Research Article

An Interactive Model of Target and Context for Aspect-Level Sentiment Classification

Hu Han,1,2 Guoli Liu,1 and Jianwu Dang1,2

1School of Electronic & Information Engineering, Lanzhou Jiaotong University, Lanzhou 730070, China
2Gansu Provincial Engineering Research Center for Artificial Intelligence and Graphic & Image Processing, Lanzhou 730070, China

Correspondence should be addressed to Guoli Liu; 0617514@stu.lzjtu.edu.cn

Received 16 September 2019; Revised 15 November 2019; Accepted 2 December 2019; Published 19 December 2019

Academic Editor: Maciej Lawrynczuk

Copyright © 2019 Hu Han et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Aspect-level sentiment classification aims to identify the sentiment polarity of a review expressed toward a target. In recent years, neural network-based methods have achieved success in aspect-level sentiment classification, and these methods fall into two types: the first takes the target information into account for context modelling, and the second models the context without considering the target information. It is concluded that the former is better than the latter. However, most of the target-related models just focus on the impact of the target on context modelling, while ignoring the role of context in target modelling. In this study, we introduce an interactive neural network model named LT-T-TR, which divided a review into three parts: the left context with target phrase, the target phrase, and the right context with target phrase. And the interaction between the left/right context and the target phrase is utilized by an attention mechanism to learn the representations of the left/right context and the target phrase separately. As a result, the most important words in the left/right context or in the target phrase are captured, and the results on laptop and restaurant datasets demonstrate that our model outperforms the state-of-the-art methods.

1. Introduction

The aspect-level sentiment classification is a fine-grained task in sentiment analysis [1]. Given a review and a target occurring in the review, it aims to identify the sentiment polarity (e.g., negative, neutral, or positive) expressed on each target in its context. For example, considering this review “the voice quality of this phone is amazing, but the price is ridiculous,” we observe that there are two targets (“voice quality” and “price”) with completely opposite polarities. The sentiment expressed on target “voice quality” is positive, whereas the sentiment for target “price” is negative. Jiang et al. [2] introduced a target-dependent Twitter sentiment classifier, which showed that not considering the target information discussed in the review results in 40% of sentiment classification errors. Therefore, the task of aspect-level sentiment classification is also aimed at predicting a sentiment category for a review-target pair.

Different from sentence- and document-level sentiment analysis [3–6], for aspect-level sentiment classification, a review may contain multiple review-target pairs, and thus, separating different contexts for different targets is a challenge. Many methods based on neural networks have been proposed for aspect-level sentiment classification. For example, Dong et al. [7] used the adaptive recursive neural network to evaluate the sentiments of specific targets in context words. Vo and Zhang [8] separated the whole review into three sections (the target, its left contexts, and its right contexts) and used neural pooling functions and sentiment lexicon to extract the feature vector for a given target. Tang et al. [9] divided the review into the left part with the target and the right part with the target and then used two long short-term memory (LSTM) networks to encode the two parts, respectively. Zhang et al. [10] used a gated neural network to capture the interaction information between the target and its surrounding contexts. To further focus on
important words of the sentence that modulate the sentiment of the targets, Wang et al. [11] introduced LSTM networks and an attention mechanism to concatenate word representations with target embeddings to generate the final sentiment representations.

Although the previous approaches have realized the importance of targets in sentiment classification, these approaches only focus on the impact of targets on context modeling. How to use the interaction information between contexts and the target phrase to separately model contexts and targets has become a new research issue. Ma et al. [12] proposed an interactive attention network (IAN) that uses two LSTM networks to model the contexts and target phrase, respectively, and then uses the hidden states from the contexts to generate an attention vector for the target phrase, and vice versa. Based on [12], Huang et al. [13] proposed an attention-over-attention (AOA) neural network, which models targets and reviews simultaneously using two LSTMs and then the target representation and text representation can be interacted through the AOA module. Zheng and Xia [14] designed a left-center-right separated neural network to model the left context, target phrase, and right context, respectively, and modeled the relation between the target and the left/right context using a rotatory attention mechanism.

To further improve the representations of targets and contexts, we propose an interactive neural network model named LT-T-TR. Firstly, it divides a review into three parts: the left context with the target phrase, the target phrase, and the right context with the target phrase. Three Bidirectional Long Short-Term Memory networks (Bi-LSTMs) are used to model these parts, respectively. Secondly, different words in reviews have different contributions to the final representation, and contexts and targets are influenced by each other, so attention weights of the target phrase and the left/right context are computed by interactive attention between the target phrase and the left/right context. The process is made up of two parts: the first is target-to-context attention, which includes the target-to-left context attention and the target-to-right context attention, to get better representations of the left/right context; the second is context-to-target attention that includes the left context-to-target attention and the right context-to-target attention. After computing these attention weights, we get the target phrase and left/right context using attention weights.

(c) Concatenating these representations to form the final classification vectors and evaluating our model on laptop and restaurant datasets.

2. Model

In this section, we first give the task definition of aspect-level sentiment classification. Afterward, we introduce the different components of our model as displayed in Figure 1.

2.1. Task Definition. Given a review \( S = [w_1, \ldots, w_m, w_{m+1}, \ldots, w_{m+j}, \ldots, w_n] \) consisting of \( n \) words, \( w_1, w_2, \ldots, w_m \) are the preceding context words, \( w_{m+1}, w_{m+2}, \ldots, w_{m+j}, \ldots, w_n \) are the target words, and \( w_{m+j}, \ldots, w_n \) are the following context words. We divide the review into three parts: the left context \( LT = [w_1, w_2, \ldots, w_{m+j}] \) consisting of \( w_1, w_2, \ldots, w_m \) and \( w_{m+1}, w_{m+2}, \ldots, w_{m+j} \), the target phrase \( T = w_{m+1}, w_{m+2}, \ldots, w_{m+j} \), the target to context \( T \) right context \( RT = [w_{m+1}, w_{m+2}, \ldots, w_n] \) consisting of \( w_{m+1}, w_{m+2}, \ldots, w_n \) and \( w_{m+j}, \ldots, w_n \). Aspect-level sentiment classification aims at determining the sentiment polarity of review \( S \) toward target \( T \). For example, the sentiment polarity of review “the voice quality of this phone is amazing, but the price is ridiculous” toward target “voice quality” is positive, but the polarity toward target “price” is negative.

2.2. Bi-LSTMs. First, we represent each word in \( S \) as word embedding [15] and get word vectors \([v_{1}, v_{2}, \ldots, v_{m+j}] \in R^{(r-1)d} \), \([v'_{m+1}, v'_{m+2}, \ldots, v'_{n}] \in R^{(r-m-1)d} \), and \([v''_{m+1}, v''_{m+2}, \ldots, v''_{n}] \in R^{(r-n)d} \) for LT, \( T \), and RT, where \( d \) is the embedding dimension. Then, we feed these three-part word vectors to three Bi-LSTMs [16], respectively, to learn the hidden word semantics. Each Bi-LSTM is obtained by stacking a forward LSTM and a backward LSTM, which are good at learning long-term dependencies. In the LSTM architecture, there are three gates (input gate, forget gate, and output gate) and a cell memory state. Each cell can be updated as follows:

\[
X = \begin{bmatrix} h_{k-1} \\ v_k \end{bmatrix},
\]

\[
i_k = \sigma(W_i \cdot X + b_i),
\]

\[
f_k = \sigma(W_f \cdot X + b_f),
\]

\[
o_k = \sigma(W_o \cdot X + b_o),
\]

\[
c_k = f_k \odot c_{k-1} + i_k \odot \tanh(W_c \cdot X + b_c),
\]

\[
h_k = o_k \odot \tanh(c_k),
\]

where \( \sigma \) is the sigmoid function, \( \odot \) denotes elementwise multiplication, and \( \cdot \) stands for matrix multiplication; \( W \) and \( b \) denote the weight matrices and biases, respectively; \( v_k \) is the input word vector, and \( h_{k-1} \) is the previous hidden state.
For the left context \( LT \), the input of Bi-LSTM is \( [v_1, v_2, \ldots, v_{s-1}] \in R^{(s-1) \times d} \) and we get hidden states as follows:

\[
[h_1^L, h_2^L, \ldots, h_{s-1}^L] = \text{Bi-LSTM}(\{v_1, v_2, \ldots, v_{s-1}\}),
\]

where the output \( h_k^L (k = 1, \ldots, s-1) \) is obtained by concatenating the corresponding states of the forward and backward LSTM. Similarly, we can get the hidden semantic states \( [h_{s+1}^L, h_{s+2}^L, \ldots, h_n^L] \) for target \( T \) and the hidden states \( [h_1^R, h_2^R, \ldots, h_{n-1}^R] \) for the right context \( RT \) in the same way.

Then, through an average pooling operation, we can obtain the initial representations of \( LT, T \), and \( RT \) as follows:

\[
LT_{\text{initial}} = \sum_{k=1}^{s-1} \frac{h_k^L}{s-1},
\]

\[
T_{\text{initial}} = \sum_{k=s+1}^{n-1} \frac{h_k^L}{s-m-1},
\]

\[
TR_{\text{initial}} = \sum_{k=m+1}^{n} \frac{h_k^R}{n-m}.
\]

2.3. Attention Layer. After getting the hidden representations of the context and the target phrase generated by three Bi-LSTMs, we use the attention mechanism to calculate the different importance of words in the left/right context and the target phrase.

2.3.1. Target-to-Context Attention. Given the hidden representations of the left context \( [h_1^L, h_2^L, \ldots, h_{s-1}^L] \) and the average representation of target \( T_{\text{initial}} \), we first get the target-to-left context attention representation \( LT_{\text{final}} \) by

\[
LT_{\text{final}} = \sum_{k=1}^{s-1} \alpha_k^L h_k^L,
\]

where \( \alpha_k^L \) is the weight of \( h_k^L \) that we can obtain from a softmax function:

\[
\alpha_k^L = \frac{\exp(f_{\text{att}}(h_k^L, T_{\text{initial}}))}{\sum_{j=1}^{s-1} \exp(f_{\text{att}}(h_j^L, T_{\text{initial}}))}.
\]

Here, \( f_{\text{att}} \) is a score function that indicates the importance of words in the left context influenced by the target:

\[
f_{\text{att}}(h_k^L, T_{\text{initial}}) = \tanh(h_k^L \cdot W_a \cdot T_{\text{initial}}^T + b_a),
\]

where \( \tanh \) is a nonlinear function, \( W_a \) is the weight matrix, \( b_a \) is the bias, and \( T_{\text{initial}}^T \) is the transpose of \( T_{\text{initial}} \).

Similar to equations (6)–(8), we can also obtain the target-to-right context attention representation \( TR_{\text{final}} \) using the average representation of the target \( T_{\text{initial}} \).

2.3.2. Context-to-Target Attention. For the hidden representations of target \( [h_{s+1}^L, h_{s+2}^L, \ldots, h_n^L] \), we first compute the weight representations as follows:

\[
\alpha_k^R = \text{softmax}(f_{\text{att}}(h_k^R, LT_{\text{initial}})),
\]

\[
f_{\text{att}}(h_k^R, LT_{\text{initial}}) = \tanh(h_k^R \cdot W_L \cdot LT_{\text{initial}}^T + b_L),
\]
where $W_L$ and $b_L$ are the weight matrix and bias, respectively.

Then, through calculating the weighted combination of the hidden states of the target phrase, we can obtain the left context-to-target representation as follows:

$$T_{\text{final}}^{l} = \sum_{k=m+1}^{s-1} a_k h_k^l. \quad (11)$$

Similar to equations (9)–(11), we can obtain the right context-to-target representation $T_{\text{final}}^{r}$ by using $TR_{\text{initial}}$ and the hidden representations of the target.

After getting $T_{\text{final}}^{l}$ and $T_{\text{final}}^{r}$, we get the final representation of the target phrase through concatenating $T_{\text{final}}^{l}$ and $T_{\text{final}}^{r}$:

$$T_{\text{final}} = [T_{\text{final}}^{l}, T_{\text{final}}^{r}] \quad (12)$$

2.4. Final Classification. Then, we concatenate $LT_{\text{final}}$, $T_{\text{final}}^{r}$, and $TR_{\text{final}}$ as the final representation of review $S$:

$$v = [LT_{\text{final}}; T_{\text{final}}^{r}; TR_{\text{final}}]. \quad (13)$$

We project $v$ into the space of targeted $C$ classes through a nonlinear function:

$$x = \tanh(W_v \cdot v + b_v), \quad (14)$$

where $W_v$ and $b_v$ are the parameters. Finally, the sentiment polarity of the review $S$ with sentiment polarity $c \in C$ toward a target $T$ is calculated as follows:

$$P(y = c) = \frac{\exp(x_c)}{\sum_{c \in C} \exp(x_c)}. \quad (15)$$

2.5. Model Training. The model is trained in an end-to-end way. The loss function is the crossentropy error:

$$\text{loss} = -\sum_{(S,T) \in D} \sum_{c \in C} g(y(S,T) = c) \cdot \log(P(y(S,T) = c)), \quad (16)$$

where $D$ means all training data, $(S,T)$ means a review-target pair, $C$ is the number of categories of sentiment, $P(y(S,T) = c)$ is the probability of predicting $(S,T)$ as class $c$ given by the softmax function, and $g(y(S,T) = c)$ shows whether class $c$ is the correct sentiment category.

3. Experiment

3.1. Experimental Settings

3.1.1. Datasets. We conduct our experiments using the dataset for SemEval 2014 Task 4 [18]. This dataset contains customer reviews on restaurants and laptops. Each review has one or more targets with their corresponding polarities. The polarity of targets can be positive, negative, neutral, or conflict. However, we only consider the first three labels for classification. The statistics of the datasets are shown in Table 1.

| Dataset            | Positive | Neutral | Negative |
|--------------------|----------|---------|----------|
| Laptop-train       | 994      | 464     | 870      |
| Laptop-test        | 341      | 169     | 128      |
| Restaurant-train   | 2164     | 637     | 807      |
| Restaurant-test    | 728      | 196     | 196      |

3.1.2. Parameters and Evaluation Metric. In our experiments, the dimensions of word embeddings, attention vectors, and LSTM hidden states are set to 300. All word embeddings are initialized by GloVe [19], and we randomly initialize the out-of-vocabulary words from uniform distribution $U(-0.1, 0.1)$. All weight matrices are randomly initialized from uniform distribution $U(-0.1, 0.1)$, and all bias terms are set to zero. The dropout rate is set to 0.5.

We adopt the accuracy to evaluate the performance of our model, which is defined as follows:

$$Acc = \frac{T}{N}, \quad (17)$$

where $T$ is the number of correctly predicted samples and $N$ is the total number of samples.

3.2. Model Comparisons. We compare our model with some baseline approaches:

- Majority: the largest sentiment polarity in the training set is regarded as the classification result of each sample in the test set.
- LSTM: a standard LSTM which models the review as a whole and uses the last hidden state of LSTM as the final revise representation [9].
- TD-LSTM: TD-LSTM obtains the final sentiment representation by concatenating two LSTM networks which model the preceding and following contexts surrounding the target, respectively [9].
- AE-LSTM: AE-LSTM concatenated the target vector with each word in review as the input of LSTM [11].
- ATAE-LSTM: ATAE-LSTM appends the aspect embedding into each word vector to strengthen the importance of the target [11].
- IAN: two LSTM networks are used to model the review and target phrase, respectively. It uses the hidden states of the review to generate an attention vector for the aspect, and vice versa. Based on these two attention vectors, it outputs a review representation and an aspect representation for classification [12].

The experimental results are shown in Table 2. First, the worst method is Majority, demonstrating that for aspect-level sentiment classification, a powerful feature representation is important. Then, among all the other methods based on LSTM, the basic LSTM approach has the worst performance because it just models the whole review and ignores the target information. TD-LSTM has an improvement of 1% on the restaurant dataset and 2% on the laptop dataset over LSTM when target information is taken...
To validate the effectiveness of the LT-T-TR model, we design several models in this section. We first input the review as a whole (rather than as three segments) into Bi-LSTM for modeling, and then use the attention mechanism to calculate the importance of each word toward sentiment categories. We refer to this model as No-Separation. Second, we simplify the LT-T-TR model by removing context-to-target attention, which is based on Target-to-Context [12]. Finally, we create an L-T-R model through dividing a review into the preceding context (without target), the target, and the following context (without target) and then model these three parts in the same way as in the LT-T-TR model.

Table 5 shows the experimental results. It can be seen that the No-Separation model achieves the worst performance among all approaches, and the No-Target-Learned model performs worse than No-Interaction and Target-to-Context model. This verifies that the target representation is important to judge the final sentiment categories, and the target should be modeled separately.

And L-T-R and LT-T-TR perform better than the No-Interaction model and the Target-to-Context model, which shows that the interaction between the target phrase and left/right context is important to final sentiment classification. Moreover, L-T-R has slightly worse results than the LT-T-TR model because the target phrase is not contained in the left/right context.

Furthermore, we compare the effect of interactive attention modeling between the target and left/right context. First, we build a model (named No-Interaction) without interactive information by removing the attention interaction operation between the left/right context and target phrase and just learn the attention weight representation by their own Bi-LSTM hidden states. Then, we build the Target-to-Context model by removing context-to-target attention, which is based on Target-to-Context [12]. Finally, we create an L-T-R model through dividing a review into the preceding context (without target), the target, and the following context (without target) and then model these three parts in the same way as in the LT-T-TR model.

Table 5 shows the experimental results. It can be seen that the No-Separation model achieves the worst performance among all approaches, and the No-Target-Learned model performs worse than No-Interaction and Target-to-Context model. This verifies that the target representation is important to judge the final sentiment categories, and the target should be modeled separately.

And L-T-R and LT-T-TR perform better than the No-Interaction model and the Target-to-Context model, which shows that the interaction between the target phrase and left/right context is important to final sentiment classification. Moreover, L-T-R has slightly worse results than the LT-T-TR model because the target phrase is not contained in the left/right context.

3.6. Qualitative Analysis. In this section, we selected three examples from the restaurants dataset to analyze which words contribute the most to the final classification. We get the attention weights and then visualize them by using a visualization tool Helmet [11]. The results are shown in Figure 2, in which the color depth represents the importance of a word: the darker, the more important.

The review in Figures 2(a) and 2(b) is "The people with carts of food don’t understand you because they don’t speak English, their job is to give you the delicious food you point at." The corresponding targets are "food" and "people with carts of food," respectively. It can be seen that when a review contains two targets, the correct sentiment categories for each target can be calculated automatically through our model, that is, the
attention mechanism can dynamically obtain the important words from the whole review. In Figure 2(b), we can see that “people” is the most important word in the target phrase “people with carts of food.” In Figure 2(c), the target is a multiword phrase “fried mini buns with the condensed milk and the assorted fruits on beancurd” and “buns” and “fruits” are more important than other words, so our model pays more attention to “buns” and “fruits.” This also proves that just averaging the vectors of words the target phrase contains to represent the target does not help much. Therefore, modeling the target phrase and context interactively is important for aspect-level sentiment classification.

3.7. Error Analysis. We made an error analysis of the experimental results. The first type of error is caused by non-compositional sentiment expression [20]. For instance, in this review “not only was the look of the food fabulous, but also the taste was to die for,” “taste” is a target and “to die for” is the relevant sentiment expression, whose meaning should not be understood literally. The second kind of error comes from complex sentimental relation expressions such as double negatives, assumptions, and comparisons, like “even though the price of this camera is unacceptable, I love its lens.” Our model fails to deal with the complex sentiment expression in this case. Furthermore, in the review “the movie was really on point—I was surprised,” “movie” is the target word and the idiom “on point” is the relevant sentiment expression, which is difficult to be identified by our model.

4. Related Work

4.1. Aspect-Level Sentiment Classification. Sentiment analysis, also known as opinion mining [1, 21], has brought the widespread attention from both industry and academic communities. As a fine-grained task in the field of sentiment analysis [1], aspect-level sentiment classification has drawn a lot of attention, which is also considered as a kind of text classification problem. Traditional text classification methods depend greatly on the effectiveness of the feature engineering [22], which lacks generalization and is difficult for us to discover the potential explanatory or discriminative factors of data. In recent years, distributed word representation and neural network methods have been proposed and shown promising performance on this task [7, 8]. Dong et al. [7] used an adaptive recursive neural network to evaluate the sentiments of specific targets in context words. Vo and Zhang [8] separated the whole review into three sections (the target, its left contexts, and its right contexts) and used neural pooling functions and sentiment lexicon to extract the feature vector for a given target.
4.2. Neural Network for Aspect-Level Sentiment Classification. Today, neural network approaches are extremely fashionable for many natural language processing tasks and obviously, the field of sentiment classification is no exception. Many sentence/document-level sentiment classification tasks are dominated by neural network architectures [23–25]. To further incorporate context information with target information, several models have been proposed, such as the target-dependent LSTM [9], which models each sentence toward the aspect. Attn-LSTM and AT-LSTM [11] are attentional models inspired by [26]. AT-LSTM can be considered as a modification of the neural attention proposed in [26] for entailment detection, swapping the premise’s last hidden state for the aspect embedding. Han et al. [27] proposed a novel neural network based on LSTM and the attention mechanism for word context extraction and document representation. Chen et al. [28] combined regional long short-term memory and convolutional neural network for target-based sentiment classification. Zhang et al. [29] introduced dynamic memory networks based on multiple attention mechanism and LSTM, which showed a significant performance in aspect-level sentiment classification. Yang et al. [30] designed a coattention-LSTM network based on coattention mechanism for aspect-based sentiment analysis, combining the target and context attention vectors of sentences. The work most relevant to ours is IAN [12], which models the sentence and aspect term using two LSTM networks, respectively. It uses the hidden states from the sentence to generate an attention vector for the aspect, and vice versa. Based on these two attention vectors, it outputs a sentence representation and an aspect representation for classification.

Despite these aforementioned methods are effective, discriminating different sentiment polarities for different targets is still a challenging issue. Therefore, it is necessary to design a powerful neural network for aspect-level sentiment classification.

5. Conclusions

In this study, we have proposed an interactive neural network for aspect-level sentiment classification. The approach uses Bi-LSTM and an attention mechanism to interactively learn the important words in the target and context and generates the review representation for the final sentiment classification. Experimental results on the SemEval 2014 dataset show that our method achieves significant improvements. Our model analysis also shows that different sequence models can discriminatively learn the important words in the context and in the target. Furthermore, our model cannot handle several error cases effectively.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was partially supported by the National Social Science Foundation of China under grant no. 17BWX071, the National Natural Science Foundation of China under grant no. 61562057, and the Technology Program of Gansu Science and Technology Department under grant no. 18JR3RA104.

References

[1] B. Liu, “Sentiment analysis and opinion mining,” Synthesis Lectures on Human Language Technologies, vol. 5, no. 1, pp. 1–167, 2012.
[2] L. Jiang, M. Yu, M. Zhou, X. Liu, and T. Zhao, “Target-dependent twitter sentiment classification,” in Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pp. 151–160, Portland, OR, USA, June 2011.
[3] K. Yoon, “Convolutional neural networks for sentence classification,” in Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1746–1751, Doha, Qatar, October 2014.
[4] X. Zhang, J. Zhao, and Y. LeCun, “Character-level convolutional networks for text classification,” 2015, https://arxiv.org/abs/1509.01626.
[5] R. Johnson and T. Zhang, “Effective use of word order for text categorization with convolutional neural networks,” 2014, https://arxiv.org/abs/1412.1058.
[6] D. Tang, B. Qin, and T. Liu, “Document modeling with gated recurrent neural network for sentiment classification,” in Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pp. 1422–1432, Lisbon, Portugal, September 2015.
[7] L. Dong, F. Wei, C. Tan, D. Tang, M. Zhou, and K. Xu, “Adaptive recursive neural network for target-dependent twitter sentiment classification,” in Proceedings of the 52th Annual Meeting of the Association for Computational Linguistics, pp. 49–54, Baltimore, MD, USA, June 2014.
[8] D.-T. Vo and Y. Zhang, “Target dependent twitter sentiment classification with rich automatic features,” in Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI), pp. 1347–1353, Buenos Aires, Argentina, July 2015.
[9] D. Tang, B. Qin, X. Feng, and T. Liu, “Effective LSTMs for target-dependent sentiment classification,” in Proceedings of the International Conference on Computational Linguistics, pp. 3298–3307, Osaka, Japan, December 2016.
[10] M. Zhang, Y. Zhang, and D.-T. Vo, “Gated neural networks for targeted sentiment analysis,” in Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, pp. 3087–3093, Phoenix, AZ, USA, February 2016.
[11] Y. Wang, M. Huang, X. Zhu, and L. Zhao, “Attention-based LSTM for aspect-level sentiment classification,” in Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 606–615, Austin, TX, USA, November 2016.
[12] D. Ma, S. Li, X. Zhang, and H. Wang, “Interactive attention networks for aspect-level sentiment classification,” in Proceedings of the Twenty-Sixth International Joint Conference on
B. Huang, Y. Ou, and K. M. Carley, “Aspect level sentiment classification with attention-over-attention neural networks,” in Proceedings of the 11th International Conference on Social, Cultural, and Behavioral Modeling, SBP-BRiMS 2018, pp. 197–206, Washington, DC, USA, July 2018.

S. Zheng and R. Xia, “Left-center-right separated neural network for aspect-based sentiment analysis with rotary attention,” https://arxiv.org/abs/1802.00892.

Y. Bengio, R. Ducharme, P. Vincent et al., “A neural probabilistic language model,” Journal of Machine Learning Research, vol. 3, no. 6, pp. 1137–1155, 2003.

A. Graves, A.-R. Mohamed, and G. Hinton, “Speech recognition with deep recurrent neural networks,” in Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), no. 3, pp. 6645–6649, Vancouver, Canada, May 2013.

S. Hochreiter and J. Schmidhuber, “Long short-term memory,” Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997.

M. Pontiki, D. Galanis, I. Pavlopoulos, H. Papageorgiou, I. Androustopoulos, and S. Manandhar, “SemEval-2014 task 4: aspect based sentiment analysis,” in Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), pp. 27–35, Dublin, Ireland, August 2014.

J. Pennington, R. Socher, and C. D. Manning, “Glove: global vectors for word representation,” in Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543, Doha, Qatar, October 2014.

D. Tang, B. Qin, and T. Liu, “Aspect level sentiment classification with deep memory network,” in Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 214–224, Austin, TX, USA, November 2016.

B. Pang and L. Lee, “Opinion mining and sentiment analysis,” Foundations and Trends in Information Retrieval, vol. 2, no. 1-2, pp. 1–135, 2008.

S. Kiritchenko, X. Zhu, C. Cherry, and S. Mohammad, “NRC-Canada-2014: detecting aspects and sentiment in customer reviews,” in Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), pp. 437–442, Dublin, Ireland, August 2014.

J. Bradbury, S. Merity, C. Xiong, and R. Socher, “Quasi-recurrent neural networks,” 2016, https://arxiv.org/abs/1611.01576.

Q. Qiao, M. Huang, J. Lei, and X. Zhu, “Linguistically regularized LSTM for sentiment classification,” in Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, pp. 1679–1689, Vancouver, Canada, July 2017.

K. S. Tai, R. Socher, and C. D. Manning, “Improved semantic representations from tree-structured long short-term memory networks,” 2015, https://arxiv.org/abs/1503.00075.

T. Rocktäschel, E. Grefenstette, K. M. Hermann, T. Kočiský, and P. Blunsom, “Reasoning about entailment with neural attention,” 2015, https://arxiv.org/abs/1509.06664.

H. Han, X. Bai, and P. Li, “Augmented sentiment representation by learning context information,” Neural Computing and Applications, vol. 31, no. 12, pp. 8475–8482, 2018.

S. Chen, C. Peng, L. Cai, and L. Guo, “A deep neural network model for target-based sentiment analysis,” in Proceedings of the 2018 International Joint Conference on Neural Networks (IJCNN), Rio de Janeiro, Brazil, July 2018.

Z. Zhang, L. Wang, Y. Zou, and C. Gan, “The optimally designed dynamic memory networks for targeted sentiment classification,” Neurocomputing, vol. 309, pp. 36–45, 2018.
