A Span-based Joint Model for Opinion Target Extraction and Target Sentiment Classification

Yan Zhou[1,2] , **Longtao Huang**[3] , **Tao Guo**[1] , **Jizhong Han**[1] and **Songlin Hu**[1,2*]
[1]Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
[2]School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
[3]Alibaba Group Turing Lab
zhouyan@iie.ac.cn, kaiyang.hlt@alibaba-inc.com, {guotao, hanjizhong, husonglin}@iie.ac.cn

Abstract

Target-Based Sentiment Analysis aims at extracting opinion targets and classifying the sentiment polarities expressed on each target. Recently, token-based sequence tagging methods have been successfully applied to jointly solve the two tasks, which aims to predict a tag for each token. Since they do not treat a target containing several words as a whole, it might be difficult to make use of the global information to identify that opinion target, leading to incorrect extraction. Independently predicting the sentiment for each token may also lead to sentiment inconsistency for different words in an opinion target. In this paper, inspired by span-based methods in NLP, we propose a simple and effective joint model to conduct extraction and classification at span level rather than token level. Our model first emulates spans with one or more tokens and learns their representation based on the tokens inside. And then, a span-aware attention mechanism is designed to compute the sentiment information towards each span. Extensive experiments on three benchmark datasets show that our model consistently outperforms the state-of-the-art methods.

1 Introduction

Target-Based Sentiment Analysis (TBSA) is a fundamental problem in sentiment analysis [Liu, 2012; Pontiki *et al.*, 2014; Thelwall *et al.*, 2010]. The goal of TBSA is to identify the opinion targets mentioned in a sentence, and predict the sentiment polarity (e.g. positive, neutral, negative) for each opinion target. For example, in sentence "*The hard drive still works well but the left mouse button is broken*", "hard drive" and "left mouse button" are the opinion targets, and the sentiments expressed on them are positive and negative, respectively.

The complete TBSA task involves two subtasks: opinion target extraction (OTE) and target sentiment classification (TSC). There have been many works on the subtask of opinion target extraction [Wang *et al.*, 2017; Li *et al.*, 2018b; Xu *et al.*, 2018]. These researches only extract the opinion targets mentioned in the text, but do not determine the sentiment for the targets. The subtask of target sentiment classification has also been extensively studied [Chen *et al.*, 2017; Fan *et al.*, 2018]. These approaches assume that the opinion targets are given in advance and merely predict the sentiment towards them. From the above we can see that most of the studies only focus on one of the subtasks and process them independently. However, in most of the real-world applications, we need to perform the complete TBSA task, i.e. both opinion target extraction and target sentiment classification. One straightforward approach is to pipeline the models of the two subtasks together. But according to the observation of other tasks [Finkel and Manning, 2009; Li and Ji, 2014], if two subtasks have strong couplings (e.g. NER and relation extraction), a joint method generally perform better than a pipelined method.

Recently, some researches attempted to solve the two subtasks in a joint model [Mitchell *et al.*, 2013; Zhang *et al.*, 2015; Li *et al.*, 2019]. These studies designed a unified tagging scheme for the complete TBSA task: tags {*B, I, E, S*}-{*POS, NEG, NEU*} and tag *O*, where {*B, I, E, S*} represent the token position in an opinion target; {*POS, NEG, NEU*} represent the sentiment polarity of a token; tag *O* denotes a word outside the opinion targets. For example, the tags for the sentence "*The hard drive still works well but the left mouse button is broken*" are "*O, B-POS, E-POS, O, O, O, O, O, B-NEG, I-NEG, E-NEG, O, O*". Based on the defined tags, the complete TBSA task is converted into a token-based sequence tagging problem, where a tag is sequentially assigned to each token in the input sentence.

Though the token tagging based joint methods have achieved better results than the pipelined approaches on the complete TBSA task, they still have some limitations. Firstly, for a target comprised by multiple words, the existing joint methods predict a tag for these words separately. It is difficult to use the global information of the target, which might cause incorrect extraction. For the target "hard drive" in the example sentence, if the model predicts the two words separately, it may regard the meaning of "hard" as "difficult" and regard "drive" as a verb, and thus fail to identify this target. And treating the two words as a whole can provide the global information of the phrase "hard drive", which may help the model to extract this target. Secondly, the token tagging based methods calculate the sentiment information of

*Corresponding author

each token independently, which might lead to sentiment information learned by different words in one opinion target different. Therefore, they may predict the tags of the target "left mouse button" as "B-POS, I-NEG, E-NEG", which is obviously wrong. Existing methods propose some components to maintain the sentiment consistency (i.e. the sentiment towards each word in an opinion target should be the same). However, since these components depend on the matrix weights learned from the training data, they can not completely ensure the sentiment consistency to all the targets.

In this paper, we propose an effective span-based joint model for the complete TBSA task, which can overcome the limitations mentioned above. Inspired by recent span-based models in NLP [Xu *et al.*, 2017; Ouchi *et al.*, 2018; He *et al.*, 2018], our method enumerates all the token spans (up to a certain length) in a sentence and predicts the labels of them. Specifically, our model first learns the contextualized representation of each span which can capture the global information of a span. Then we design an attention mechanism to compute the sentiment information towards each span, and integrate this information to the span. Thus our model can maintain the sentiment consistency to different word in a span. Finally, our model predicts the label of the spans based on the learned representations. The main contributions of this paper are concluded as follows:

- This paper proposes an effective span-based joint model for the complete TBSA task, which can take advantage of the span-level information to identify opinion target. To the best of our knowledge, this is the first span-based model for this task.
- This paper introduces a span-aware attention mechanism to detect the sentiment context of a span. Computing the sentiment information of each span rather than each word, our model can avoid the sentiment inconsistency problem.
- We conduct experiments on three benchmark datasets and the results show our method outperforms the state-of-the-art models.

2 Our Approach

2.1 Task Definition

This paper focuses on the complete TBSA task, which aims to extract opinion targets and predict sentiment polarity of each target at the same time. First of all, we define a set of labels $C = \{TPOS, TNEG, TNEU, O\}$ for the spans. The labels $\{TPOS, TNEG, TNEU\}$ denote a span is an opinion target with positive, negative or neutral sentiment and label O represents a span is not an opinion target. And then we enumerate all the spans in the text (up to a certain length) and predict the labels of them. Formally, given a sentence consists of T words $X = \{w_1, w_2, \ldots, w_T\}$, our goal is to predict a set of labeled spans $Y = \{(i, j, l)|1 \leq i \leq j \leq T; j - i + 1 \leq L; l \in C\}$, where i and j are the word indices in the sentence, l represents the label of the span and L is the maximum length of the spans. We present an example sentence "*The hard drive still works well*" in Table 1. The labeled span (2,3,TPOS) represents that "hard drive" is the opinion target and the sentiment towards this target is positive. The other spans with label "O" indicate that they are not opinion targets. Thus we can get the opinion target of the input sentence is "hard drive" and the sentiment expressed on it is positive.

Input	The_1 hard_2 drive_3 still_4 works_5 well_6
Output	(1,1,O); (2,2,O); (3,3,O); (4,4,O); (5,5,O); (6,6,O); (1,2,O); (2,3,TPOS); (3,4,O); (4,5,O); (5,6,O); (1,3,O); (2,4,O); (3,5,O); (4,6,O);

Table 1: An example for the span-based approach. The maximum length is set to 3.

2.2 Neural Architecture

The architecture of our model is shown in Figure 1. We first adopt a shared stacked bidirectional LSTM (BiLSTM) to learn the word-level contextual information of the input tokens. Then we represent each span by using the contextual representation of the tokens. After that, a span-aware attention mechanism is designed to compute the sentiment context of each span. Finally, the label of a span is predicted based on its span representation and context representation.

Input embedding. Our model represents each token w_t by concatenating its traditional word embedding x_t^w and contextualized word embedding x_t^c:

$$x_t = [x_t^w; x_t^c] \tag{1}$$

We employ word2vec to obtain the traditional word embeddings. Recently, contextualized word embeddings have shown promising results across a range of NLP tasks. To further improve the performance, our model utilizes the contextualized word embeddings ELMo (Embeddings from Language Models) [Peters *et al.*, 2018] which are encoded by the pre-trained ELMo encoders. ELMo is produced by a bidirectional language model that takes characters as input and uses LSTMs to capture contextual information.

Contextual layer. We employ a shared stacked BiLSTM to learn the word-level contexts of the inputs. For a stacked BiLSTM consisting of *M* layers, the hidden states of the *m*-layer ($m \in \{1, \ldots, M\}$) is computed as follows:

$$\begin{aligned} &\left\{h_1^{(m)}, \ldots, h_t^{(m)}, \ldots, h_T^{(m)}\right\} = \\ &BiLSTM^{(m)}\left\{h_1^{(m-1)}, \ldots, h_t^{(m-1)}, \ldots, h_T^{(m-1)}\right\} \end{aligned} \tag{2}$$

where $h_t^{(m)}$ represents the *t*-th hidden state of the *m*-layer. For the first layer of BiLSTM, we utilize the embeddings computed by the input embedding layer as inputs. We use the hidden states of the top layer $\left\{h_1^{(M)}, \ldots, h_T^{(M)}\right\}$ as the contextual representation of the tokens.

Span representation. All the possible spans in the sentence are regarded as the candidates of opinion targets. We use S to denote the set of the spans:

$$S = \{(i, j)|1 \leq i \leq j \leq T; j - i + 1 \leq L\} \tag{3}$$

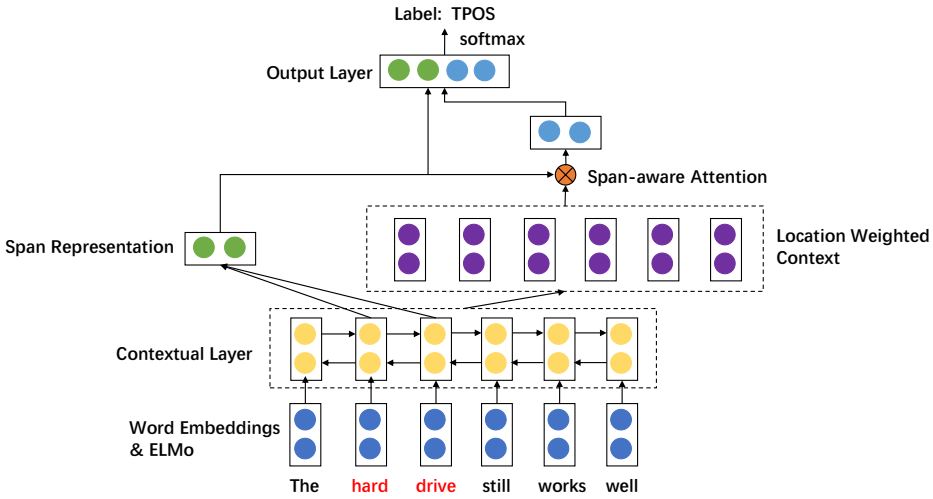

Figure 1: The architecture of the span-based model. For clarity, we only draw one BiLSTM layer and show the prediction for the span "hard drive".

where L is the maximum length of the spans; i and j denote the start position and the end position of the span, respectively. For each span $(i, j) \in S$, we use two types span-level features to capture the global information of it: the boundary representation and the merged representation. In our model, we directly represent the boundary information by the outputs of the contextual layer corresponding to the boundary words. And the merged representation, which contains all the information of the tokens included in a span, is also important to the prediction. Considering the sentence in Figure 1, the meanings of "hard" and "drive" will be combined together when a person identifies "hard drive" as an opinion target. We add the contextual representation of the tokens in a span to represent the merged information of it. We concatenate the span-level features to get the representation of span (i, j) as follows:

$$s_{(i,j)} = [h_i^{(M)}; \sum_{k=i}^{j} h_k^{(M)}; h_j^{(M)}] \tag{4}$$

where $h_i^{(M)}$, $h_j^{(M)}$ and $h_k^{(M)}$ are the outputs from the stacked BiLSTM. Although the span-level features can be easily incorporated into our model, they are difficult to be used in token tagging based joint methods.

Span-aware Attention

The complete TBSA task need to predict the sentiment polarity towards the target simultaneously. From the traditional target sentiment classification task, we know that the key information to determine the sentiment polarity of a target generally lies in the context of it. Take the sentence in Figure 1 as an example, the context word "well" indicates the sentiment expressed on "hard drive" is positive. To learn the sentiment information of each span, we design a span-aware attention mechanism in our model.

Intuitively, the context words closer to a span may have a greater impact to it. We use the location weighted context to simulate this observation. We first define the weight $w_t^{'}$ for each context word w_t according its distance to span (i, j):

$$w_t^{'} = 1 - l_t/T \tag{5}$$

where T is the sentence length and l_t is the distance of word w_t towards the span (i, j). For the words in the span, the distance l_t is set to 0. If the span contains multiple words, the distance l_t is calculated with its left or right boundary index according to which side the word w_t locates. Then the weights are utilize to produce the location weighted context $E = \{e_1, e_2, \ldots, e_T\}$ of span (i, j) and e_t is calculated as:

$$e_t = w_t^{'} * h_t^{(M)} \tag{6}$$

Based on the context E, we adopt a span-aware attention to compute the related information of the span. For the span (i, j), we first compute the attention score of each context words as follows:

$$\alpha_{(i,j)}^t = softmax(tanh(s_{(i,j)} \boldsymbol{W_\alpha} e_t^T + \boldsymbol{b_\alpha})) \tag{7}$$

where $\boldsymbol{W_\alpha}$ is the weight matrix and $\boldsymbol{b_\alpha}$ is the bias term. Then the sentiment information towards the span is computed as the weighted sum of the context:

$$c_{(i,j)} = \sum_{t=1}^{T} \alpha_{(i,j)}^t e_t \tag{8}$$

From the above, we find that the sentiment information is computed towards each span rather than each word, thus our span-based method can avoid sentiment inconsistency problem. However, the token tagging based methods can not completely ensure the sentiment consistency to all the opinion targets, since these methods calculate the sentiment information to individual word.

Output layer. We concatenate the span representation and the representation computed by the attention mechanism to predict the label of each span. For span (i, j), we compute the probability distribution as follows:

$$f_{(i,j)} = [s_{(i,j)}; c_{(i,j)}] \tag{9}$$

$$y = softmax(\boldsymbol{W_y} f_{(i,j)} + \boldsymbol{b_y}) \quad (10)$$

where $y \in \mathbb{R}^K$ is the probability distribution of the labels, K is the number of labels in C which is 4 here, $\boldsymbol{W_y}$ and $\boldsymbol{b_y}$ are the weight matrix and bias term, respectively.

2.3 Model Training

The goal of the training is to optimize all the parameters so as to minimize the loss function as much as possible. We use g_i which is a one-hot vector to represent the gold label of a span. Let y_i denote the predicted distribution of a span. We use the cross entropy between them as the loss function:

$$loss = -\sum_{h \in H} \sum_{i \in S_h} g_i log(y_i) \quad (11)$$

where H denotes all the training sentences, S_h denotes the set of span in sentence h.

3 Experiments

3.1 Datasets

Following the experiments of a recent complete TBSA task paper [Li *et al.*, 2019], we conduct experiments on three benchmark datasets as shown in Table 2. The first dataset D_L contains the reviews of the laptop domain from SemEval Challenge 2014 [Pontiki *et al.*, 2014]. We use the same train-test split as the original dataset. The second dataset D_R consists of the reviews from the restaurant domain. We merge the restaurant datasets from SemEval Challenge 2014, 2015 and 2016 [Pontiki *et al.*, 2014; Pontiki *et al.*, 2015; Pontiki *et al.*, 2016]. The new training and testing datasets are obtained by merging the three years' training datasets and testing datasets, respectively. The third dataset D_T is comprised by the tweets collected by Mitchell [Mitchell *et al.*, 2013]. Similar to the work of [Li *et al.*, 2019], we randomly sample 10% data from the training set as the validation set for D_L and D_R. Since there is no standard train-test split for dataset D_T, we present the ten-fold cross validation results, as done in previous works [Mitchell *et al.*, 2013; Zhang *et al.*, 2015; Li *et al.*, 2019].

3.2 Experimental Settings

We employ word2vec tool[1] on two different corpora to get the traditional word embeddings of dataset D_L and D_R. For D_L, we use the corpus from laptop domain in Amazon reviews [McAuley *et al.*, 2015], which contains 1M reviews. For D_R, we train word embeddings on the Yelp Challenge dataset[2] which consists of 2.2M reviews. For D_T, we directly use the glove.840B.300d embeddings [Pennington *et al.*, 2014]. The word embeddings are fine tuned during training. The dimensions of the word embeddings for D_L, D_R and D_T are 200, 200 and 300, respectively. We adopt the pre-trained ELMo encoder from AllenNLP toolkit[3] to generate ELMo. The dimension of ELMo is 256.

[1]https://radimrehurek.com/gensim/models/word2vec.html

[2]http://www.yelp.com/dataset challenge

[3]http://allennlp.org/

Dataset		Train	Test	Total
D_L	POS	987	339	1326
	NEG	860	130	990
	NEU	450	165	615
D_R	POS	2607	1524	4131
	NEG	1035	500	1535
	NEU	664	263	927
D_T	POS	-		692
	NEG	-		263
	NEU	-		2244

Table 2: Statistics of the datasets.

We adopt 2-layers BiLSTM in our model and the number of the hidden units for each BiLSTM layer is 128. We apply dropout over the input embeddings and the dropout rate is set to be 0.5. We update the parameters of our model by backpropagation using RMSprop with learning rate 0.003 and batch size 32. According to statistics, more than 97% opinion targets in the datasets are comprised by no more than 4 tokens. Thus the maximum length L is set to 4 in our experiments.

For evaluation, we use the Precision (P), Recall(R) and F1 score as metrics in our experiments. An output span is considered to be correct only if it exactly match with the gold annotated target. The exact match means both the words in the span and the sentiment towards the span are the same as the gold annotated target.

3.3 Results and Analysis

Baselines

We compare our span-based joint model with both the pipelined methods and the token tagging based joint methods. The pipelined methods extract opinion targets first, and then predict the sentiment expressed on them. The token tagging joint methods process the two tasks at the same time. These methods are listed as follows:

- **CRF-{pipeline, joint}** [Mitchell *et al.*, 2013]: A sequence tagger based on Conditional Random Fields (CRF). "pipeline" represents applying the model in a pipeline way. And "joint" denotes the model based on the unified tagging scheme.
- **NN-CRF-{pipeline, joint}** [Zhang *et al.*, 2015]: Extend a CRF based model with word embeddings and automatic feature extractors.
- **HAST-TNet**: The HAST model [Li *et al.*, 2018b] is adopted to extract the opinion targets. And then the TNet model [Li *et al.*, 2018a] is used to determine the sentiment towards these targets.
- **LSTM**: The standard LSTM model based on the unified tagging scheme.
- **LSTM-CRF-1** [Lample *et al.*, 2016]: LSTM-CRF model enhanced with word-level and character-level embeddings.
- **LSTM-CRF-2** [Ma and Hovy, 2016]: The model is similar to LSTM-CRF-1 model except that the character-level embeddings are learned by CNN instead of LSTM.

Model		D_L			D_R			D_T		
		P	R	F1	P	R	F1	P	R	F1
pipeline	CRF-pipeline	59.69	47.54	52.93	52.28	51.01	51.64	42.97	25.21	31.73
	NN-CRF-pipeline	57.72	49.32	53.19	60.09	61.39	61.00	43.71	37.12	40.06
	HAST-TNet	56.42	54.20	55.29	62.18	73.49	67.36	46.30	49.13	47.66
tagging-based joint	CRF-joint	59.72	41.86	49.06	63.39	57.74	60.03	48.35	19.64	27.86
	NN-CRF-joint	58.72	45.96	51.56	62.61	60.53	61.56	46.32	32.84	38.36
	LSTM	57.91	46.21	51.40	62.80	63.49	63.14	51.45	37.62	43.41
	LSTM-CRF-1	58.61	50.47	54.24	66.10	66.30	66.20	51.67	44.08	47.52
	LSTM-CRF-2	58.66	51.26	54.71	61.56	67.26	64.29	53.74	42.21	47.26
	LM-LSTM-CRF	53.31	59.4	56.19	68.46	64.43	66.38	43.52	52.01	47.35
	BG-SC-OE	61.27	54.89	57.90	68.64	71.01	69.80	53.08	43.56	48.01
span-based joint	our model	61.40	58.20	**59.76**	76.20	68.20	**71.98**	54.84	48.44	**51.44**

Table 3: Comparison results with baselines.

Model	D_L			D_R			D_T		
	P	R	F1	P	R	F1	P	R	F1
our model	61.40	58.20	**59.76**	76.20	68.20	**71.98**	54.84	48.44	**51.44**
-ELMo	60.46	57.40	58.89	77.31	65.41	70.87	53.69	47.33	50.31
-BR(boundary representation)	58.25	55.68	56.94	76.33	63.31	69.21	51.75	46.96	49.20
-MR(merged representation)	59.18	54.89	56.95	71.93	68.91	70.39	53.65	47.05	50.13
-SA(span-aware attention)	59.46	55.82	57.58	71.09	69.78	70.43	53.52	47.61	50.34

Table 4: Ablation tests for complete TBSA task.

Model	target	sentiment
	F1	Acc
our model	**78.54**	**76.08**
-ELMo	77.53	75.96
-BR(boundary representation)	77.26	73.69
-MR(merged representation)	77.74	73.26
-SA(span-aware attention)	78.43	73.42

Table 5: Ablation tests for opinion target extraction and target sentiment classification on D_L.

- **LM-LSTM-CRF** [Liu *et al.*, 2018]: A competitive LSTM-CRF model in several sequence tagging tasks enhanced with language model.
- **BG-SC-OE** [Li *et al.*, 2019]: A model which designs three components to promote the complete TBSA task: boundary guidance (BG) component, sentiment consistency (SC) component and opinion-enhanced (OE) component.

The models of LSTM, LSTM-CRF-1, LSTM-CRF-2, LM-LSTM-CRF and BG-SC-OE are all based on the unified tagging scheme mentioned above.

Main Results

Table 3 shows the results of our model compared with the baselines. Our span-based method achieves significant improvements over all the baselines in F1 score. In particular, compared with the current state-of-the-art method BG-SC-OE which is a carefully-designed model using external sentiment lexicon, our model still achieves 1.86%, 2.18% and 3.43% improvements on D_L, D_R and D_T, respectively. The improvements presumably benefit from two aspects. First of all, our model can utilize both the boundary representation and the merged representation to capture the global information of a target, while the BG-SC-OE model can not make use of this information. Secondly, our model calculates the sentiment information to each span rather than each word, thus it can completely guarantee the sentiment consistency. For the BG-SC-OE model, the sentiment consistency relies on the matrix weights learned from the training process which can not ensure the consistency for all cases.

We also notice that both our model and BG-SC-OE achieve better performance than HAST-TNet which is the pipeline of two state-of-the-art models. This indicates that a joint model is more effective than a pipelined solution for the complete TBSA task. This observation is consistent with other strong couplings tasks such as NER and relation extraction.

Ablation Study

To investigate the effect of each component in our model, we conduct a set of ablation experiments as shown in Table 4. The F1 score drops when ELMo is removed which indicates the contextualized embedding is useful for the prediction. We also see that even without ELMo, our performance is still better than the current state-of-the-art method, which demonstrates the effectiveness of our span-based model. The performance drops considerably by removing one of the span-level features (ie. BR or MR). This proves the global information of the words in an opinion target, which can not be used in the token tagging based method, is helpful for the complete TBSA task. From the last line, we find that the attention mechanism, which is used to select the sentiment information in the context, can facilitate the prediction.

To better understand how the components affect the complete TBSA task, we give a detailed analysis for the two subtasks (i.e. opinion target extraction and target sentiment classification) on D_L. For the evaluation of opinion target extrac-

	Input	Prediction of BG-SC-OE	Prediction of our model
S1	$[Set\ up]_{POS}$ was easy.	Set up was easy.	$[Set\ up]_{POS}$ was easy.
S2	They also have a great $[assortment\ of\ wraps]_{POS}$ if your not in the mood for $[traditional\ Mediterranean\ fare]_{NEG}$.	They also have a great assortment of wraps if your not in the mood for $[traditional\ Mediterranean\ fare]_{NEG}$.	They also have a great $[assortment\ of\ wraps]_{POS}$ if your not in the mood for $[traditional\ Mediterranean\ fare]_{NEG}$.
S3	I had $[roast\ \ chicken]_{NEU}$ and a $[salad]_{NEU}$.	I had $[roast]_{NEU}$ $[chicken]_{POS}$ and a $[salad]_{NEU}$.	I had $[roast\ \ chicken]_{NEU}$ and a $[salad]_{NEU}$.
S4	so i called $[technical\ support]_{NEU}$.	so i called $[technical]_{NEG}$ $[support]_{NEU}$.	so i called $[technical\ support]_{NEU}$.

Table 6: Output from different models. The first column is the gold standard. The second and the third columns are the results of BG-SC-OE model and our model, respectively.

tion, we regard a predicted span as correct if it matches the gold annotation regardless of the sentiment. For sentiment classification, if a span is correctly predicted as an opinion target, we compute the sentiment accuracy of it. We report the F1 score for opinion target extraction and the accuracy for target sentiment classification in Table 5. We observe that ELMo is able to promote the complete TBSA task mainly due to it can improve the performance of opinion target extraction. When we remove one of the span-level features, the performance of the two tasks will drop, which indicates these features are useful for both the two tasks. This is because, on the one hand, the span-level features can capture the global information of the words in the targets which is important to the opinion target extraction. On the other hand, the sentiment information in the context is computed based on the span representation, thus it can affect the sentiment classification. Furthermore, compared with MR, BR has more effect for opinion target extraction and less effect for target sentiment classification. From the last line, we find that the attention mechanism can provide useful context information to both the two subtasks.

Case Analysis

We pick some examples from the test dataset and present the prediction results of our model and BG-SC-OE model in Table 6. As illustrated in S1 and S2, our model can better identify opinion target than BG-SC-OE model. Take "Set up" in S1 as an example, it is failed to identify the span as an opinion target when separately predicting the tags of "Set" and "up". But treating the two words as a whole and using the span-level features of them, our model can correctly extract the target from the sentence. From the example of S3 and S4, we observe that though the tagging based joint methods propose some components to maintain the sentiment consistency, there still exists sentiment inconsistency in their outputs (e.g. the sentiment of "roast" and "chicken" are different in S3). This is because these components always depend on the matrix weights learned from the training data, which can hardly ensure the consistency for all the targets. And our method can completely guarantee the sentiment consistency, since we treat all the words in an opinion target as a whole and incorporate the sentiment information to the target.

4 Related Work

Target-based sentiment analysis can be divided into two subtasks: opinion target extraction and target sentiment classification. Most of the existing studies focused on one of the subtasks. The goal of the first subtask is to detect the opinion targets in a sentence, which have been studied by many researchers [Wang *et al.*, 2017; Li *et al.*, 2018b; Xu *et al.*, 2018; Wang and Pan, 2018]. There are also a lot of works on the target sentiment classification [Chen *et al.*, 2017; Fan *et al.*, 2018], which assumes the opinion targets are given in advance and aims to predict the sentiment towards the targets. However, in more practical applications, we need to perform both of the two subtasks. Recently, some researches attempted to conduct them jointly. [Mitchell *et al.*, 2013] and [Zhang *et al.*, 2015] employed CRF with hand-crafted features and automatic extracted features to the complete TBSA task, respectively. [Ma *et al.*, 2018] proposed a HMBi-GRU based joint model to this task. [Li *et al.*, 2019] carefully designed an integrated model which has achieved better results than all the previous methods. All the above works converted the complete TBSA task into a sequence tagging problem and aimed to predict the tag of each token. The opinion targets and the sentiment expressed on them can be reconstructed from the predicted tags.

Recently, the span-based methods have achieved highly competitive performance in NLP tasks. [Xu *et al.*, 2017] proposed a local detection methods based on FOFE for NER. Lee et al. presented an end-to-end model for coreference resolution, which considered all spans in a document as potential mentions [Lee *et al.*, 2017; Lee *et al.*, 2018]. The span-based models have also been utilized for SRL, which predicted the role of the spans and inferenced the realtions between them [Ouchi *et al.*, 2018; He *et al.*, 2018].

5 Conclusion

In this paper, we propose a span-based joint model for the complete TBSA task. Different from current token tagging based joint methods, our model can take advantage of the global information of a target. Furthermore, we present a span-aware attention mechanism to compute the sentiment information of the span. Calculating this information to each span rather than each word, our model avoid the sentiment inconsistency problem. We conduct experiments on three public datasets and the results show the effectiveness of our model.

Acknowledgements

This research is supported in part by the Beijing Municipal Science and Technology Project under Grant Z181100002718004, the National Key Research and Development Program of China under Grant 2017YFB1010000 and 2018YFC0806900, and the National Natural Science Foundation of China under Grant 61702500.

References

[Chen *et al.*, 2017] Peng Chen, Zhongqian Sun, Lidong Bing, and Wei Yang. Recurrent attention network on memory for aspect sentiment analysis. In *EMNLP*, pages 452–461, 2017.

[Fan *et al.*, 2018] Feifan Fan, Yansong Feng, and Dongyan Zhao. Multi-grained attention network for aspect-level sentiment classification. In *EMNLP*, pages 3433–3442, 2018.

[Finkel and Manning, 2009] Jenny Rose Finkel and Christopher D Manning. Joint parsing and named entity recognition. In *ACL*, pages 326–334, 2009.

[He *et al.*, 2018] Luheng He, Kenton Lee, Omer Levy, and Luke Zettlemoyer. Jointly predicting predicates and arguments in neural semantic role labeling. In *ACL*, pages 364–369, 2018.

[Lample *et al.*, 2016] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. Neural architectures for named entity recognition. In *NAACL*, pages 260–270, 2016.

[Lee *et al.*, 2017] Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. End-to-end neural coreference resolution. In *EMNLP*, pages 188–197, 2017.

[Lee *et al.*, 2018] Kenton Lee, Luheng He, and Luke Zettlemoyer. Higher-order coreference resolution with coarse-to-fine inference. In *NAACL*, pages 687–692, 2018.

[Li and Ji, 2014] Qi Li and Heng Ji. Incremental joint extraction of entity mentions and relations. In *ACL*, pages 402–412, 2014.

[Li *et al.*, 2018a] Xin Li, Lidong Bing, Wai Lam, and Bei Shi. Transformation networks for target-oriented sentiment classification. In *ACL*, pages 946–956, 2018.

[Li *et al.*, 2018b] Xin Li, Lidong Bing, Piji Li, Wai Lam, and Zhimou Yang. Aspect term extraction with history attention and selective transformation. In *IJCAI*, pages 4194–4200, 2018.

[Li *et al.*, 2019] Xin Li, Lidong Bing, Piji Li, and Wai Lam. A unified model for opinion target extraction and target sentiment prediction. In *AAAI*, 2019.

[Liu *et al.*, 2018] Liyuan Liu, Jingbo Shang, Xiang Ren, Frank Fangzheng Xu, Huan Gui, Jian Peng, and Jiawei Han. Empower sequence labeling with task-aware neural language model. In *AAAI*, 2018.

[Liu, 2012] Bing Liu. Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1):1–167, 2012.

[Ma and Hovy, 2016] Xuezhe Ma and Eduard Hovy. End-to-end sequence labeling via bi-directional lstm-cnns-crf. In *ACL*, pages 1064–1074, 2016.

[Ma *et al.*, 2018] Dehong Ma, Sujian Li, and Houfeng Wang. Joint learning for targeted sentiment analysis. In *EMNLP*, pages 4737–4742, 2018.

[McAuley *et al.*, 2015] Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton Van Den Hengel. Image-based recommendations on styles and substitutes. In *SIGIR*, pages 43–52, 2015.

[Mitchell *et al.*, 2013] Margaret Mitchell, Jacqui Aguilar, Theresa Wilson, and Benjamin Van Durme. Open domain targeted sentiment. In *EMNLP*, pages 1643–1654, 2013.

[Ouchi *et al.*, 2018] Hiroki Ouchi, Hiroyuki Shindo, and Yuji Matsumoto. A span selection model for semantic role labeling. In *EMNLP*, pages 1630–1642, 2018.

[Pennington *et al.*, 2014] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *EMNLP*, pages 1532–1543, 2014.

[Peters *et al.*, 2018] Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In *NAACL*, pages 2227–2237, 2018.

[Pontiki *et al.*, 2014] Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. Semeval-2014 task 4: Aspect based sentiment analysis. In *SemEval 2014*, pages 27–35, 2014.

[Pontiki *et al.*, 2015] Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. Semeval-2015 task 12: Aspect based sentiment analysis. In *SemEval 2015*, pages 486–495, 2015.

[Pontiki *et al.*, 2016] Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, AL-Smadi Mohammad, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, et al. Semeval-2016 task 5: Aspect based sentiment analysis. In *SemEval 2016*, pages 19–30, 2016.

[Thelwall *et al.*, 2010] Mike Thelwall, Kevan Buckley, Georgios Paltoglou, Di Cai, and Arvid Kappas. Sentiment strength detection in short informal text. *Journal of the American Society for Information Science and Technology*, 61(12):2544–2558, 2010.

[Wang and Pan, 2018] Wenya Wang and Sinno Jialin Pan. Transition-based adversarial network for cross-lingual aspect extraction. In *IJCAI*, pages 4475–4481, 2018.

[Wang *et al.*, 2017] Wenya Wang, Sinno Jialin Pan, Daniel Dahlmeier, and Xiaokui Xiao. Coupled multi-layer attentions for co-extraction of aspect and opinion terms. In *AAAI*, pages 3316–3322, 2017.

[Xu *et al.*, 2017] Mingbin Xu, Hui Jiang, and Sedtawut Watcharawittayakul. A local detection approach for named entity recognition and mention detection. In *ACL*, pages 1237–1247, 2017.

[Xu *et al.*, 2018] Hu Xu, Bing Liu, Lei Shu, and Philip S Yu. Double embeddings and cnn-based sequence labeling for aspect extraction. In *ACL*, pages 592–598, 2018.

[Zhang *et al.*, 2015] Meishan Zhang, Yue Zhang, and Duy Tin Vo. Neural networks for open domain targeted sentiment. In *EMNLP*, pages 612–621, 2015.