Aspect Specific Opinion Expression Extraction Using Attention Based LSTM-CRF Network

Abhishek Laddha[1] and Arjun Mukherjee[2](✉)

[1] Indian Institute of Technology Delhi, New Delhi 110016, India
[2] Department of Computer Science, University of Houston, Houston, TX, USA
arjun@cs.uh.edu

Abstract. Opinion phrase extraction is one of the key tasks in fine-grained sentiment analysis. While opinion expressions could be generic subjective expressions, aspect specific opinion expressions contain both the aspect as well as the opinion expression within the original sentence context. In this work, we formulate the task as an instance of token-level sequence labeling. When multiple aspects are present in a sentence, detection of opinion phrase boundary becomes difficult and label of each word depend not only upon the surrounding words but also with the concerned aspect. We propose a neural network architecture with bidirectional LSTM (Bi-LSTM) and a novel attention mechanism. Bi-LSTM layer learns the various sequential pattern among the words without requiring any hand-crafted features. The attention mechanism captures the importance of context words on a particular aspect opinion expression when multiple aspects are present in a sentence via location and content based memory. A Conditional Random Field (CRF) model is incorporated in the final layer to explicitly model the dependencies among the output labels. Experimental results on Hotel dataset from Tripadvisor.com showed that our approach outperformed several state-of-the-art baselines.

Keywords: Sentiment analysis · Opinion expressions · Conditional random field

1 Introduction

Aspect based sentiment analysis [15] is one of the main frameworks for fine-grained sentiment analysis and is used in several downstream tasks such as opinion summarization, extracting opinion targets, opinion holders, opinion expressions etc. One of the main goals of aspect based sentiment analysis is to identify the fine-grained product properties (aspects) and their opinion. In [10,16,24] the aspect term and opinion words are jointly extracted but lack correspondence between the aspect and opinion terms. For example, in the sentence *"the food was excellent and plentiful* and the waitstaff was extremely friendly and helpful",

A. Gelbukh (Ed.): CICLing 2018, LNCS 13397, pp. 442–454, 2023.
https://doi.org/10.1007/978-3-031-23804-8_34

discovering aspect words as {*food, waitstaff*} and opinion words as {*excellent, plentiful etc.*} is definitely useful but being able to extract phrases that retain the sentence context as aspect specific opinion expressions such as (*food was excellent and plentiful*, *waitstaffwas extremely friendly and helpful*) would be more expressive and provide more information about the aspect. These opinion phrases can be further used in downstream application such as aspect sentiment classification, aspect summarization.

Traditionally, subjective expression extraction [2,3] has been formulated as a token-level sequence labeling task and has employed a CRF based approach using hand-crafted features. Recent success of distributed representation of words [18,20] provides alternate approach to learn the continuous valued dense vectors for latent features in hidden layers. [9,16] apply deep Recurrent Neural Network (RNN) to extract the opinion expression and opinion target from sentences. They have shown that a deep RNN approach outperforms traditional CRF and semi-CRF. Approaches in [9,16] learn the opinion phrase representation based on the latent features learned of context words but are incapable of explicitly providing the cues from the aspect word. [22,23,28] proposed models to extract the sentiment of an aspect in a sentence by taking into account the aspect words. They are mainly focusing on positive or negative sentiment instead of generic opinion phrase about aspect.

In this paper, we present a neural network architecture with Bi-LSTM and an attention mechanism to take into account the aspect cues. Bi-LSTM layer learns the various sequential patterns among the words without requiring any hand-crafted features. Most of the current work in aspect classification [22,23,28] assumes presence of one aspect in the sentence. If there are multiple aspects in the same sentence they consider them as separate instance ignoring the effect of one aspect on another. We believe that if there are multiple aspects in a sentence, explicitly feeding the importance of context word based on the content and location for a particular aspect is an essential signal to decide whether a context word is in opinion expression of aspect or not.

Inspired by recent success of attention based computational model as in aspect sentiment classification [23,28], machine translation [1], we propose an attention mechanism which takes into account the multiple aspects in the sentence based on the context/surrounding word's location from multiple aspect word. This layer would be helpful in tagging the words which are in between the two aspect and can be included into both aspect opinion expression thereby helping in locating the precise boundary of aspect specific opinion phrases. A CRF model is incorporated in the final layer to explicitly model the dependencies among the output labels.

2 Related Work

We briefly review the existing studies on subjective expression extraction task and aspect-based sentiment analysis using neural networks in this section.

2.1 Subjective Expression Extraction

Early works on fine-grained opinion extraction [2,3] have used various parsing, syntactic, lexical and dictionary based features to extract a subjective expression employing a CRF based approach. Various features based on dependency relations [11] and opinion lexicon have been used for opinion expression extraction. Further, [29,30] employed semi-CRFs which allowed sequence labeling at the segment level. [30] proposed a joint inference model to jointly detect opinion expressions, opinion holders and target as well as relation among them. While they have made important progresses, their performances mainly rely on rich hand crafted features and other pre-processing steps such as dependency parsing.

There have been work exploring the combinations of sequential neural network (e.g. LSTM, RNN) on sequence labeling tasks such as Named Entity Recognition (NER), language understanding. [8,14,31] added a CRF layer on top of RNN network and showed performance improvement on Named Entity Recognition (NER) and language understanding. [17] extended above model by using CNN on character of words to get word level representation. These works have mostly explored the neural networks in NER as opposed to opinion phrase extraction.

The works in [12,26], are the closest to ours as they focus on aspect specific opinion terms. While [26] does not discover phrases, [12] employs higher order CRF features for phrase extraction and is considered as a baseline.

2.2 Neural Network for Aspect Based Sentiment Analysis

Recent studies have shown that deep learning models can automatically learn the inherent semantics and syntactic information from data and this achieves better performance for sentiment analysis. Regarding aspect based sentiment analysis [16,27,32] models target aspect term extraction as a sequence tagging task using neural network. In [16], RNN and word embedding were combined to extract explicit aspects. In [27], recursive neural network based on dependency tree and CRF were integrated in a unified framework to extract the aspect and opinion terms. [32] used word and dependency paths embeddings as features in CRF. These methods are mostly focused on aspect term extraction instead of aspect specific opinion expression.

Also related are the works around aspect based sentiment classification [22,23,28] and the work in [19] which proposed an extension of RNN to identify the aspect's sentiment. [28] proposed an LSTM model with attention mechanism which focuses on different part of sentences given the aspect input. Further, in [23], a deep memory network was proposed for explicitly capturing the importance of each context word when inferring the polarity of given aspect. These approaches provide the sentiment about the aspect but do not give in-depth information about the aspect such as the associated opinion expression.

3 Background: LSTM-CRF Model

This section briefly describes bidirectional LSTM-CRFs that lays the foundation for the proposed attention based LSTM-CRF network. For more details, refer to [8,14,17].

3.1 Bidirectional LSTM Network

Recurrent Neural Networks (RNNs) are a family of neural networks that take an input $X = \{x_1, \ldots, x_T\}$ and yield the sequence hidden representation $\{h_1, \ldots, h_T\}$ where each $h_t \in \mathbb{R}^{d \times 1}$ represents the semantic information of the left context of x_t. In practice these models fail to capture the long term dependencies and suffer from the vanishing gradient problem. [6] proposed an LSTM cell which eliminates these problems by employing several gates to control the information flow in the cell. For each word of a given sentence, an LSTM computes a representation $\overrightarrow{h_t}$ of the left context of sentence. Similarly, a right context information $\overleftarrow{h_t}$ also contains the useful information. This can be achieved by employing another LSTM network which reads the current sentence in the reverse order. Now, the word representation in this bidirectional LSTM would be a concatenation of its left and right latent context representations $h_t = [\overrightarrow{h_t}; \overleftarrow{h_t}] \in \mathbb{R}^{2d \times 1}$.

3.2 LSTM-CRF Model

For the task of sequence tagging, a very simple approach would be to predict the labels y_t independently for each token using a simple feed-forward neural network classification model which takes the output of the LSTM as input vectors. But for labeling of opinion expressions, there is a strong dependency associated with the previous and current label. Therefore, instead of predicting label for each token independently, we model them jointly using CRF [13]. Let's consider $P \in \mathbb{R}^{T \times q}$ to be a matrix of the output of a Bi-LSTM after projecting it on to a linear layer whose size is equal to number of distinct labels q, T is the number of tokens. We also define a transition matrix $A \in \mathbb{R}^{q \times q}$ where each entry $a_{i,j}$ represents a probability of transition from state i to j of consecutive output labels. Then, score for a complete sentence can be defined as follows

$$s(X, y) = \sum_{t=0}^{T} A_{y_t, y_{t+1}} + \sum_{t=1}^{T} P_{t, y_t} \tag{1}$$

where start state and end state is added in the sentence. Since we are considering only bigram interactions among labels, dynamic programming [13] can be used to compute A and the best possible sequence labels during inference [4].

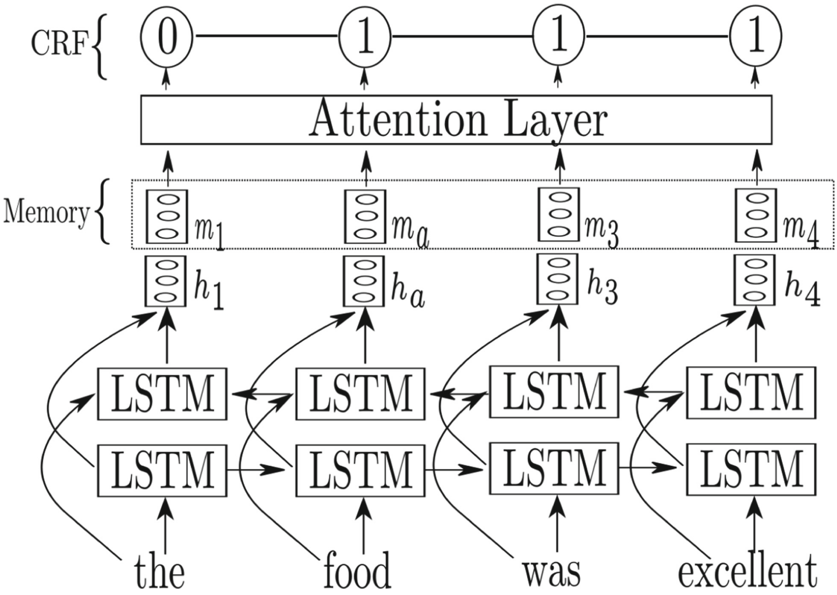

Fig. 1. Main Architecture of network. Word input are given as word embedding to Bi-LSTM.

4 Attention Based LSTM-CRF Network (LSTM-ATT-CRF)

4.1 Task Definition and Notation

Given a sentence $X = (w_1, \dots w_T)$ with T number of words and set of n aspect words $\{w_{a_1}, \dots w_{a_n}\}$ where $a_k \in [1, T]$ is mentioned in sentence X. Our task is to extract set of relevant opinion expression containing the aspect and opinion phrase. We formulated this as sequence labeling problem where each word of a sentence has label $y_t \in \{0, 1\}$. y_t is assigned to 1 if a word is in any aspect specific opinion expression or 0 otherwise. To represent each word, we map it onto a low dimensional continuous vector and corresponding to each word w_t, it's embedding is represented as $v_t \in \mathbb{R}^{d_w \times 1}$ where d_w is the word embedding size. The complete architecture of network is shown in Fig. 1 where Bi-Directional LSTM is similar to network described in the previous section.

4.2 Attention Network

The basic idea behind attention is to assign importance to each context word based on the latent representation and memory. In our setting, memory refers to the multi aspect information present in a sentence. Using the location of the context word from the multiple aspects, memory vector m_t is computed using Eq. 2. Main intuition is that every aspect doesn't contribute equally to determine

	food	was	excellent	and	plentiful	and	the	waitstaff	was	extremely	friendly
Before	attention:										
$P(y=1)$	0.6	0.43	0.79	0.64	**0.46**	0.3	0.2	0.6	0.45	0.76	0.82
$P(y=0)$	0.4	0.57	0.21	0.36	**0.54**	0.7	0.8	0.4	0.55	0.24	0.18
After	attention:										
$P(y=1)$	0.7	0.45	0.86	0.64	**0.56**	0.3	0.2	0.7	0.47	0.83	0.9
$P(y=0)$	0.3	0.55	0.14	0.36	**0.44**	0.7	0.8	0.3	0.53	0.17	0.1

Fig. 2. An illustration of example of our neural attention network for aspect specific opinion labeling. Words in blue and red corresponds to opinion expression about "food", and"waitstaff" respectively. (Color figure online)

the polarity of each context word in the sentence. The words which are distant from the particular aspect word would influence less by that aspect word. In this work, we define the location of context words as it's absolute distance with the aspect in the original sentence. Memory vector for each token at t is defined

$$m_t = \frac{\sum_{k=1}^{n} v_{a_k} \left(1 - \frac{l_{t,k}}{T}\right)}{n} \tag{2}$$

$$g_t = tanh\left(W_{attn}\left[m_t; h_t\right] + b_{attn}\right) \tag{3}$$

where $l_{t,k}$ is number of word between w_t and w_{a_k}, $W_{attn} \in \mathbb{R}^{(2d+d_w)\times 1}$, v_{a_k} is the embedding vector of aspect a_k. Based on the memory vector and hidden representation, the model assigns an score g_t to each context word using Eq. 3 which takes into account the relation between word and multiple aspects. After getting the g_t's we feed them to softmax function to get the importance weights α_t for each context word, such that $\sum_t \alpha_t = 1$ and each $\alpha_t \in [0, 1]$.

$$P_t = W_{linear}^{Tr}(\alpha_t \times h_t) + b_{linear} \tag{4}$$

A linear layer is applied to transform the hidden representation vector to the scores of each output tag using the Eq. 4, where $W_{linear} \in \mathbb{R}^{2d\times q}$ and $b_{linear} \in \mathbb{R}^{q\times 1}$ and after that score of a sequence was calculated using Eq. 1. Here, P_{t,y_t} is unscaled probability of word w_t having label y_t. In the absence of attention α_t will become 1 however, attention will provide weighted hidden state with respect to aspect words. Figure 2 shows a example where word "plentiful" have low confidence in inclusion of opinion expression due to long distance from aspect word "food" and closeness to aspect word "waitstaff" for which it doesn't express opinion. While attention will learn to give more importance to those words because there is direct interaction of hidden vector with aspect word and there are lots of opinion expression about aspect "food" which includes the words of similar meaning as of "plentiful".

4.3 Model Training

The model can be trained end-to end using backpropogation, where the objective function is to maximize the log-probability of correct sequence score as defined in Eq. 5.

$$p(y|X) = \frac{exp(s(X, y))}{\sum_{\hat{y}} exp(s(X, \hat{y}))} \quad (5)$$

where X denotes the sequence of words and y is the corresponding sequence label. $s(X, y)$ is score defined in Eq. 1, P_t learns the probability of each label independently from Bi-LSTM while A learns the dependency among the various labels. For *e.g.,* in Fig. 2 some stopwords such as "was" could have low probability of label $y_t = 1$, but decoding complete sequence using Eq. 1 will consider surrounding label and hence inclusion of such words in opinion expression. Model parameters are the LSTM weights, $W_{attn}, W_{linear}, b_{attn}, b_{linear}, A$ and word embeddings. Except word embeddings, other parameters are initialized using sampling from uniform distribution $U(-\epsilon, \epsilon)$, where $\epsilon = 0.01$.

Word Embeddings: Word embeddings are initialized using the pre-trained embeddings. We use Stanford's publicly available GloVe[1] 100-dimensional embeddings [20]. We also experimented with two other embeddings, namely Senna[2] 50-dimensional embeddings [4] and Google's word2vec[3] 300-dimensional [18]. Parameter d_w depends on dimension of pre-trained word embedding vectors.

Features: Although NNs learn the word features (i.e. embedding) automatically, [16] shows that incorporating other linguistic features like part of speech (POS) and syntactic information (e.g., phrase chunks) helps in the training to learn a better model. We used the syntactic features (POS tags) and phrase chunk features as input in the LSTM network. Similar to word embedding, each feature is also mapped to feature embedding which gets learned during training. Input to LSTM network is a concatenation of word embedding and feature embeddings.

5 Experiments

5.1 Dataset

To demonstrate the effectiveness of our model we performed experiments on the Hotel dataset from Tripadvisor.com used in [25]. Labeling opinion phrases for each aspect is a tedious task and require lots of human effort. Further, training deep learning model generally needed substantial amount of training data. To overcome this bottleneck, we tried to label the phrases using heuristic rules. Seed words for each aspect used in [25] were used to label the location of aspect words in review sentences. Once, aspect words in sentence were determined, opinion expressions around those aspect words are labeled as described next.

Labeling Using Heuristic Rules. Since, we mainly focus on opinion expressions surrounding the aspect word, heuristic rules could be written with the help of Part of Speech (POS) tags and polarity of the surrounding words. We used

[1] http://nlp.stanford.edu/projects/glove/.

[2] http://ronan.collobert.com/senna/.

[3] https://code.google.com/archive/p/word2vec/.

the opinion lexicon[4] derived from [7] for positive and negative words. Labeling of the phrase boundary surrounding the aspect word for positive opinions phrases was done as follows:

In the first step, we searched for the positive terms (using the sentiment lexicon) in the window of $[-5, 5]$ around the aspect word. To have a compact opinion phrase, it should not include the opinion about the other aspects. So, only considering the extremes with a sequence index of [aspect, positive-term] would not yield good phrases. Therefore, we divided the presence of positive words in two cases. First, when the positive word was before the aspect word, to capture all the opinion words in a phrase talking about noun aspect, we took the farthest adjective from the aspect word. If aspect word was verb, then we took the nearest adverb since adverbs are mostly situated immediately before the verb. Second, when the positive word is after the aspect word, we took the nearest adjective (from the aspect word) because adjectives are generally immediately followed by nouns. We also included the negative phrases with negator words in surrounding window by finding negative terms in window of $[-5, 5]$ and then finding negator terms such as "not", "don't" and following the same procedure described above. This process generally yielded us the minimum boundaries of phrases but could have omitted some opinion words which we included using the method described below.

Next, we extended the phrase boundaries using the basic definition of adjective and adverb: (i) If first word of a phrase is coming before the aspect word and it is a verb, then we look at the word before the verb, if the word was an adverb, then we include it in that phrase, (ii) If the last word of the phrase was an adjective and the next word after that was noun, then we included all the consecutive nouns after that, (iii) If last word of the phrase was an adverb, then we included the next word if it was a verb, (iv) If last word of the phrase was noun, then we included all the consecutive nouns after that. Similar process was applied for extracting negative opinion expressions as well. We explain our procedure using the following *e.g.,* "The room provided a nice view of the lagoon", aspect word is "room" which is noun and opinion word is "nice" which is adjective. Since adjective word is after the aspect word we took the nearest adjective and extracted phrase would be "room provided a nice" which is not complete. Now, we extend this using the rule (ii) which will include the noun word "view" since it immediately follows the adjective. Thus this completes the opinion expression. We observed that following these heuristic rule are able to capture various fluid opinion expression like "wonderful hotel at a reasonable price," and "rooms do feel quite bland".

Dataset Dissection. Using the above procedure, we labeled a total of $10,775$ sentences which was split in $80:20$ ratio for training and validation. We want to evaluate on the dataset which is completely realistic and wanted to test the ability of our model to retrieve phrases which might not have labeled using the heuristic rules. Hence, for testing, we manually labeled another disjoint set of

[4] http://www.cs.uic.edu/~liub/FBS/opinion-lexicon-English.rar.

Table 1. Comparison of results with baselines

Model	Precision	Recall	F-score
CRF	82.77	69.01	75.26
Semi-CRF	84.63	**78.27**	81.29
LSTM-CNN-CRF	88.46	72.47	79.67
LSTM-ATTN-CRF	**88.80**	75.86	**81.82**

1, 683 sentences after getting the location of aspect labeled from the seed words. Dataset will be released to serve as language resource. We preprocessed the data by lowercasing all the word and replaced all cardinal numbers with "NUM" symbol and removed words appearing only once.

5.2 Parameters Settings

Our model was implemented in tensorflow[5] using the Adam optimizer with initial learning rate of .01 and early stopping criteria [5] was used based on validation set accuracy. The decaying learning rate for Adam was set to 0.05. Care was taken to reduce overfitting by adding a dropout layer regularizer [21] with keep probability of 0.5 and gradients were clipped at 5. Other hyperparamters such as dimension of the hidden states of LSTM were kept same for all model $d = 100$, # of layers as 3, batch size as 10, and maximum length of sentence was set to 50 which were determined using pilot experiments.

5.3 Comparison with Baselines

We compared our model with the following most relevant baselines.

CRF: We use linear chain-CRF [13] and higher order features as described in [12].

Semi-CRF: This is due to model in [29] that used dependency tree features with semi-CRF to label the sequence at segment level.

LSTM-CNN-CRF: This is due to model in [17] which used the word and character level representation using combination of LSTM, CNN and CRF for sequence labeling task.

LSTM-ATT-CRF: Our complete proposed model which have attention over the output of Bi-LSTM using aspect memory with CRF layer.

Next, we also explored two simplified versions of our model.

LSTM-ATT: This model employs the cross-entropy between the predicted and target labels for loss instead of maximizing the CRF score.

LSTM-CRF: The concatenated output vectors of Bi-LSTM are passed directly into the linear layer for computing the CRF score.

[5] http://tensorflow.org/.

5.4 Discussion

We used word based micro precision, recall and F-score to evaluate the quality of the model. [30] has shown that it is difficult to get exact opinion expression boundaries even for human annotators and hence focused on precision, recall measures at word level instead of complete expression level. Precision is defined as $\frac{|C \cap P|}{P}$ and recall as $\frac{|C \cap P|}{C}$, where C and P are the correct and predicted set of word labels respectively.

Table 2. Performance on aspect specific opinion expression task on Precision, Recall, F1 for different models when initialized using various pre-trained embeddings

Model	Senna			word2vec			Glove		
	P	R	F	P	R	F	P	R	F
LSTM-CNN-CRF	88.46	72.47	79.67	89.93	71.18	79.46	87.35	73.15	79.62
LSTM-ATT-CRF	**88.80**	75.86	**81.82**	88.40	75.08	81.20	87.73	76.30	81.62
LSTM-ATT	87.3	73.31	79.58	88.22	74.57	80.82	87.4	74.84	80.67
LSTM-CRF	88.14	75.94	81.59	88.68	74.13	80.75	87.98	75.40	81.2

Table 3. Example of attention weight for different example. Underlined words are aspect words, weights colored in blue are probably correct while weights in red are wrong. True opinion expression are included in [[ ]]. Higher weights mean the more probable a word is in opinion expression.

hotel	in	[[excellent	location	close	to	everything]]	we	were	impressed	[[excellent	service	at	reception	upon	arrival]]	
0.021	0.013	0.048	0.066	0.075	0.026	0.075	0.015	0.015	0.067	0.076	0.073	0.06	0.050	0.031	0.026	
there	are	[[waterfalls	in	lobby	area]]	and	[[free	easy	fast	internet	access]]	but	only	in	lobby	area
0.027	0.015	0.0187	0.024	0.107	0.066	0.083	0.103	0.106	0.103	0.04	0.026	0.038	0.019	0.016	0.027	0.016

Table 1 illustrates the comparison results of baselines with our best model LSTM-ATT-CRF. Our model significantly outperforms CRF and LSTM-CNN-CRF on F-score. It also improves over semi-CRF at $p < 0.05$ semi-CRF performs close to our model due to the fact that many opinion phrases are noun phrases (NPs) and verb phrases (VPs), and its use of segment level labeling greatly improved recall but it suffers in precision.

Further from Table 2, we note LSTM-ATTN-CRF outperforms character based LSTM-CNN-CRF and LSTM-CRF across all word embedding which shows including the aspect information using the attention is effective even when features based on POS and syntactic tags (phrase chunk units) are included as input. Our complete model outperform LSTM-ATT significantly which shows that adding the CRF layer to capture the dependency among output label is useful. Also interesting to note is that Senna embeddings perform best for aspect specific opinion expression. This is due to the fact that Senna was trained on various NLP task such as NER, POS and SRL whereas other Glove, word2vec were trained on general co-occurence of words.

For a yet deeper understanding of the attention mechanism, Table 3 shows the attention weights for two examples. Weights of context words around aspect confirms that our attention mechanism is able to assign the weights based on the word and aspect. Reason for incorrect word such as ‘waterfall’ is due to their low occurrences in the corpus while others are stopwords which sometimes get included in the opinion expression. Our model is able to assign the substantial weight to many neutral words such as ‘close’ and ‘everything’ based on the aspect which contributes to its effectiveness over other baselines.

6 Conclusion

In this paper, we presented an attention based LSTM-CRF (LSTM-ATT-CRF) for aspect specific opinion expression task. The model works for both single and multiple aspect sentences and improves phrase discovery by leveraging the latent interactions among the aspect and opinion words based on the content and location which we modeled via attention mechanism. Experimental results on a hotel dataset showed superior performance over several baselines. The work also produced a labeled dataset which shall be released as a resource.

Acknowledgement. This work is supported in part by NSF 1527364.

References

1. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473 (2014)
2. Breck, E., Choi, Y., Cardie, C.: Identifying expressions of opinion in context. In: IJCAI, vol. 7, pp. 2683–2688 (2007)
3. Choi, Y., Cardie, C., Riloff, E., Patwardhan, S.: Identifying sources of opinions with conditional random fields and extraction patterns. In: Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing, pp. 355–362. Association for Computational Linguistics (2005)
4. Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., Kuksa, P.: Natural language processing (almost) from scratch. J. Mach. Learn. Res. **12**(Aug), 2493–2537 (2011)
5. Graves, A., Mohamed, A.R., Hinton, G.: Speech recognition with deep recurrent neural networks. In: 2013 IEEE International Conference on Acoustics, speech and signal processing (ICASSP), pp. 6645–6649. IEEE (2013)
6. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)
7. Hu, M., Liu, B.: Mining and summarizing customer reviews. In: Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 168–177. ACM (2004)
8. Huang, Z., Xu, W., Yu, K.: Bidirectional lstm-crf models for sequence tagging. arXiv preprint arXiv:1508.01991 (2015)
9. Irsoy, O., Cardie, C.: Opinion mining with deep recurrent neural networks. In: EMNLP, pp. 720–728 (2014)

10. Jo, Y., Oh, A.H.: Aspect and sentiment unification model for online review analysis. In: Proceedings of the Fourth ACM International Conference on Web Search and Data Mining, pp. 815–824. ACM (2011)
11. Johansson, R., Moschitti, A.: Extracting opinion expressions and their polarities: exploration of pipelines and joint models. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2, pp. 101–106. Association for Computational Linguistics (2011)
12. Laddha, A., Mukherjee, A.: Extracting aspect specific opinion expressions. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pp. 627–637 (2016)
13. Lafferty, J., McCallum, A., Pereira, F., et al.: Conditional random fields: probabilistic models for segmenting and labeling sequence data. In: Proceedings of the Eighteenth International Conference on Machine Learning, ICML, vol. 1, pp. 282–289 (2001)
14. Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., Dyer, C.: Neural architectures for named entity recognition. arXiv preprint arXiv:1603.01360 (2016)
15. Liu, B.: Sentiment analysis and opinion mining. Synth. Lect. Hum. Lang. Technol. **5**(1), 1–167 (2012)
16. Liu, P., Joty, S.R., Meng, H.M.: Fine-grained opinion mining with recurrent neural networks and word embeddings. In: EMNLP, pp. 1433–1443 (2015)
17. Ma, X., Hovy, E.: End-to-end sequence labeling via bi-directional LSTM-CNNs-CRF. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Berlin, Germany, pp. 1064–1074. Association for Computational Linguistics, August 2016. http://www.aclweb.org/anthology/P16-1101
18. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems, pp. 3111–3119 (2013)
19. Nguyen, T.H., Shirai, K.: Phrasernn: phrase recursive neural network for aspect-based sentiment analysis. In: EMNLP, pp. 2509–2514 (2015)
20. Pennington, J., Socher, R., Manning, C.D.: Glove: global vectors for word representation. In: EMNLP, vol. 14, pp. 1532–1543 (2014)
21. Srivastava, N., Hinton, G.E., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. **15**(1), 1929–1958 (2014)
22. Tang, D., Qin, B., Feng, X., Liu, T.: Effective LSTMS for target-dependent sentiment classification. arXiv preprint arXiv:1512.01100 (2015)
23. Tang, D., Qin, B., Liu, T.: Aspect level sentiment classification with deep memory network. arXiv preprint arXiv:1605.08900 (2016)
24. Titov, I., McDonald, R.: Modeling online reviews with multi-grain topic models. In: Proceedings of the 17th International Conference on World Wide Web, pp. 111–120. ACM (2008)
25. Wang, H., Lu, Y., Zhai, C.: Latent aspect rating analysis on review text data: a rating regression approach. In: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 783–792. ACM (2010)
26. Wang, S., Chen, Z., Liu, B.: Mining aspect-specific opinion using a holistic lifelong topic model. In: Proceedings of the 25th International Conference on World Wide Web, pp. 167–176. International World Wide Web Conferences Steering Committee (2016)

27. Wang, W., Pan, S.J., Dahlmeier, D., Xiao, X.: Recursive neural conditional random fields for aspect-based sentiment analysis. arXiv preprint arXiv:1603.06679 (2016)
28. Wang, Y., Huang, M., Zhao, L., Zhu, X.: Attention-based LSTM for aspect-level sentiment classification. In: EMNLP, pp. 606–615 (2016)
29. Yang, B., Cardie, C.: Extracting opinion expressions with semi-markov conditional random fields. In: Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pp. 1335–1345. Association for Computational Linguistics (2012)
30. Yang, B., Cardie, C.: Joint inference for fine-grained opinion extraction. In: ACL (1), pp. 1640–1649 (2013)
31. Yao, K., Peng, B., Zweig, G., Yu, D., Li, X., Gao, F.: Recurrent conditional random field for language understanding. In: 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 4077–4081. IEEE (2014)
32. Yin, Y., Wei, F., Dong, L., Xu, K., Zhang, M., Zhou, M.: Unsupervised word and dependency path embeddings for aspect term extraction. arXiv preprint arXiv:1605.07843 (2016)