Revising Attention with Position for Aspect-Level Sentiment Classification

Dong Wang[1,2], Tingwen Liu[1,2*], and Bin Wang[3]

[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]Xiaomi AI Lab, Beijing, China

{wangdong, liutingwen}@iie.ac.cn and wangbin11@xiaomi.com

Introduction

Problems

- As a fine-grained classification task, aspect-level sentiment classification aims at determining the sentiment polarity given a particular target in a sentence.
- The key point of this task is to distinguish target-related words and target-unrelated words. To this end, attention mechanism is introduced into this task, which assigns high attention weights to target-related words and ignores target-unrelated words.
- However, existing work not explicitly take into account the position information of context words when calculating the attention weights.

Our goal

Combine position information and attention mechanism explicitly. We get the position distribution according to the distances between context words and target, then leverage the position distribution to modify the attention weight distribution to get a more reasonable attention weight distribution.

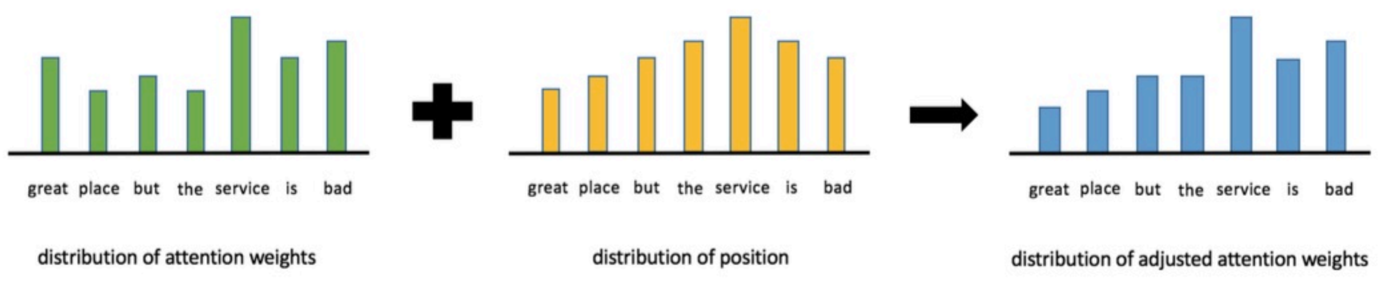

Fig. 1. The illustration of how position distribution adjusts attention weight distribution. The green, yellow and blue represent original attention weight distribution, position distribution, and adjusted attention weight distribution respectively. The target is "*service*". (Color figure online)

Our contributes

- We propose a novel approach to explicitly use position information: leverage the position distribution to adjust the attention weight distribution.
- We explore a variety of ways to utilize position information and introduce CNN to replace LSTM for capturing local n-gram feature more effectively.
- Our approach achieves comparable performance on two public benchmark datasets Restaurants and Laptops from SemEval 2014.

Our Approach

On one hand, we also use the standard attention mechanism to calculate the weight of each word and get the attention weight distribution. On the other hand, we use the distances between context words and target to get the position distribution. Then, we calculate the difference between attention weight distribution and position distribution, and add it to the classification loss as a penalty.

During the training process, model will adjust the original attention weight distribution to reduce the difference between it and the position distribution, and finally reaches a balance between classification error and distribution difference. The Fig.1 shows the above process. Actually, we explore multiple ways to calculate the position distribution and the difference between position distribution and attention weight distribution.

Considering that sentiment polarity is usually represented by a phrase, we used CNN to replace LSTM to capture local n-gram feature. Similar to textCNN, we also apply the CNN on word embeddings. The difference is that we will first adjust the word embeddings with the adjusted attention weights to eliminate the information of target-unrelated words, then use CNN to get the final sentiment polarity.

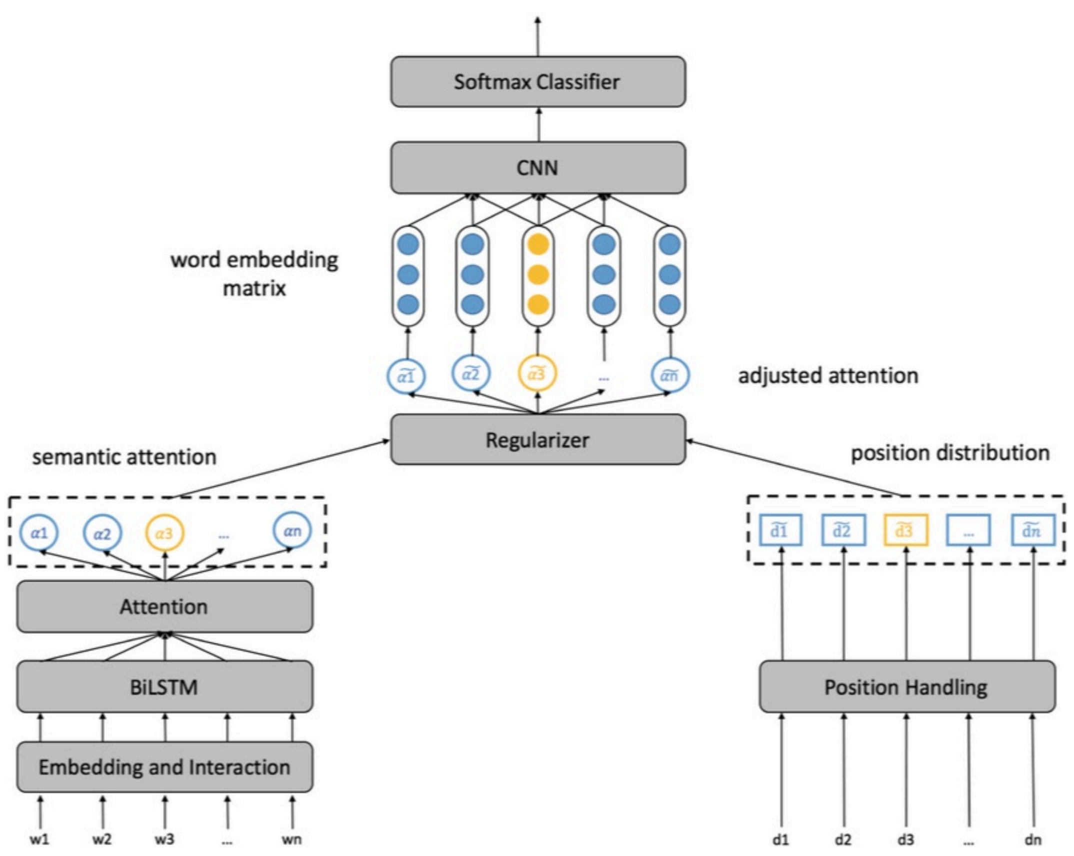

Fig. 2. An overview of our model. The lower left part is the module for calculating of attention, and the lower right part is the module for calculating of distance. The upper part is the classification module.

Experiment

Datasets

We perform experiments on SemEval 2014 which includes two datasets: Laptop and Restaurant. Each sample in the datasets consists of a sentence, a target and the corresponding sentiment polarity. Table1 shows the statistic results of the datasets.

Table 1. Statistic results of *Laptop* and *Restaurant*.

	Set	Total	Positive	Negative	Neutral
Laptop	Train	2328	994	870	464
	Test	638	341	128	169
Restaurant	Train	3608	2164	807	637
	Test	1120	728	196	196

Experimental Results and Analysis

Table 2 shows the experimental results of our proposed model and other baseline models. We call our proposed basic model as LAC-Pos, which means LSTM-Attention-CNN-Position. LAC-Pos-N-AD means basic LAC-Pos equipped with normalized position distribution and using absolute difference to calculate the difference between position distribution and attention weight distribution.

From Table 2 we can observe that our approach outperforms all the compared approaches on both Laptop and Restaurant. And the performances among all the LAC-Pos variants are very close which demonstrates that our approach is robust, generalized and not limited to specific methods of calculating position distribution and calculating the difference.

Case Study

In this part, we take a case to show the effectiveness of our method. Figure3 shows two heat maps of attention weights from LAC and LAC-Pos-N-AD respectively. The attention weights are both transformed into between 0 and 1 which indicates how much information should be preserved for every word. The weight of each word is visualized by the color depth where the redder the color, the greater the attention weight.

Table 2. Experimental results of our proposed model and compared baseline models. Models with * indicate that position information is taken into account.

	Models	Laptop		Restaurant	
		ACC	Macro-F1	ACC	Macro-F1
Baselines	SVM	70.49	–	80.16	–
	AE-LSTM	68.90	–	76.60	–
	ATAE-LSTM	68.70	–	77.20	–
	TD-LSTM	71.83	68.43	78.00	66.73
	IAN	72.10	–	78.60	–
	EAM*	71.94	69.23	80.63	71.32
	MemNet*	72.21	–	80.95	–
	PBAN*	74.12	–	81.16	–
LAC-Pos variants	LAC-Pos-N-AD	74.92	70.67	81.25	71.64
	LAC-Pos-N-KL	**75.08**	70.73	80.98	71.71
	LAC-Pos-N-JS	74.76	70.21	81.25	71.73
	LAC-Pos-G-AD	74.76	70.50	81.07	70.76
	LAC-Pos-G-KL	74.61	70.30	**81.34**	**71.91**
	LAC-Pos-G-JS	74.76	**70.76**	80.89	70.65

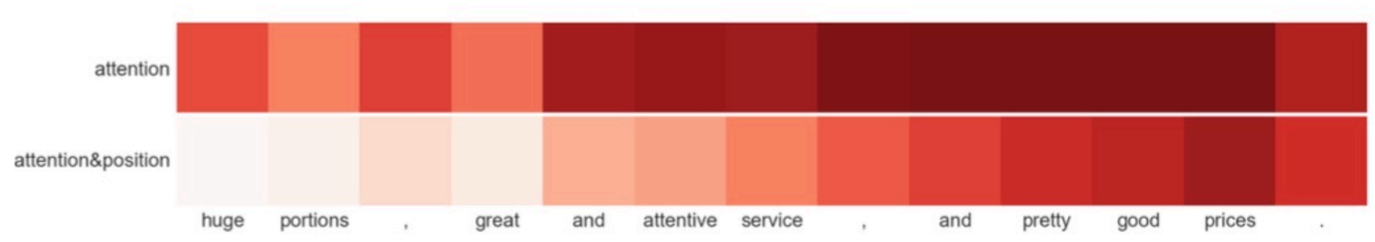

Fig. 3. Illustration of attention where horizontal axis is the sentence and the vertical axis is the method. The upper heat map is the attention distribution of LAC and the lower heat map is the attention distribution of LAC-Pos-N-AD, where the targets are *prices*.

Conclusion

In this paper, we analyze the importance of position information in aspect-level sentiment classification and propose a novel approach to combine position information and attention mechanism: leverage the position distribution to modify the attention weight distribution. Then we use the adjusted attention weights to adjust the word embeddings and apply CNN on the weighted embedding matrix to capture the local n-gram features for sentiment classification. We perform experiments on two datasets of SemEval 2014 and the experimental results show the effectiveness of our model.