Combining Fine-Tuning with a Feature-Based Approach for Aspect Extraction on Reviews (Student Abstract)

Xili Wang,[1,2,3] Hua Xu,[1,2] Xiaomin Sun,[1,2] Guangcan Tao[4]
[1]State Key Laboratory of Intelligent Technology and Systems, Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China
[2]Beijing National Research Center for Information Science and Technology(BNRist), Beijing 100084, China
[3]Department of Automation, Tsinghua University, Beijing 100084, China
[4]Food Safety and Nutrition (Guizhou) Information Technology Co., Ltd
zdlp@sina.cn, {xuhua, sxm123}@tsinghua.edu.cn, tgcan@fsnip.com

Abstract

One key task of fine-grained sentiment analysis on reviews is to extract aspects or features that users have expressed opinions on. Generally, fine-tuning BERT with sophisticated task-specific layers can achieve better performance than only extend one extra task-specific layer (e.g., a fully-connected + softmax layer) since not all tasks can easily be represented by Transformer encoder architecture and special task-specific layer can capture task-specific features. However, BERT fine-tuning may be unstable on a small-scale dataset. Besides, in our experiments, directly fine-tuning BERT on extending sophisticated task-specific layers did not take advantage of the features of task-specific layers and even restrict the performance of BERT module. To address the above consideration, this paper combines Fine-tuning with a feature-based approach to extract aspect. To the best of our knowledge, this is the first paper to combine fine-tuning with a feature-based approach for aspect extraction.

1 Introduction

Aspect extraction is an important task in aspect-based sentiment analysis (Hu and Liu 2004) which, in recent works, usually be divided into two stages: Aspect extraction and sentiment classification. It aims to extract opinion aspect (or target) from opinion text. In reviews, aspects are attributes or features of opinion targets rather than opinion targets. For example, from "Set up was easy", in a laptop review, it aims to extract "Set up", but from "This mac has been a problem since we got it" in a laptop review, the "mac" should not be extracted as an aspect since "mac" is an opinion target in laptop review. Inspired by the recent success of deep learning approach, most of the recent work in aspect extraction is modeling in the supervised deep learning approach (Xu et al. 2018; Shu, Xu, and Liu 2019; Xu et al. 2019).

Although these approaches can achieve better performances than their prior works, there are some other considerations that are also important. Recently, Significant progress has been made by using contextualized representation learning (e.g., BERT) in the literature of NLP. Recent works (Hewitt and Manning 2019) found that context embeddings (e.g., BERT) contain word sense(e.g., is "bark" an animal noise or part of a tree?) and represent dependency parse trees geometrically which is especially important for aspect extraction. However, most of the existing methods for aspect extraction on reviews did not utilize contextualized embedding. Special task-specific layer can capture task-specific features which can improve model's performance, though BERT-PT uses BERT as the base model, they only use one linear layer as extend task-specific layer. CNN layers proved to be a good structure for aspect extraction. However, since the architecture of CNN is greatly different from transformer architecture, If we directly train them synchronously, the model tends to converge to local minima. Actually in our experiment, directly fine-tuning BERT on task-specific model structure (CNN layers) didn't take advantage of the features of task-specific layers and even restrict the performance of BERT module. To address this consideration, compare to traditional BERT training procedures, we introduce a new BERT training procedure which provides significant improvements in aspect extraction.

2 Proposed Methodology

Given a review sequence W which has n positions (tokens). The goal of our model is to predict the label for each position of inputs. We modeled aspect extraction as a sequence labeling problem. The labeling space is $Y = \{B, I, O\}$. Since an aspect can be a phrase, we make the label B, I indicate the beginning word and non-beginning word of an aspect phrase respectively. Mean-while, O indicates non-aspect words.

2.1 Embedding Layer

BERT is less task-awareness and domain-awareness (Xu et al. 2019), and BERT fine-tuning may unstable on a small-scale dataset. Based on the above consideration, we did not simply use the pre-trained BERT to generate word vectors of sequence. To address domain challenge and task-awareness challenge, after pre-train BERT on large-scale corpora, we simply apply the post-training procedure proposed by (Xu et al. 2019). Specifically, to post-train on domain knowledge and task-aware knowledge, we leverage the two pre-training objectives from BERT: masked language model (MLM) and

next sentence prediction (NSP), and train BERT on domain-specific and task-specific datasets respectively.

In our experiments, after post-training, we found that if we extend BERT with CNN layers, such as DE-CNN(a great model for aspect extraction), and fine-tune BERT on aspect extraction task, the performance of this model will become unstable. The model didn't take advantage of the features of task-specific layers and even restrict the performance of BERT module. Besides, hyper-parameters are hard to tune since CNN layers need different learning rates compare with one linear extra task-specific layer. If we simultaneously train BERT layers and CNN layers, it would be greatly time-consuming too. To handle this problem, after post-training BERT procedure, we first fine-tune BERT on one linear extra task-specific layer to get contextual embedding, which will contain word sense and syntax information since we train the model on aspect-specific domain datasets. Then we use the BERT last layer (remove the pool layer of BERT) as our model's embedding layer. We do freeze the BERT embedding layer parameters because our test dataset may contain many unseen words. If we set the embedding parameters to be trained, the follow-up CNN layers will be adjusted according to the training dataset. But the embeddings of unseen words from test data still have the old features that may be mistakenly extracted by CNN.

2.2 multiple CNN layers

Note that we keep the architecture of (Xu et al. 2018) as far as possible for the sake of comparing with DE-CNN. Specifically, Assume the input is a sequence of word indices: $X = (x_1, x_2, ..., x_n)$, then the BERT embeddings (embedding matrices) is E. since we only have one embedding, we remove the embedding concatenating layer in DE-CNN and directly feed our embedding into a stack of 4 CNN layers. Each CNN layer performs convolution operation and ReLU activation. For the first CNN layer, we employ two different filter sizes: 3 and 5. For the rest 3 CNN layers, we only use one filter size: 5. We apply dropout after the embedding layer and each ReLU activation. We also apply a fully-connected layer with weights shared across all positions and a softmax layer to compute label distribution for each word as in DE-CNN. The output size of the fully-connected layer is $Y = 3$.

3 Experiment

We evaluate our model on one 2080 Ti GPU on two benchmark datasets from SemEval challenges: SemEval-14 Laptop,SemEval-16 Restaurant. These two datasets consist of review sentences with aspect terms labeled as spans of characters. Each review (one sentence) contains one or more aspects. To be consistent with existing research, we use the standard evaluation scripts come with the SemEval datasets and evaluate our model by F1 score. We compare our model with several state-of-the-art methods including DE-CNN (Xu et al. 2018), Ctrl (Shu, Xu, and Liu 2019), BERT (Devlin et al. 2018), BERT-PT (Xu et al. 2019). To demonstrate the effectiveness of our model, we also compared CFF-CNN (proposed) with BERT-PT-CNN (ours) which directly fine-tune BERT with CNN layers as extend task-specific layers.

Table 1: Comparison results in F1 score: numbers in the second group are averaged scores of 50 runs. * indicates the result is statistically significant at the level of 0.05.

Model	Laptop	Restaurant
DE-CNN	81.59	74.37
Ctrl	82.73	75.64
BERT	79.28	74.1
BERT-PT	84.26	77.97
CFF-CNN (ours)	**85.41***	**79.24***
BERT-PT-CNN (ours)	84.63	76.40

4 Results and Analysis

From Table 1, we can see that our model CFF-CNN performs the best. CNNs summarize a fixed size context through multiple layers. However, BERT-PT-CNN didn't take advantage of this feature. In our experiment, BERT-PT-CNN didn't correctly judge some long aspect term. For example, from "No installation disk (DVD) is included" in a laptop review, ground truth aspect term is "installation disk (DVD)". BERT-PT-CNN tends to extract "installation disk(" as BERT-PT do. Compare to BERT-PT-CNN, CFF-CNN does correctly extract "installation disk (DVD)". The results show that CFF-CNN can take advantage of the features of CNN layers and improve the performance of BERT module.

5 Acknowledgments

This paper is founded by National Natural Science Foundation of China (Grant No: 61673235) and National Key R&D Program Projects of China (Grant No: 2017YFC1601804).

References

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Hewitt, J., and Manning, C. D. 2019. A structural probe for finding syntax in word representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 4129–4138.

Hu, M., and Liu, B. 2004. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, 168–177. ACM.

Shu, L.; Xu, H.; and Liu, B. 2019. Controlled cnn-based sequence labeling for aspect extraction. *arXiv preprint arXiv:1905.06407*.

Xu, H.; Liu, B.; Shu, L.; and Yu, P. S. 2018. Double embeddings and cnn-based sequence labeling for aspect extraction. *arXiv preprint arXiv:1805.04601*.

Xu, H.; Liu, B.; Shu, L.; and Yu, P. S. 2019. Bert post-training for review reading comprehension and aspect-based sentiment analysis. *arXiv preprint arXiv:1904.02232*.