**Evaluation.** We validate our model with the accuracy, and remove the label from test dataset before training the model. If the output label of the model answer module matches the label by manual method, it will become the right classification result.

**Parameter Setting.** In the model, the word embeddings matrix of the contexts and targets in the dataset are assigned 200-dimensional vectors from Tencent AI Lab Embedding [21]. All words out of the vocabulary are randomly assigned a vector that obeys uniform distribution  $U(-0.01, 0.01)$ . To prevent data overfitting, we set the loss rate to 0.1. Our optimizer is Adam whose batch size and



**Table 1.** Statistics of dataset

Type	Aspect	Negative	Neutral	Positive
Train	Traffic	629	629	629
	Service	1929	1929	1929
	Price	2197	2097	2097
	Environment	1505	1505	1505
Test	Traffic	88	88	88
	Service	871	871	871
	Price	280	280	280
	Environment	208	208	208

learning rate is 8 and  $6.25e-5$  respectively. We use jieba [22] to do Chinese phrase segmentation and generate the word vector matrix of our experiment. It is noted that the results of the experiment will change with each randomly assigned word vector even if we set up the seeds of the experiment. In order to solve this problem, the experimental data are obtained through average 10 experimental results.

## 4.2 Experiment Setting

In order to test the performance of our model, we did the following set of experiments: SVM, LSTM, TD-LSTM [17], AT-LSTM [23], IAN [24], BILSTM-ATT, MENNET [16].

Firstly, for the SVM experiment, we directly call the svm class inside sklearn [25]. Then, since the number of samples is much larger than the number of features, we use a nonlinear kernel rbf. As for the optimal parameters of the model, we search for parameters in a large-range and large-step grid by the grid search method.

Secondly, we use the LSTM and BILSTM module in tensorflow, where we regard Weibo text as input. Then, we output the results of the classification through the softmax layer.

Finally, we use the open source TD-LSTM and AT-ISTM [26], IAN [27], MENNET [28] code on github to do experiment. Some parameters and settings of the experiment are as close as possible to the original author’s paper.

## 4.3 Model Comparisons and Analysis of Results

In these models, SVM belongs to the traditional machine learning field; LSTM and TD-LSTM are general neural network model methods; AT-LSTM, IAN, BILSTM-ATT, MEMNET are mainly apply the attention mechanism.

We compared the TF-MN model with the correctness of other models, and the results of this comparison are listed in Table 2.

In this table, we can observe that the performance difference between SVM performance and TF-MN model is the largest. We suggest it is caused by two reasons: the SVM training model does not use aspect; the SVM model only classifies the text, and does not mine deep text features.

The effect of the LSTM model on this data set is also not very good, because LSTM can cause the loss of key sentiment information due to the mechanism of forget-door when processing long text.

The accuracy of TD-LSTM is much better than the single LSTM model because it combines ASPECT and text.

Unlike TD-LSTM, AT-LSTM uses a attention mechanism to effectively extract important emotional information from text with aspect, which greatly enhances the final classification result.

BILSTM-ATT is derived from the improvement of the AT-LSTM model, which uses a bidirectional LSTM model. Bidirectional LSTM can significantly improve the performance of the model for sequence classification problems.

The above models mainly focus on the impact of ASPECT on text, but the IAN model also uses the influence of text on ASPECT as a basis for classification. It believes that ASPECT and text should be mutually influential and not just one-way connections.

Based on the above models, MENNET proposes that the attention mechanism should directly affect the process of LSTM coding. Therefore, this model abandon the LSTM model and uses a simpler memory module to encode information. This memory model repeatedly uses the local attention mechanism to extract information and achieves good results.

Although MENNET is good enough, we found that the separate memory module does not encode the text information well during the experiment, and it is as bad as the LSTM model for longer text encoding. At the same time, we also feel that the previous treatment of ASPECT is too rough. Based on these two problems, we improve the MENNET model and achieve better model results.

**Table 2.** 3-way experimental results in accuracy. 3-way represents the three polarities of positive, negative, neutral. Best scores in each group are in bold.

Model	3-way restaurant
SVM	51.55
LSTM	52.45
TD-LSTM	55.26
AT-LSTM	56.34
BILSTM-ATT	57.47
IAN	58.33
MEMNET	60.78
TF-MN	<b>61.67</b>

#### 4.4 Memory Network Optimization

To improve the effect of the memory module to extract emotional information, we conducted a number of experiments to optimize and adjust the number of our memory updates. We found that when the updated hop count is set to 5, the model classification works best. The results of these groups of experiments are shown in Table 3. We believe this is due to excessive update operations that cause the local attention mechanism to repeatedly operate on the same block of text.

**Table 3.** The result of Memory network hop count comparison.

Hops	Restaurant accuracy
1-hop	58.78
2-hop	59.40
3-hop	60.55
4-hop	60.99
5-hop	<b>61.67</b>

## 5 Conclusion

In this work, we explore the use of memory network architecture to model sentiment classification into the question answering task. The key is to frame the goal as an emotional question. Therefore, we believe that memory networks can be replaced by other more efficient network architectures. We believe that the attention gate in the memory module of TF-MN can add syntactic information. Other tasks with context but no clear issues may also benefit from this work. In the paper, the TF-MN model is proposed, which uses the memory network model to model the Weibo sentiment analysis to the question answering task. We turn the goal into an emotional question. We have done 3-way experiments in the field of restaurant on the Weibo dataset. The results show that this way of modeling improves the accuracy of classification.

**Acknowledgements.** This work is supported by Zhejiang Provincial Technical Plan Project (No. 2018C03039, 2018C03052).

## References

1. Chen, X., et al.: Degree of integration into social networks (DISN) evaluation model based on micro-blogging platforms and information dissemination prediction. In: 2016 IEEE International Conference on Computer and Information Technology (CIT). IEEE (2016)
2. Pontiki, M., Galanis, D., Papageorgiou, H., et al.: SemEval-2016 task 5: aspect based sentiment analysis. In: Proceedings of the 10th International Workshop on Semantic Evaluation, SemEval-2016, pp. 19–30 (2016)

3. Vaswani, A., et al.: Attention is all you need. In: *Advances in Neural Information Processing Systems* (2017)
4. Hochreiter, S., Schmidhuber, J.: Long short-term memory. *Neural Comput.* **9**(8), 1735–1780 (1997)
5. Chung, J., et al.: Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint [arXiv:1412.3555](https://arxiv.org/abs/1412.3555) (2014)
6. Zheng, S., Xia, R.: Left-center-right separated neural network for aspect-based sentiment analysis with rotatory attention. arXiv preprint [arXiv:1802.00892](https://arxiv.org/abs/1802.00892) (2018)
7. Do, H.H., et al.: Deep learning for aspect-based sentiment analysis: a comparative review. *Expert Syst. Appl.* **118**, 272–299 (2018)
8. Liu, B.: Sentiment Analysis and Opinion Mining. *Synthesis Lectures on Human Language Technologies*, vol. 5, no. 1, pp. 1–167 (2012)
9. Hu, M., Liu, B.: Mining and summarizing customer reviews. In: *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM (2004)
10. Ding, X., Liu, B., Yu, P.S.: A holistic lexicon-based approach to opinion mining. In: *Proceedings of the 2008 International Conference on Web Search and Data Mining*. ACM (2008)
11. Nguyen, T.H., Shirai, K.: Aspect-based sentiment analysis using tree kernel based relation extraction. In: Gelbukh, A. (ed.) *CICLing 2015*. LNCS, vol. 9042, pp. 114–125. Springer, Cham (2015). [https://doi.org/10.1007/978-3-319-18117-2\\_9](https://doi.org/10.1007/978-3-319-18117-2_9)
12. Ramesh, A., et al.: Weakly supervised models of aspect-sentiment for online course discussion forums. In: *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*. Long Papers, vol. 1 (2015)
13. Wang, H., Ester, M.: A sentiment-aligned topic model for product aspect rating prediction. In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (2014)
14. Kiritchenko, S., et al.: NRC-Canada-2014: detecting aspects and sentiment in customer reviews. In: *Proceedings of the 8th International Workshop on Semantic Evaluation, SemEval 2014* (2014)
15. Nguyen, T.H., Shirai, K.: PhraseRNN: phrase recursive neural network for aspect-based sentiment analysis. In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing* (2015)
16. Tang, D., et al.: Effective LSTMs for target-dependent sentiment classification. arXiv preprint [arXiv:1512.01100](https://arxiv.org/abs/1512.01100) (2015)
17. Wang, Y., Huang, M., Zhao, L.: Attention-based LSTM for aspect-level sentiment classification. In: *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing* (2016)
18. Tang, D., Qin, B., Liu, T.: Aspect level sentiment classification with deep memory network. arXiv preprint [arXiv:1605.08900](https://arxiv.org/abs/1605.08900) (2016)
19. Cheng, J., et al.: Aspect-level sentiment classification with heat (hierarchical attention) network. In: *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. ACM (2017)
20. He, R., et al.: Effective attention modeling for aspect-level sentiment classification. In: *Proceedings of the 27th International Conference on Computational Linguistics* (2018)
21. Song, Y., Shi, S., Li, J., Zhang, H.: Directional skip-gram: explicitly distinguishing left and right context for word embeddings. In: *NAACL 2018. Short Paper* (2018)
22. Sun, J.: Jieba' Chinese word segmentation tool (2012). <https://github.com/fxsjy/jieba>. Accessed 10 May 2019

23. Wang, Y., Huang, M., Zhao, L.: Attention-based LSTM for aspect-level sentiment classification. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (2016)
24. Ma, D., et al.: Interactive attention networks for aspect-level sentiment classification. arXiv preprint [arXiv:1709.00893](https://arxiv.org/abs/1709.00893) (2017)
25. sklearn. <https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html>. Accessed 5 Mar 2019
26. TD-LSTM and AT-LSTM. <https://github.com/jimmyyfeng/TD-LSTM>. Accessed 5 Mar 2019
27. IAN. <https://github.com/lpq29743/IAN>. Accessed 5 Mar 2019
28. MENNET. [https://github.com/Humanity123/MemNet\\_ABSA](https://github.com/Humanity123/MemNet_ABSA). Accessed 5 Mar 2019