A Joint Model for Sentiment Classification and Opinion Words Extraction

Dawei Cong[1], Jianhua Yuan[1], Yanyan Zhao[⋆2], and Bing Qin[1]

[1] Research Center for Social Computing and Information Retrieval
Harbin Institute of Technology, China
{dwcong, jhyuan, bqin}@ir.hit.edu.cn
[2] Department of Media Technology and Art, Harbin Institute of Technology, China
yyzhao@ir.hit.edu.cn

Abstract. In recent years, mining opinions from customer reviews has been widely explored. Aspect-level sentiment analysis is a fine-grained subtask, which aims to detect the sentiment polarity towards a particular target in a sentence. While most previous works focus on sentiment polarity classification, opinion words towards the target are also very important for that they provide details about target and contribute to judging polarity. To this end, we propose a hierarchical network for jointly modeling aspect-level sentiment classification and word-level opinion words extraction. Our joint model acquires superior performance in opinion words extraction and achieves comparable results in sentiment polarity classification on two datasets from SemEval 2014.

Keywords: aspect-level sentiment analysis · opinion words extraction · neural network · attention mechanism.

1 Introduction

Aspect-level sentiment analysis[12][8][14] has received much attention these years both in academic communities and industry. Given a sentence and a target, aspect-level sentiment analysis aims at inferring the sentiment polarity(i.e. positive, negative, neutral) towards the target in the sentence. Sentiment analysis at

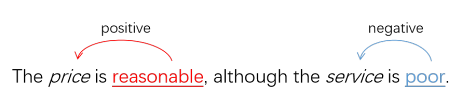

Fig. 1. An example of a review with two aspect terms which have different sentiments. The underlined word are opinion words and point to their corresponding targets.

aspect-level is difficult because the polarity of distinct targets in a sentence may

⋆ Email correspondence.

be different or even opposite. For the example in Figure 1, the sentence expresses a positive sentiment towards aspcet "price" while a negative sentiment to "service". With the development of deep learning, various methods have utilized neural network models[2][6][22][11] to capture relevant information and learn semantic representation for classification automatically. In context of aspect-level sentiment analysis, Target-Dependent LSTM(TD-LSTM) and Target-Connection LSTM(TC-LSTM)[20] take more target information into consideration by modeling contexts surrounding the target string and incorporating a target connection component.

Intuitively, corresponding opinion expression about an aspect plays a vital role in aspect-level sentiment polarity classification. In addition, opinion words provide more information rather than polarity about the aspect. For instance, the aspect "food" may be praised for its taste("delicious") or its freshness("fresh"). With opinion words extracted from the sentence, we will know the specific reasons why polarities towards certain aspects are positive, negative or neutral. However, researchers pay more attention to aspect-level sentiment classification but less attention to opinion words extraction. Recently, Wang et al.[1] notice the importance of opinion words extraction and propose a model combining bidirectional long short-term memory networks(BiLSTM)[3] and conditional random field to capture both polarity and opinion information. Unfortunately, their result of opinion words extraction is not ideal.

To alleviate this, we propose a joint model to solve aspect-level sentiment classification and opinion words extraction. Specifically, we employ two BiLSTM layers to acquire sentiment information. The first LSTM extracts opinion words by their attention weight in the sentence, and the second LSTM captures the semantic information for sentiment classification. Moreover, position information is exploited in the second LSTM to reflect relation between aspect and each other words in sentence. We further optimize our model by adding constraints to loss function.

The main contributions of this work can be summarized as follows:

- We extract opinion words and classify sentiment polarity jointly. Since opinion words play an important part in sentiment classification, we employ attention mechanism to focus on opinion words by adding constraints in loss function.
- We use position information rather than syntactic parser to make connection between aspect and words in sentence, which is computationally efficient. We employ gate mechanism to capture semantic information for sentiment classification, which proves to be effective.
- Experimental results indicate that our approach outperforms several baselines, and we achieve remarkable improvement on opinion words extraction.

The rest of this paper is structured as follows: Section 2 discusses related works, Section 3 gives a detailed description of our proposed model for aspect-level sentiment classification and opinion words extraction. Section 4 compares several model experiments to prove the effectiveness of our proposed model, and Section 5 summarizes this work.

2 Related Work

2.1 Aspect-level Sentiment Classification

In many NLP tasks, earlier approaches mainly include rule-based and traditional machine learning methods. Approaches to sentiment analysis formerly include lexicon-based methods[17][15][4][10] and SVM-based methods[5]. Those methods usually rely heavily on manual features, and models work only if the sets of manual features take effect. However, hand-craft features may be time and labor consuming. Neural Networks(NN) solve the problem by capturing semantic features automatically. Some classical models, such as Recursive Neural Network(RNN)[2][16] and LSTM[18] and Tree-LSTMs[19], are applied to sentiment analysis and prove to be useful. But RNN suffers vanishing gradient and exploding gradient, tree-LSTMs highly depends on the result of syntactic parser. To avoid those problems, LSTM is widely adopted and has shown superior performance. In consideration of target information, TD-LSTM and TC-LSTM[20] average the target words vectors to represent the semantics of target and led to better performance. Wang et al.[24] propose an Attention-based LSTM to explore the connection between an aspect and the content of a sentence. The attention mechanism concentrates on different parts of a sentence when different aspects are given as input. Gated neural network[25] is used to model the interaction between the target mention and its surrounding contexts. However, the above methods only focus on sentiment polarity classification.

2.2 Joint Sentiment Analysis Model

Several subtasks are defined to analyze sentiment at aspect-level, e.g., opinion targets extraction, aspect category detection, etc. Some approaches are proposed to solve the above tasks jointly. Li et al.[7] capture both opinion expressions and the polarity information jointly to extract opinions using sequence labeling by adding sentiment polarity. Zhao et al.[26] model aspect and opinion words jointly for extraction with a MaxEnt-LDA Hybrid. Mitchell et al.[9] extract the sentiment target with its sentiment polarity based on the assumption that surrounding context provides target's sentiment detection with enough information. Wang et al.[1] propose a segmentation based model which can capture the structural dependencies between the target and the sentiment expressions with a linear-chain conditional random field layer. While this model achieves the state-of-the-art performance in sentiment classification, its opinion words extraction result is not good.

To our best knowledge, few research has studied the joint task of sentiment classification and opinion words, and no approach to sentiment classification and opinion words extraction jointly achieves an acceptable result.

3 Methodology

In this section, we introduce our attention-based network approach for aspect level sentiment classification and opinion words extraction. We first give the task definition. Afterwards, we introduce our approach to aspect-level sentiment classification and opinion words extraction.

Given a sentence $s = \{s_1, s_2, ..., s_i, s_{i+1}, ...s_j, ...s_p, s_{p+1}, ...s_q, ...s_n\}$ (actually p may be less than i, order depends on means of expression) consisting of n words and an aspect phrase $a = \{s_i, ...s_j\}$ appearing in sentence s, aspect level sentiment classification aims at predicting sentiment polarity of sentence s towards aspect a, while opinion words extraction detects words $o = \{s_p, ...s_q\}$ implying polarity occurring in sentence s. For example, given s, "The price is reasonable" and a, "price", the sentence express a positive sentiment towards "price" by using the opinion word "reasonable".

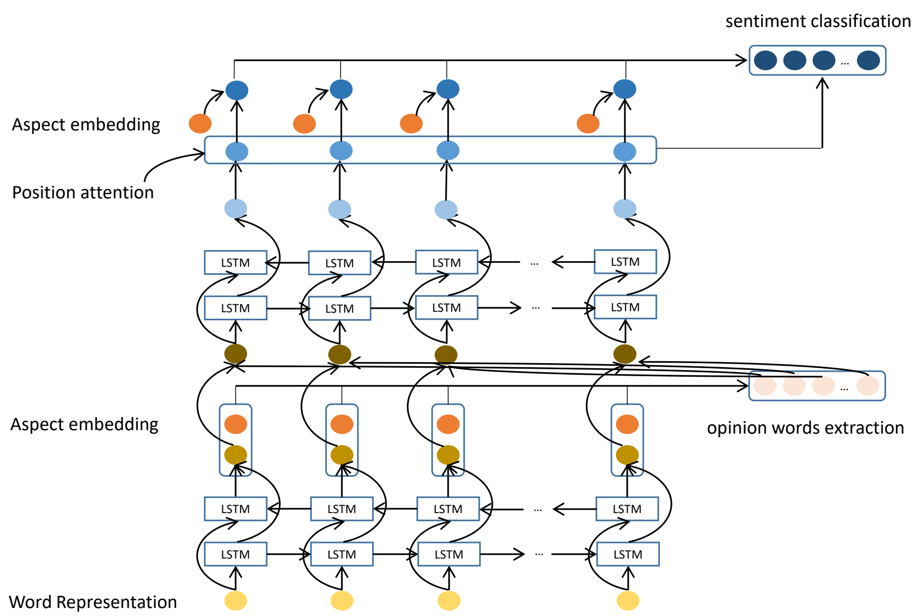

Fig. 2. Our proposed model for aspect-level sentiment classification and opinion words extraction.

Figure 2 presents an overview of our model for aspect level sentiment classification and opinion words extraction. Two BiLSTMs are applied to capture information at different levels and performs its own functions. Specifically speaking, the first BiLSTM acquires sentiment information by paying more attention to opinion words for extraction and feed it to the second BiLSTM. And the second BiLSTM learns semantic representation for sentiment polarity classification according to the output of the first BiLSTM and position information. Next, we will introduce our approach in detail.

3.1 Opinion Words Extraction

Representation of aspects and sentences. Word embedding is widely used for representing meanings of words. Formally, $W \in \mathbb{R}^{d_w \times |W|}$ is made up of all word embeddings, where d_w is dimension of word embedding, and $|W|$ is vocabulary size. To make full use of aspect information, we use aspect embedding for words in an aspect term to represent it just like word embedding. Aspect embedding vectors (unequal to word embedding vectors) of words in an aspect term are averaged and the result is regarded as representation of the aspect. Similar to word embedding, vector $v_{a_i} \in \mathbb{R}^{d_a}$ represents embedding of i-th word a_i in aspect a, where d_a is the dimension of aspect embedding. A is a set of all words in aspect terms. $A \in R^{d_a \times |A|}$ contains all aspect embeddings. Representation of an aspect $a = \{a_1, a_2, ...a_m\}$ contains m words is computed as Equation 1. We will treat the average result v_a as aspect embedding.

$$v_a = \frac{\sum_{k=1}^{m} v_{a_k}}{m} \tag{1}$$

We then fed $\{w_1, w_2, ...w_n\}$ into a bidirectional LSTM to acquire sentence information, next we will introduce how our model incorporates aspect information.

Attention Mechanism for Opinion Words Extraction. As discussed above, more attention should be paid to opinion words about sentiment polarity in sentences. So an aspect-based attention mechanism is employed to compute relatedness of each word with aspect in sentence according to aspect and hidden state vectors of BiLSTM $H = \{h_1, h_2, ...h_n\}$, where n is length of the sentence. An activation function receives aspect information and sentence information to control the propagation of sentiment features. Two linear layers adjust dimensions of hidden and aspect vectors for combination. Finally, a softmax layer uses the vectors to predict attention weights of the input sentence. Let d_{h1} be dimension of hidden state vectors, $H \in \mathbb{R}^{2d_{h1} \times n}$. And v_a represents aspect embedding and $e_N \in \mathbb{R}^n$ is a vector of 1s. The calculation method of attention is as follows.

$$P = W_h H + b_h \tag{2}$$

$$Q = W_a(v_a \otimes e_n) + b_a \tag{3}$$

$$M = tanh(P \oplus Q) \tag{4}$$

$$\alpha = softmax(w^T M) \tag{5}$$

where $M \in \mathbb{R}^{(d_{h1}+d_a) \times n}$, $\alpha \in \mathbb{R}^n$, $W_h \in \mathbb{R}^{d_{h1} \times 2d_{h1}}$, $w \in \mathbb{R}^{d+d_a}$, $W_a \in \mathbb{R}^{d_{h1} \times d_{h1}}$, $b_h, b_a \in \mathbb{R}^{d_{h1}}$, $v_a \otimes e_n$ means concatenates v_a for n times repeatedly and $\oplus$ means element-wise addition. Then we regard words with higher weights in attention α as opinion words. The selection rule is defined as follows.

$$o_i = \begin{cases} true, & if \quad there\ is\ a_j\ in\ a,\ a_i\ /\ a_j\ \geq\ \beta, \\ false, & otherwise. \end{cases} \tag{6}$$

where o_i stands for whether to select w_i for opinion words, β is a hyper parameter.

3.2 Sentiment Polarity Classification

Then we use the other BiLSTM to capture the semantic information for sentiment classification. After last step, we gain weight of every word in sentence and combine it with hidden vectors of the first LSTM in this step.

$$r_i = (\alpha_i \otimes e_{d_{h1}}) \odot h_i \tag{7}$$

where $\alpha_i \otimes e_{d_{h1}}$ means concatenates α_i for d_{h1} times repeatedly, $\odot$ means element-wise multiplication, $\{r_1, r_2, ...r_n\}$ is then fed into the second BiLSTM to learn semantic representation for sentiment polarity and output $\{q_1, q_2, ...q_n\}$. However, not every opinion word occurring in sentence contribute equally to sentiment polarity towards a particular aspect.

Position Information. Intuitively, a context word closer to the aspect should be more important than a farther one. Thus, we make use of position information based on this assumption. The details are described below.

$$p_i = n - l_i \tag{8}$$

$$\gamma = softmax(p) \tag{9}$$

$$g_i = \gamma_i \otimes e_{d_{h2}} \odot q_i \tag{10}$$

where l_i be the distance between aspect and w_i, d_{h2} is dimension of hidden state vectors, $p = \{p_1, p_2, ...p_n\}$ is a sequence of position information, q_i is i-th hidden vector from the second LSTM, $\gamma = \{\gamma_1, \gamma_2, ...\gamma_n\}$ is a sequence of position attention and g_i is semantic representation considering position information for i-th word in the sentence.

Gating Mechanism. The previous attention mechanism with aspect information is employed for opinion words extraction, which extracts opinion words well. In terms of sentiment polarity classification, sentiment-related words should be taken more attention to. We utilize a gating mechanism because it proves to be effective in LSTM for learning semantic information[25].

Here we apply ReLU-Tanh gate to filter information. Let β be attention for semantic of polarity and d_{h2} be dimension of hidden state vectors.

$$\beta_i = sum(relu(W_t q_i + V_t v_a + b_t)) \tag{11}$$

$$U = tanh(W_u(G\beta) + b_u) \tag{12}$$

where $W_t \in \mathbb{R}^{d_{h2} \times 2d_{h2}}$, $V_t \in \mathbb{R}^{d_{h2} \times d_{h2}}$, $b_t \in \mathbb{R}^{d_{h2}}$, $W_u \in \mathbb{R}^{d_{h2} \times 2d_{h2}}$, $b_u \in \mathbb{R}^{d_{h2}}$, $G = \{g_1, g_2, ...g_n\}$ and U is last representation of sentiment polarity. Then a softmax layer is employed to transform U to conditional probability distribution.

$$P = softmax(W_c U + b_c) \tag{13}$$

where $W_c \in \mathbb{R}^{|C| \times d_{h2}}$, $b_c \in \mathbb{R}^{|C|}$, C is the collection of sentiment polarity categories and P is conditional probability distribution of sentiment polarity.

3.3 Loss Function

The model is trained in a end-to-end fashion by minimizing the cross entropy error of sentiment classification and smooth l1 loss of opinion words extraction. And we use a hyper parameter η to balance loss of sentiment classification and opinion words extraction. Thus loss of whole model is computed as 14 .

$$loss = \eta * loss_{polarity} + loss_{extraction} \tag{14}$$

Specially, loss function of sentiment classification $loss_{polarity}$ is defined as:

$$loss_{polarity} = -\sum_i \sum_j y_i^j log \hat{y}_i^j \tag{15}$$

where i, j is the index of sentence-aspect pair and class respectively, y_i^j is the probability of predicting y_i as category j by our model and $\hat{y}$ is 1 or 0, indicating whether the correct answer is j.

The calculation method of $loss_{extraction}$ is set as follows.

$$\hat{x}_i = \begin{cases} \frac{1}{z}, & if\ w_i\ is\ an\ opinion\ word, \\ 0, & otherwise. \end{cases} \tag{16}$$

$$loss_{extraction} = \frac{1}{n} \sum_i \sum_j \begin{cases} 0.5 * (x_i^j - \hat{x}_i^j)^2, & if\ |x_i^j - \hat{x}_i^j| < 1, \\ |x_i^j - \hat{x}_i^j| - 0.5, & otherwise. \end{cases} \tag{17}$$

where i, j is the index of sentence-aspect pair and class respectively, x is the attention weight produced by our system, z is number of opinion words in sentence.

4 Experiment

We describe experimental settings and report empirical results in this section.

Table 1. Statistics of the datasets from SemEval 2014.

Dataset	Pos.	Neg.	Neu.
Restaurant-Train	2164	807	637
Restaurant-Test	728	196	196
Laptop-Train	994	870	464
Laptop-Test	341	128	169

4.1 Dataset

We conduct experiment on datasets from SemEval 2014 task 4[14] to verify the effectiveness of our approach. The datasets consist of customers reviews from two domains, namely restaurant and laptop. Each review contains aspects and corresponding polarities. Following previous work[21], we remove conflict category when preprocessing. Statistics of the datasets are given in Table 1. We also make use of the additional annotations for these two datasets from[23] which contain manually annotated labels for opinion words.

4.2 Experimental Setting

In our experiments, all word vectors are initialized by Glove[13]. The aspect vectors are initialized by sampling from a uniform distribution $U(-0.01, 0.01)$. The dimension of word vectors, aspect embedding and the size of hidden layer are 300. On account of the size of corpus, we set different number of layer and dropout for two datasets: 1 layer of first LSTM for opinion words extraction and 2 layers of second LSTM for sentiment classification in restaurant dataset, both 1 layer in laptop dataset. PyTorch[3] is used for implementing our neural network models. We optimize our models using Adam with initial learning rate of 1e-4, weight delay of 5e-5. The batch size for training is set to 15. Two hyper parameters β, η are 5 and 7 respectively. About one-sixth of training data in restaurant and one-seventh in laptop is left out as the validation set for tuning hyper parameters and selecting model.

4.3 Comparison with Other Methods

We describe empirical results on sentiment classification and opinion words extraction in this section. In terms of sentiment polarity classification, we compare with the following baseline methods on both datasets. We use the same Glove word vectors for fair comparison.

(1) **Majority** is a basic baseline method, which assigns the majority sentiment label in training set to each instance in the test set.

(2) We compare with three LSTM models[20]. In **LSTM**, a LSTM based recurrent model reads the start to the end of a sentence, and the last hidden vector is used as the sentence representation. **TDLSTM** takes aspect information into

[3] https://pytorch.org/

Table 2. Classification accuracy(%) of different methods on laptop and restaurant datasets.

	Laptop	Restaurant
Majority	53.45	65.00
LSTM	66.45	74.28
TDLSTM	68.13	75.63
TDLSTM+ATT	66.24	74.31
SA-LSTM-P	75.1	81.6
Our method	67.29	77.07

consideration by using two LSTM networks, a forward one and a backward one, towards the aspect. **TDLSTM+ATT** extends TDLSTM by incorporating attention mechanism over the hidden vectors.

(3) To the best of our knowledge, **SA-LSTM-P**[1] is the only existing model that supports sentiment classification and opinion words extraction jointly, which achieves state-of-the-art on aspect level sentiment classification.

Experimental results of sentiment polarity classification on review datasets from SemEval 2014 task 4 are given in Table 2.

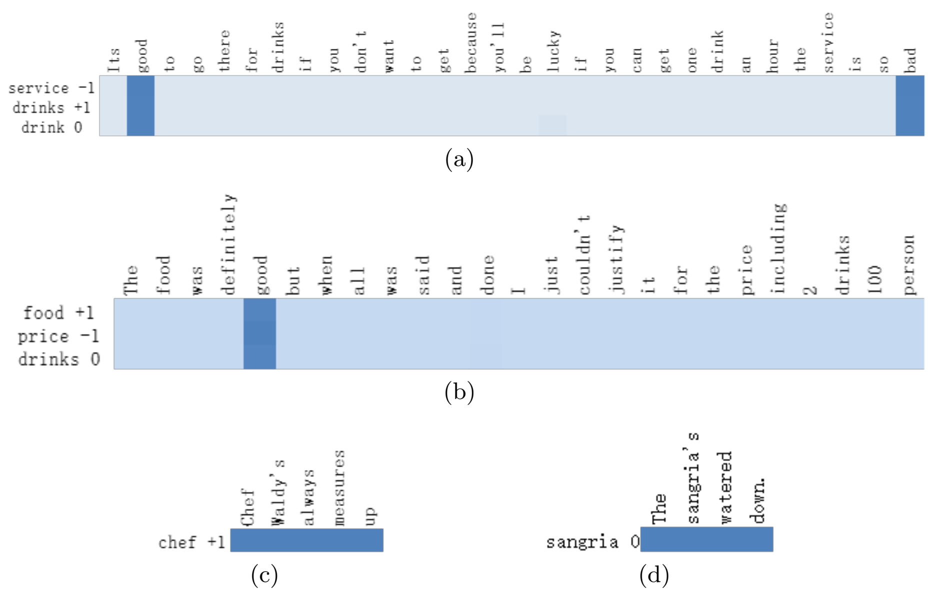

Fig. 3. Visualization of sentiment classification and opinion words results from our model. "+1", "0" and "-1" on the aspect indicate positive, negative and neutral sentiment respectively.

Table 3. Performance (%) of opinion words extraction at word level on laptop and restaurant datasets.

	Laptop			Restaurant		
	P	R	F	P	R	F
SA-LSTM-P	49.1	58.7	53.5	39.4	59.9	55.1
Our method	62.3	65.4	63.8	66.6	73.8	70.0

As shown in Table 3, our model achieves better performance in opinion words extraction. Although our sentiment classification accuracy is a little worsen than SA-LSTM-P, it significantly improves result of opinion words extraction.

4.4 Case studies

To further show the advantages of our model, we visualize some examples from test data. As we can see from the first example in Figure 3(a), the model successfully extracts opinion words in the sentence and distinguishes influence of different opinion words towards different aspects for sentiment polarity classification. Second example in Figure 3(b) shows that when no explicit opinion word for aspect, model captures semantic information from context and classify correctly.

And we also analyze error. Model misunderstands latent sentiment occurring in sentences. It seems difficult to recognize the sentiment in sentences without obvious or usual expression in Figure 3(c) and Figure 3(d).

5 Conclusion

In this work, we propose a hierarchical model of reviews for joint aspect-level sentiment classification and word-level opinion extraction. The key idea is to learn semantic information from two level LSTMs for opinion word extraction and sentiment classification respectively. Experiments on two datasets verify that the proposed approach achieves a comparable result in sentiment polarity classification and makes a significant improvement in extraction at word level.

Acknowledgements

We thank the anonymous reviewers for their valuable suggestions. This work was supported by the National Natural Science Foundation of China (NSFC) via grant 61632011, 61772153 and 71490722.

References

1. Bailin, W., Lu, W.: Learning latent opinions for aspect-level sentiment classification (2018)

2. Dong, L., Wei, F., Tan, C., Tang, D., Zhou, M., Xu, K.: Adaptive recursive neural network for target-dependent twitter sentiment classification. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). pp. 49–54. Association for Computational Linguistics, Baltimore, Maryland (June 2014), http://www.aclweb.org/anthology/P14-2009
3. Graves, A., Mohamed, A.R., Hinton, G.: Speech recognition with deep recurrent neural networks. In: IEEE International Conference on Acoustics, Speech and Signal Processing. pp. 6645–6649 (2013)
4. Kaji, N., Kitsuregawa, M.: Building lexicon for sentiment analysis from massive collection of html documents. In: Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (2007)
5. Kiritchenko, S., Zhu, X., Cherry, C., Mohammad, S.: Nrc-canada-2014: Detecting aspects and sentiment in customer reviews. In: International Workshop on Semantic Evaluation. pp. 437–442 (2014)
6. Lakkaraju, H., Socher, R., Manning, C.D.: Aspect specific sentiment analysis using hierarchical deep learning (2014)
7. Li, F., Han, C., Huang, M., Zhu, X., Xia, Y.J., Zhang, S., Yu, H.: Structure-aware review mining and summarization. In: International Conference on Computational Linguistics. pp. 653–661 (2010)
8. Liu, B.: Sentiment Analysis and Opinion Mining. Morgan Claypool Publishers (2012)
9. Mitchell, M., Aguilar, J., Wilson, T., Durme, B.V.: Open domain targeted sentiment (2014)
10. Mohammad, S.M., Kiritchenko, S., Zhu, X.: Nrc-canada: Building the state-of-the-art in sentiment analysis of tweets. Computer Science (2013)
11. Nguyen, T.H., Shirai, K.: Phrasernn: Phrase recursive neural network for aspect-based sentiment analysis. In: Conference on Empirical Methods in Natural Language Processing. pp. 2509–2514 (2015)
12. Pang, B.: Opinion mining and sentiment analysis (foundations and trends(r) in information retrieval)
13. Pennington, J., Socher, R., Manning, C.: Glove: Global vectors for word representation. In: Conference on Empirical Methods in Natural Language Processing. pp. 1532–1543 (2014)
14. Pontiki, M., Galanis, D., Pavlopoulos, J., Papageorgiou, H., Androutsopoulos, I., Manandhar, S.: Semeval-2014 task 4: Aspect based sentiment analysis. Proceedings of International Workshop on Semantic Evaluation at pp. 27–35 (2014)
15. Prez-Rosas, V.: Learning sentiment lexicons in spanish. In: Eighth International Conference on Language Resources and Evaluation (2013)
16. Qian, Q., Tian, B., Huang, M., Liu, Y., Zhu, X., Zhu, X.: Learning tag embeddings and tag-specific composition functions in recursive neural network. In: Meeting of the Association for Computational Linguistics and the International Joint Conference on Natural Language Processing. pp. 1365–1374 (2015)
17. Rao, D., Ravichandran, D.: Semi-supervised polarity lexicon induction. In: Eacl 2009, Conference of the European Chapter of the Association for Computational Linguistics, Proceedings of the Conference, March 30 - April3, 2009, Athens, Greece. pp. 675–682 (2009)
18. S, H., J, S.: Long short-term memory. Neural Computation **9**(8), 1735–1780 (1997)
19. Tai, K.S., Socher, R., Manning, C.D.: Improved semantic representations from tree-structured long short-term memory networks. Computer Science **5**(1), : 36. (2015)

20. Tang, D., Qin, B., Feng, X., Liu, T.: Effective lstms for target-dependent sentiment classification. Computer Science (2015)
21. Tang, D., Qin, B., Liu, T.: Aspect level sentiment classification with deep memory network. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. pp. 214–224. Association for Computational Linguistics, Austin, Texas (November 2016), https://aclweb.org/anthology/D16-1021
22. Vo, D.T., Zhang, Y.: Deep learning for event-driven stock prediction. In: Proceedings of IJCAI. BueNos Aires, Argentina (August 2015)
23. Wang, W., Pan, S.J., Dahlmeier, D., Xiao, X.: Recursive neural conditional random fields for aspect-based sentiment analysis. CoRR **abs/1603.06679** (2016), http://arxiv.org/abs/1603.06679
24. Wang, Y., Huang, M., Zhu, X., Zhao, L.: Attention-based lstm for aspect-level sentiment classification. In: Conference on Empirical Methods in Natural Language Processing. pp. 606–615 (2016)
25. Zhang, M., Zhang, Y., Vo, D.T.: Gated neural networks for targeted sentiment analysis (2016)
26. Zhao, W.X., Jiang, J., Yan, H., Li, X.: Jointly modeling aspects and opinions with a maxent-lda hybrid. In: Conference on Empirical Methods in Natural Language Processing. pp. 56–65 (2010)