

Utilizing BERT Intermediate Layers for Aspect Based Sentiment Analysis and Natural Language Inference

Youwei Song, Jiahai Wang ^{*}, Zhiwei Liang, Zhiyue Liu, Tao Jiang

School of Data and Computer Science

Sun Yat-sen University

Guangzhou, China

{songyw5, liangzhw25, liuzhy93, jiangt59}@mail2.sysu.edu.cn
wangjiah@mail.sysu.edu.cn

Abstract

Aspect based sentiment analysis aims to identify the sentimental tendency towards a given aspect in text. Fine-tuning of pretrained BERT performs excellent on this task and achieves state-of-the-art performances. Existing BERT-based works only utilize the last output layer of BERT and ignore the semantic knowledge in the intermediate layers. This paper explores the potential of utilizing BERT intermediate layers to enhance the performance of fine-tuning of BERT. To the best of our knowledge, no existing work has been done on this research. To show the generality, we also apply this approach to a natural language inference task. Experimental results demonstrate the effectiveness and generality of the proposed approach.

1 Introduction

Aspect based sentiment analysis (ABSA) is an important task in natural language processing. It aims at collecting and analyzing the opinions toward the targeted aspect in an entire text. In the past decade, ABSA has received great attention due to a wide range of applications (Pang et al., 2008; Liu, 2012). Aspect-level (also mentioned as target-level) sentiment classification as a subtask of ABSA (Pang et al., 2008) aims at judging the sentiment polarity for a given aspect. For example, given a sentence “*I hated their service, but their food was great*”, the sentiment polarities for the target “*service*” and “*food*” are negative and positive respectively.

Most of existing methods focus on designing sophisticated deep learning models to mining the relation between context and the targeted aspect. Majumder et al., (2018) adopt a memory network architecture to incorporate the related information of neighboring aspects. Fan et al., (2018) combine

the fine-grained and coarse-grained attention to make LSTM treasure the aspect-level interactions. However, the biggest challenge in ABSA task is the shortage of training data, and these complex models did not lead to significant improvements in outcomes.

Pre-trained language models can leverage large amounts of unlabeled data to learn the universal language representations, which provide an effective solution for the above problem. Some of the most prominent examples are ELMo (Peters et al., 2018), GPT (Radford et al., 2018) and BERT (Devlin et al., 2018). BERT is based on a multi-layer bidirectional Transformer, and is trained on plain text for masked word prediction and next sentence prediction tasks. The pre-trained BERT model can then be fine-tuned on downstream task with task-specific training data. Sun et al., (2019) utilize BERT for ABSA task by constructing auxiliary sentences, Xu et al., (2019) propose a post-training approach for ABSA task, and Liu et al., (2019) combine multi-task learning and pre-trained BERT to improve the performance of various NLP tasks. However, these BERT-based studies follow the canonical way of fine-tuning: append just an additional output layer after BERT structure. This fine-tuning approach ignores the rich semantic knowledge contained in the intermediate layers. Due to the multi-layer structure of BERT, different layers capture different levels of representations for the specific task after fine-tuning.

This paper explores the potential of utilizing BERT intermediate layers for facilitating BERT fine-tuning. On the basis of pre-trained BERT, we add an additional pooling module, design some pooling strategies for integrating the multi-layer representations of the classification token. Then, we fine tune the pre-trained BERT model with this additional pooling module and achieve new state-

^{*}The corresponding author.

of-the-art results on ABSA task. Additional experiments on a large Natural Language Inference (NLI) task illustrate that our method can be easily applied to more NLP tasks with only a minor adjustment.

Main contributions of this paper can be summarized as follows:

1. It is the first to explore the potential of utilizing intermediate layers of BERT and we design two effective information pooling strategies to solve aspect based sentiment analysis task.
2. Experimental results on ABSA datasets show that our method is better than the vanilla BERT model and can boost other BERT-based models with a minor adjustment.
3. Additional experiments on a large NLI dataset illustrate that our method has a certain degree of versatility, and can be easily applied to some other NLP tasks.

2 Methodology

2.1 Task description

ABSA Given a sentence-aspect pair, ABSA aims at predicting the sentiment polarity (*positive*, *negative* or *neutral*) of the sentence over the aspect.

NLI Given a pair of sentences, the goal is to predict whether a sentence is an *entailment*, *contradiction*, or *neutral* with respect to the other sentence.

2.2 Utilizing Intermediate Layers: Pooling Module

Given the hidden states of the first token (i.e., [CLS] token) $\mathbf{h}_{\text{CLS}} = \{h_{\text{CLS}}^1, h_{\text{CLS}}^2, \dots, h_{\text{CLS}}^L\}$ from all L intermediate layers. The canonical way of fine-tuning simply take the final one (i.e., h_{CLS}^L) for classification, which may inevitably lead to information losing during fine-tuning. We design two pooling strategies for utilizing \mathbf{h}_{CLS} : LSTM-Pooling and Attention-Pooling. Accordingly, the models are named **BERT-LSTM** and **BERT-Attention**. The overview of BERT-LSTM is shown in Figure 1. Similarly, BERT-Attention replaces the LSTM module with an attention module.

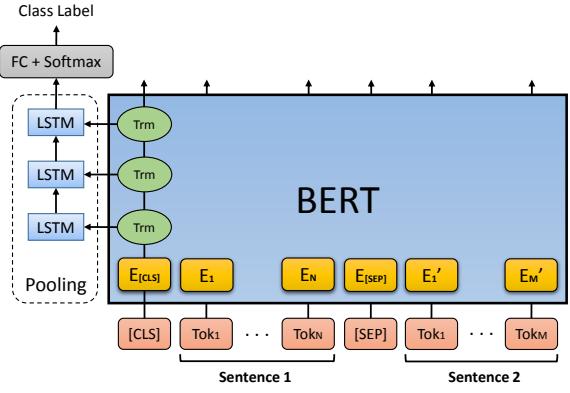


Figure 1: Overview of the proposed BERT-LSTM model. Pooling Module is responsible for connecting the intermediate representations obtained by Transformers of BERT.

LSTM-Pooling Representation of the hidden states \mathbf{h}_{CLS} is a special sequence: an abstract-to-specific sequence. Since LSTM network is inherently suitable for processing sequential information, we use a LSTM network to connect all intermediate representations of the [CLS] token, and the output of the last LSTM cell is used as the final representation. Formally,

$$o = h_{\text{LSTM}}^L = \overrightarrow{\text{LSTM}}(h_{\text{CLS}}^i), i \in [1, L]$$

Attention-Pooling Intuitively, attention operation can learn the contribution of each h_{CLS}^i . We use a dot-product attention module to dynamically combine all intermediates:

$$o = W_h^T \text{softmax}(\mathbf{q} \mathbf{h}_{\text{CLS}}^T) \mathbf{h}_{\text{CLS}}$$

where W_h^T and \mathbf{q} are learnable weights.

Finally, we pass the pooled output o to a fully-connected layer for label prediction:

$$y = \text{softmax}(W_o^T o + b_o)$$

3 Experiments

In this section, we present our methods for BERT-based model fine-tuning on three ABSA datasets. To show the generality, we also conduct experiments on a large and popular NLI task. We also apply the same strategy to existing state-of-the-art BERT-based models and demonstrate the effectiveness of our approaches.

3.1 Datasets

This section briefly describes three ABSA datasets and SNLI dataset. Statistics of these datasets are shown in Table 1.

Dataset	#Train	#Dev	#Test	#Label
Aspect Based Sentiment Analysis (ABSA)				
Laptop	2.1k	0.2k	0.6k	3
Restaurant	3.2k	0.4k	1.1k	3
Twitter	5.6k	0.6k	0.7k	3
Natural Language Inference (NLI)				
SNLI	549k	9.8k	9.8k	3

Table 1: Summary of the datasets. For ABSA dataset, we randomly chose 10% of #Train as #Dev as there is no #Dev in official dataset.

ABSA We use three popular datasets in ABSA task: Restaurant reviews and Laptop reviews from SemEval 2014 Task 4¹ (Pontiki et al., 2014), and ACL 14 Twitter dataset (Dong et al., 2014).

SNLI The Stanford Natural Language Inference (Bowman et al., 2015) dataset contains 570k human annotated hypothesis/premise pairs. This is the most widely used entailment dataset for natural language inference.

3.2 Experiment Settings

All experiments are conducted with BERT_{BASE} (uncased)² with different weights. During training, the coefficient λ of \mathcal{L}_2 regularization item is 10^{-5} and dropout rate is 0.1. Adam optimizer (Kingma and Ba, 2014) with learning rate of 2e-5 is applied to update all the parameters. The maximum number of epochs was set to 10 and 5 for ABSA and SNLI respectively. In this paper, we use 10-fold cross-validation, which performs quite stable in ABSA datasets.

Since the sizes of ABSA datasets are small and there is no validation set, the results between two consecutive epochs may be significantly different. In order to conduct fair and rigorous experiments, we use 10-fold cross-validation for ABSA task, which achieves quite stable results. The final result is obtained as the average of 10 individual experiments.

The SNLI dataset is quite large, so we simply take the best-performing model on the development set for testing.

3.3 Experiment-I: ABSA

Since BERT outperforms previous non-BERT-based studies on ABSA task by a large margin,

¹The detailed introduction of this task can be found at <http://alt.qcri.org/semeval2014/task4>.

²<https://github.com/huggingface/pytorch-pretrained-BERT>.

Domain	Laptop		Restaurant		Twitter	
	Acc.	F1	Acc.	F1	Acc.	F1
BERT _{BASE}	74.66	68.64	81.92	71.97	72.46	71.04
BERT-LSTM	75.31	69.37	82.21	72.52	73.06	71.61
BERT-Attention	75.16	68.76	82.38	73.22	73.35	71.88
BERT-PT (Xu et al., 2019)	76.27	70.66	84.53	75.33	-	-
BERT-PT-LSTM	77.08	71.65	85.29	76.88	-	-
BERT-PT-Attention	77.68	72.57	84.92	75.89	-	-

Table 2: Accuracy and macro-F1 (%) for aspect based sentiment analysis on three popular datasets.

we are not going to compare our models with non-BERT-based models. The 10-fold cross-validation results on ABSA datasets are presented in Table 2.

The BERT_{BASE}, BERT-LSTM and BERT-Attention are both initialized with pre-trained BERT_{BASE} (uncased). We observe that BERT-LSTM and BERT-Attention outperform vanilla BERT_{BASE} model on all three datasets. Moreover, BERT-LSTM and BERT-Attention have respective advantages on different datasets. We suspect the reason is that Attention-Pooling and LSTM-Pooling perform differently during fine-tuning on different datasets. Overall, our pooling strategies strongly boost the performance of BERT on these datasets.

The BERT-PT, BERT-PT-LSTM and BERT-PT-Attention are all initialized with post-trained BERT (Xu et al., 2019) weights³. We can see that both BERT-PT-LSTM and BERT-PT-Attention outperform BERT-PT with a large margin on Laptop and Restaurant dataset⁴. From the results, the conclusion that utilizing intermediate layers of BERT brings better results is still true.

3.3.1 Visualization of Intermediate Layers

In order to visualize how BERT-LSTM⁵ benefits from sequential representations of intermediate layers, we use principal component analysis (PCA) to visualize the intermediate representations of [CLS] token, shown in figure 2. There are three classes of the sentiment data, illustrated in blue, green and red, representing positive, neutral and negative, respectively. Since the task-specific information is mainly extracted from the last six layers of BERT, we simply illustrate the last six layers. It is easy to draw the conclusion that BERT-LSTM partitions different classes of

³Since our evaluation method is different from Xu et al., (2019), we post the results based on our experiment settings.

⁴Experiments are not conducted on Twitter dataset for Xu et al., (2019) does not provide post-trained BERT weights on this dataset.

⁵BERT-LSTM is more suitable for visualizing intermediate layers.

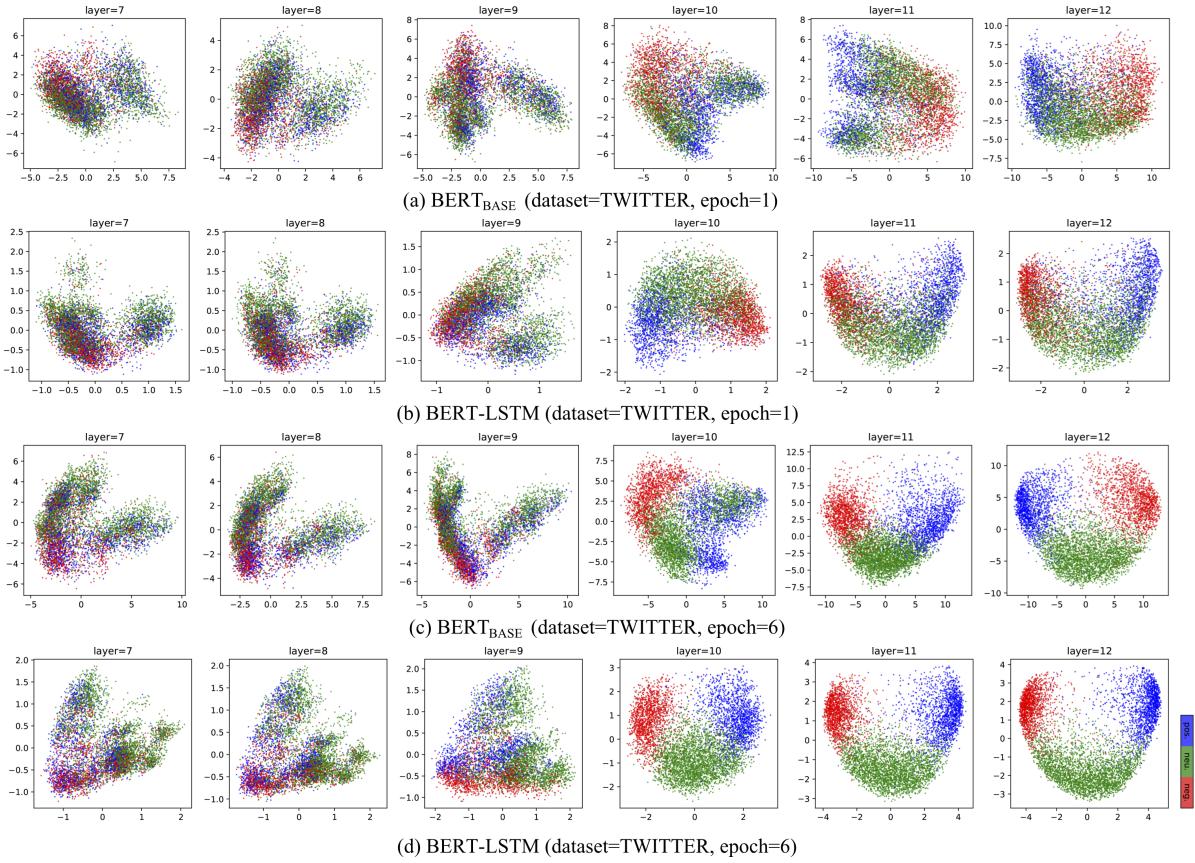


Figure 2: Visualization of BERT and BERT-LSTM on Twitter dataset with the last six intermediates layers of BERT at the end of the 1st and 6th epoch. Among the PCA results, (a) and (b) illustrate that BERT-LSTM converges faster than BERT after just one epoch, while (c) and (d) demonstrate that BERT-LSTM cluster each class of data more dense and discriminative than BERT after the model nearly converges.

Model	Dev	Test
GPT (Radford et al., 2018)	-	89.9*
Kim et al. (2018)	-	90.1*
BERT _{BASE}	90.94	90.66
BERT-Attention	91.12	90.70
BERT-LSTM	91.18	90.79
MT-DNN (Liu et al., 2019)	91.35	91.00
MT-DNN-Attention	91.41	90.95
MT-DNN-LSTM	91.50	90.91

Table 3: Classification accuracy (%) for natural language inference on SNLI dataset. Results with “*” are obtained from the official SNLI leaderboard (<https://nlp.stanford.edu/projects/snli/>).

data faster and more dense than vanilla BERT under the same training epoch.

3.4 Experiment-II: SNLI

To validate the generality of our method, we conduct experiment on SNLI dataset and apply same pooling strategies to currently state-of-the-art method **MT-DNN** (Liu et al., 2019), which is also a BERT based model, named **MT-DNN-Attention** and **MT-DNN-LSTM**.

As shown in Table 3, the results were con-

sistent with those on ABSA. From the results, BERT-Attention and BERT-LSTM perform better than vanilla BERT_{BASE}. Furthermore, MT-DNN-Attention and MT-DNN-LSTM outperform vanilla MT-DNN on Dev set, and are slightly inferior to vanilla MT-DNN on Test set. As a whole, our pooling strategies generally improve the vanilla BERT-based model, which draws the same conclusion as on ABSA.

The gains seem to be small, but the improvements of the method are straightforwardly reasonable and the flexibility of our strategies makes it easier to apply to a variety of other tasks.

4 Conclusion

In this work, we explore the potential of utilizing BERT intermediate layers and propose two effective pooling strategies to enhance the performance of fine-tuning of BERT. Experimental results demonstrate the effectiveness and generality of the proposed approach.

References

- Samuel R Bowman, Gabor Angeli, Christopher Potts, and Christopher D Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.
- Li Dong, Furu Wei, Chuanqi Tan, Duyu Tang, Ming Zhou, and Ke Xu. 2014. 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)*, volume 2, pages 49–54.
- Feifan Fan, Yansong Feng, and Dongyan Zhao. 2018. Multi-grained attention network for aspect-level sentiment classification. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3433–3442.
- Seonhoon Kim, Jin-Hyuk Hong, Inho Kang, and Nojun Kwak. 2018. Semantic sentence matching with densely-connected recurrent and co-attentive information. *arXiv preprint arXiv:1805.11360*.
- Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- Bing Liu. 2012. Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1):1–167.
- Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019. Multi-task deep neural networks for natural language understanding. *arXiv preprint arXiv:1901.11504*.
- Navonil Majumder, Soujanya Poria, Alexander Gelbukh, Md Shad Akhtar, Erik Cambria, and Asif Ekbal. 2018. Iarm: Inter-aspect relation modeling with memory networks in aspect-based sentiment analysis. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3402–3411.
- Bo Pang, Lillian Lee, et al. 2008. Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval*, 2(1–2):1–135.
- Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.
- Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. Semeval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 27–35.
- Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. URL https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/languageunsupervised/language_understanding_paper.pdf.
- Chi Sun, Luyao Huang, and Xipeng Qiu. 2019. Utilizing bert for aspect-based sentiment analysis via constructing auxiliary sentence. *arXiv preprint arXiv:1903.09588*.
- Hu Xu, Bing Liu, Lei Shu, and Philip S Yu. 2019. Bert post-training for review reading comprehension and aspect-based sentiment analysis. *arXiv preprint arXiv:1904.02232*.