Aspect-based Sentiment Analysis with Fine-tuned BERT

Fenghe Liu, Haoyu Zhang

School of Information, University of California, Berkeley

fenghe.liu@berkeley.edu, haoyuzhang@berkeley.edu

Abstract

Aspect-based sentiment analysis performs sentiment classification over multiple aspects. Apart from most recent work using graph neural networks such as relational graph attention network (R-GAT), this paper utilized the pretrained BERT to conduct classification tasks on SemEval 2014 datasets via several fine-tuning approaches. In order to achieve a comparable performance, several fine-tuning BERT approaches have been researched, which include selecting hidden states from intermediate layers, freezing various percentages of pre-trained model parameters and summing states of two hidden layers. The proposed BERT based method in this report achieves the classification accuracy of 86.61% and F1-score of 80.95%.

1. Introduction

Aspect-based sentiment analysis (ABSA) is a text analysis technique aimed at identifying and categorizing sentiments attributed to specific aspects. Aspect-based sentiment analysis can be used to analyze customer reviews by associating specific sentiments with different aspects of a product or service. With an increasing number of merchants and service providers transforming to internet platforms (e.g., Yelp and Amazon) to increase business presence, conducting ABSA can be beneficial for both business providers and internet platforms. For business providers, ABSA can help business owners to understand polarities towards various aspects of customer feedback and therefore identify the areas to improve the service. And for internet platforms, offering the ABSA advisory could be a good additional feature to attract and retain the business providers. Compared to traditional general sentiment classification techniques, aspect sentiment analysis offers fine-grained opinion polarity that provides more insight into associated aspects of the context. In summary, ABSA can provide better insights into user reviews compared with

traditional sentence-level sentiment analysis (Wang et al., 2020). In addition, the analysis can potentially help business providers improve their product or services more precisely.

In early works (Dong et al. 2014; Vo & Zhang, 2015), neural network methods have been employed for aspect-based sentiment analysis tasks (as cited in Zhang et al., 2019). More recently, the attention mechanism coupled with recurrent neural networks (RNNs) was implemented as a more promising method (Wang et al., 2016). However, Zhang et al. (2019) claimed the limitation of attention-based models that they are not able to parse syntactic dependencies within the sentence completely. This limit may cause a given aspect mistakenly connecting to syntactically unrelated context words. Aside from the RNNs, convolutional neural networks (CNNs) have also been employed to detect aspect related phrases (Xue & Li, 2018). However, the CNN-based models are incapable of determining the sentiments depicted by non-consecutive words (Zhang et al., 2019). To address the two limitations, several most recent ABSA works (Zhang et al., 2019; Wang et al., 2020) have used graph neural networks to learn representation from the dependency trees. Zhang et al. (2019) built the first graph convolutional network based model of comparable effectiveness for aspect-based sentiment classification. Wang et al. (2020) proposed a relational graph attention network (R-GAT) for the specific ABSA task in their work. The experiment results in Wang's (2020) work showed that R-GAT, basic BERT and R-GAT+BERT outperformed other mainstream methods with a remarkable margin.

In this paper, we propose multiple strategies for fine-tuning BERT over ABSA tasks, based on Wang's (2020) discovery and the fact that the self-attention mechanism in the Transformer enables BERT to resolve various downstream tasks (Devlin et al., 2018). Moreover, we conduct experiments on the SemEval 2014 restaurants dataset (Pontiki et al., 2014) for evaluation. In addition, the code part of this work is based on the pure BERT part of the source code released by Wang et al. (2020).

The contribution of this work include:

- A lightweight form of the BERT based model is proposed to accomplish ABSA tasks without sacrificing overall performance compared to other mainstream models (R-GAT, R-GAT+BERT).

- We verified that the fine-tuning BERT method is an inexpensive and straightforward approach to resolving ABSA tasks, which is of nearly the same performance as R-GAT+BERT.

2. Background

Recently, attention-based neural models are utilized in the majority of ABSA research (Wang et al, 2016; Devlin et al., 2018; Zhang et al, 2019; Wang et al, 2020). An attention-based LSTM was used in Wang's (2016) work to detect sentiment related to the target aspect. Wang et al (2016) discover that the attention mechanism can focus on different parts of a sentence when different aspects are taken as input. The milestone pre-trained language model BERT (Devlin et al., 2018) has succeeded in diverse classification tasks including ABSA. It is also demonstrated that the model fine-tuned on specific downstream tasks can benefit from the sufficiently pre-trained representations even when downstream task data is extremely small.

In the next year, Zhang et al. (2019) built an aspect-specific graph convolutional network (ASGCN) to capture both syntactical information and long-range word dependency. Most recently, considering that only a small part of the ordinary dependency tree is related to the ABSA task, Wang et al. (2020) defined a novel aspect-oriented dependency tree structure and proposed a relational graph attention network (R-GAT) to encode it. To make comparison to several mainstream models for ABSA, Wang has conducted experiments on the SemEval 2014 and twitter dataset (Dong et al., 2014) with three methods, which include R-GAT, basic BERT and R-GAT+BERT. The basic BERT is of the second best performance, outperforming all mainstream models including R-GAT by a big margin, while R-GAT+BERT leads by 0.98% in accuracy.

3. Methods

3.1 Data, Preprocessing and Baseline Accuracy

Research of this project is conducted on SemEval 2014 (Pontiki et al., 2014) dataset with an emphasis on task 2 - Aspect Term Polarity. The dataset contains restaurant reviews (train count: 3041, test count: 800) that are labeled with the associated aspect terms and categories along with the corresponding sentiments.

Based on the prior work done by Wang's team (2020), several steps were performed to preprocess the data. In the first step, filtering is implemented to select the training sets that contain labels for aspect terms. Secondly, the training and test data is unrolled into multiple representations of the sample review with each representation focusing on a specific aspect term to let the model focus on one aspect per sample. Eg. A restaurant review with two aspect terms on food and service, will be splitted into two sample rows relating to each aspect term. There are 3602 train samples after the unrolling process. Finally, the sample is transformed into a BERT supported input format: [CLS] token + review content + [SEP] token + aspect term + [SEP].

An evaluation of the test dataset shows a distribution of 65% negative aspect sentiment, 17.5% neutral sentiment and 17.5% negative sentiment out of 1120 unrolled test samples extracted from the original test dataset. This lays out the baseline accuracy of the model to be 65%.

3.2 Model Setup

The Pytorch implementation of $BERT_{base}$ is utilized for word representation and fine-tuned in this work. For the specific ABSA task, a downstream neural network of two fully connected layers is constructed on the top of the BERT. The Adam W algorithm is implemented for stochastic optimization, while the cross entropy is selected as the loss function.

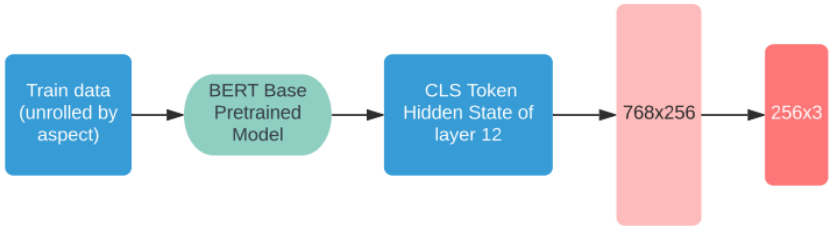

Figure 1. Default model structure for aspect sentiment analysis

3.3 Fine-tune BERT with Frozen Parameters

To fine-tune on the ABSA task, the only new parameter introduction is two fully connected layers weights W_1[768,256], W_2[[256,3] at the downstream of the BERT model, where 768 is the hidden size of the BERT, 3 is the number of the labels (figure 1). The Relu was selected as the activation function for the first linear layer. The model was fine-tuned for 30 epochs over the restaurants train data of SemEval 14 with a batch size of 16, learning rate of 1e-4, dropout rate of 0.1 and various percent of frozen BERT parameters (0, 10%, 25%, 50%, 75%, 100%). The best model performance on the test data was recorded with regard to the accuracy and F1 score.

3.4 Hidden Layer of BERT

Secondly, we explored the performance of the models based on hidden states from different layers of BERT. The hidden state of the classification token from each layer instead of the output layer of the BERT was exported to the same downstream architecture, which has been described in the previous paragraph. For this task, the model was fine-tuned for 30 epochs over the restaurants train data of SemEval 14, with the same hyperparameters as above and freezing no BERT layers.

3.5 Sum of Two Hidden Layers

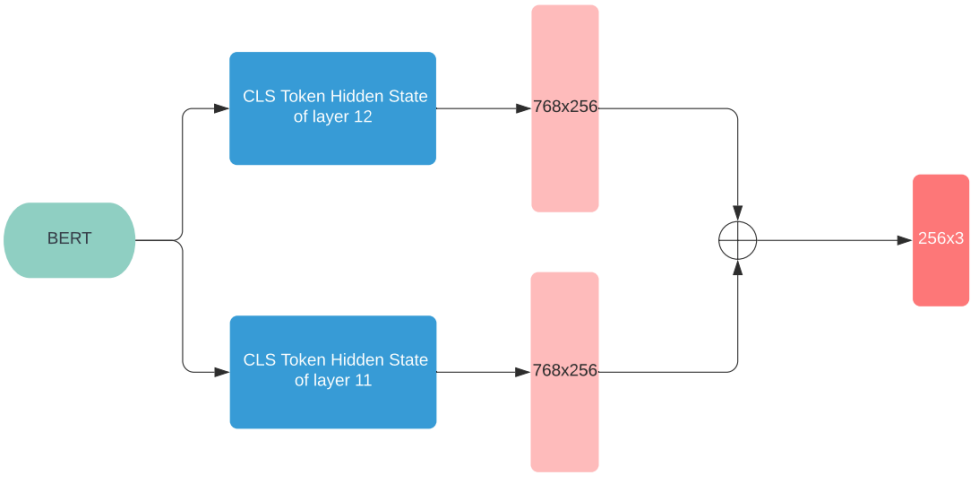

Figure 2. Model structure for two hidden layers

In addition, the potential improvement by sum of two hidden states of the classification token has also been investigated. As shown in figure 2, the downstream model architecture is similar to the one in the previous task. The only modification is that before going through the activation function and the final linear layer, the linear transformation of the two hidden layer states have been summed together. For this task, the model was also fine-tuned for 30 epochs over the restaurants train data of SemEval 14 with the above hyperparameters, different hidden layers combination and various percent of frozen BERT parameters (0, 25%, 50%, 75%).

3.6 Device

All the fine-tuning tasks mentioned above were implemented in a virtual machine with one Tesla V100 GPU of 16 GB memory. The estimated training time of the fine-tuning tasks (epoch=30) is around 10 minutes.

4. Results and Discussion

4.1 Effect of hidden layer selection

The first experiment was conducted by selecting hidden state outputs [CLS] from each layer in the BERT model and feeding it to the downstream linear fully connected network. The accuracy and F1 score measured with test data is shown in Figure 3. There are 13 layers that have been investigated in total, which include the embedding output layer as the layer 0 and the 12 hidden layers of the BERT base model. It is observed that the embedding layer offers about 65% accuracy which is consistent with the baseline accuracy of the training set. This is expected as the embedding layer does not capture any syntactic information of the review sentences. As shown in Figure 3, the accuracy and F1 score increase as the hidden layer number increases in general. The performance peaks at layer 11 with an accuracy of 86.25% and F1 score of 79.84% (Table 1 in appendix). The confusion matrix from Table 2 gives reasonable predictions for each sentiment class, with neutral prediction slightly underperformed and biased to positive sentiment prediction. Overall, the result is within expectation since in deep learning networks, the layers towards output may retain more information learned from the prior layers. In terms of why the peak happened at layer 11, our assumption is that the syntactic relationship within the sentence has been captured in layer 10 or 11 is sufficient for the downstream ABSA tasks.

In addition, the size of the training dataset is comparably small, and the result could be tied to certain bias or characteristics of the test dataset. BERT is a big model with 110 million parameters to train. Therefore, even though the sufficiently pretrained BERT model is capable of fine-tuning tasks with small datasets (Devlin et al., 2018), the test dataset may still be limited to optimize parameters of all layers.

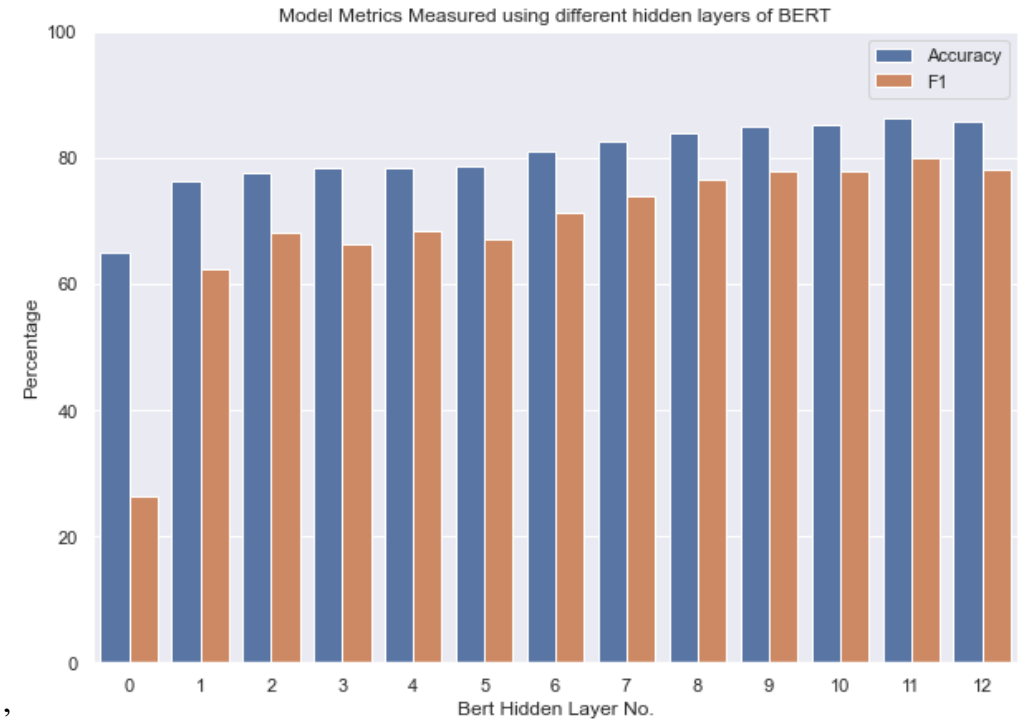

Figure 3. Overall performance of fine-tuning BERT with different hidden layer

4.2 Effect of parameters frozen rate

With the question in mind from the first experiment that the size training dataset could be one of the considerations for fine tuning the BERT model, our second research task was conducted to understand the relationship between model performance and freezing a percentage of parameters in the BERT model. The model is trained using layer 12 or the default output layer of the BERT model. As shown in Table 3 and Figure 4, the performance of the model is in a close range when the parameter frozen percentage is between 0%-75%, and then decreases when the parameter frozen percentage moves towards 100%. As the frozen parameter percentage is enforced sequentially based on the order of layer numbers, the results suggest that for most of the

pretrained BERT model parameters fine tuning happens towards the last 25% percent of layers or from layer 9 to layer 12. From our test, the highest accuracy we get from the model is 86.25% with 79.6% on F1 score, which keeps the first 25% of BERT parameters frozen. The confusion matrix from Table 4 shows that the sentiment classification is fairly predicted, and neutral sentiment prediction has slightly better results compared to the values from section 4.1.

Another finding from this experiment is that the training time decreases when increasing the parameter frozen percentage. In our test, we witnessed a 22% decrease in training time for every 25% increase in frozen parameters. The result shows that it is possible to add a small frozen percentage to perform an ABSA task while achieving a similar level of accuracy achieved by no frozen parameters.

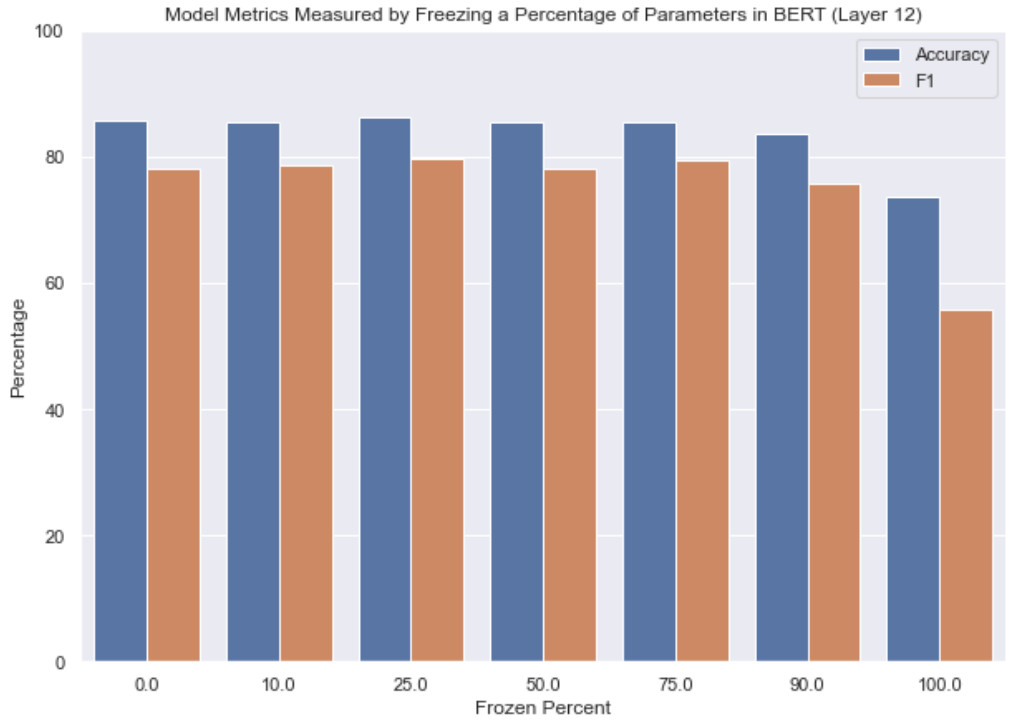

Figure 4. Overall performance of fine-tuning BERT with frozen parameter

4.3 Effect of summing two layers

The third research explored the effect of summing two hidden layers within BERT in order to check if combining the output of multiple layers could offer better results in training the model. The detailed model setup is explained in section 3.5. The experiments were first conducted on

the sum of adjacent layers, and then the sum of the layer after another. After that, tests were performed over different BERT parameter frozen percentages.

From the result in Table 5, we can observe that the weighted sum of two hidden layers does not offer significant improvement of the overall performance. And as shown from Table 6 - 8, the model performance decreases when the parameter frozen percentage increases within the 50%-75% range. The best model performance achieved in this experiment is 86.61% which is slightly better compared to the result collected from 4.1 and 4.2. The combinations in model setup that yield best performance is pairing 1ayer 12 and layer 10 with 0% parameter frozen percentage which is listed in Table 5. The confusion matrix of the best performing pair is included in Table 9 and the classification result shows a comparably more balanced accuracy result across three sentiment classes. In specific, the accuracy of neutral sentiment prediction is 68% which is about 10% higher than the best results retained from first and second experiments. A possible explanation for the observed improvement is that hidden states from different BERT layers capture the syntactic relationship between aspect term and review sentiment from relatively different views; to some extent, information encoded in these hidden states could complement each other for the ABSA task.

5. Conclusion

In this work, we proposed a relatively light model to accomplish the ABSA task by fine-tuning the pretrained BERT model with 3 different approaches. The experiment conducted on the SemEval 2014 restaurant review dataset proves that the fine-tuned BERT model produces comparably performance as the more-complicated methods including R-GAT+BERT (Table 10). The best performance achieved in this work is 86.61% accuracy and 80.95% F1-score, while R-GAT+BERT (accuracy = 86.60%, F1-score = 81.35%) only outperforms in F1-score by less than 0.4%. It is also demonstrated that the hidden state of BERT's intermediate hidden layers is sufficient for extracting syntactic information for the ABSA task.

To summarize, three findings were concluded after fine-tuning the BERT model. Firstly, the result shows that the model performance generally increases when further downstream BERT hidden layers are selected as input to the linear layer. However, the best training result may not necessarily come from the last hidden layer, and the layer selection should be analysed with the

understanding of the practical task and training data characteristics. Secondly, it is found that fine-tuning on hidden layers closer to the top of the BERT model is more effective. Therefore, it might be reasonable to enforce a parameter frozen percentage (0-25%) for the bottom hidden layers when training the model with relatively small datasets. In our case, applying the 25% level of parameter frozen percentage does not sacrifice model performance while decreasing the training time by 22%. Finally, replacing a single hidden layer with the weighted sum of two hidden layers did not offer significant improvement in model performance but did generate the best accuracy (86.61%) across our research. It is worth mentioning that the weighted sum method can potentially generate a confusion matrix that has more balanced accuracy among different classes compared to the prior two approaches. However, the results retained from this research can be limited by the training data size and characteristics. The findings could be further validated with more extensive testing on alternative datasets.

References

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.

Dong, L., Wei, F., Tan, C., Tang, D., Zhou, M., & Xu, K. (2014). Adaptive recursive neural network for target-dependent Twitter sentiment classification. Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). doi:10.3115/v1/p14-2009

Pontiki, M., Galanis, D., Pavlopoulos, J., Papageorgiou, H., Androutsopoulos, I., & Manandhar, S. (2014). Semeval-2014 task 4: Aspect based sentiment analysis. In Proceedings of the 8th International Workshop on Semantic Evaluation (Se- mEval 2014), 27–35.

Wang, K., Shen, W., Yang, Y., Quan, X., & Wang, R. (2020). Relational graph attention network for aspect-based sentiment analysis. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. doi:10.18653/v1/2020.acl-main.295

Wang, Y., Huang, M., Zhu, X., & Zhao, L. (2016). Attention-based LSTM for aspect-level sentiment classification. Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. doi:10.18653/v1/d16-1058

Xue, W., & Li, T. (2018). Aspect based sentiment analysis with gated Convolutional networks. Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). doi:10.18653/v1/p18-1234

Zhang, C., Li, Q., & Song, D. (2019). Aspect-based sentiment classification with aspect-specific graph Convolutional networks. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). doi:10.18653/v1/d19-1464

Appendix

Table 1

Overall performance of fine-tuning BERT with different hidden layer

Hidden Layer No.	*Accuracy*	*F1*
12	0.8580	0.7823
11	**0.8625**	**0.7984**
10	0.8527	0.7791
9	0.8482	0.7776
8	0.8384	0.7648
7	0.8259	0.7403
6	0.8089	0.7116
5	0.7857	0.6711
4	0.7848	0.6839
3	0.7848	0.6624
2	0.7759	0.6818
1	0.7634	0.6241
0	0.65	0.2626

Table 2

Confusion Matrix of Model using Layer 11 from BERT

	Negative_Pred	*Positive_Pred*	*Neutral_Pred*
Negative	163	19	14
Positive	14	720	12
Neutral	24	73	99

Table 3

Overall performance of fine-tuning BERT with frozen parameter

Frozen Percentage	*Accuracy*	*F1*
0	0.8580	0.7823
10%	0.8554	0.7861
25%	**0.8625**	**0.7960**
50%	0.8554	0.7807
75%	0.8545	0.7955
90%	0.8375	0.7581
100%	0.7366	0.5573

Table 4

Confusion Matrix of Model using layer 12 with parameter frozen percentage of 25%

	Negative_Pred	*Positive_Pred*	*Neutral_Pred*
Negative	157	23	16
Positive	14	695	19
Neutral	24	58	114

Table 5

Overall performance of different hidden layers pair with BERT parameter frozen percentage = 0%

Two Hidden Layers Pair	*Accuracy*	*F1*
(12, 11)	0.8607	0.7952
(11, 10)	0.8571	0.7857
(10, 9)	0.8446	0.7740
(9, 8)	0.8420	0.7598
(8, 7)	0.8188	0.7234
(12, 10)	**0.8661**	**0.8095**
(11, 9)	0.8580	0.7948
(10, 8)	0.8625	0.7969

Table 6

Overall performance of different hidden layers pair with BERT parameter frozen percentage = 25%

Two Hidden Layers Pair	*Accuracy*	*F1*
(12,11)	0.8607	0.7883
(11,10)	0.8625	0.7998
(10,9)	0.8509	0.7739
(9,8)	0.8464	0.7780
(8,7)	0.8330	0.7422

Table 7

Overall performance of different hidden layers pair with BERT parameter frozen percentage = 50%

Two Hidden Layers Pair	*Accuracy*	*F1*
(12,11)	0.8482	0.7653
(11,10)	0.8607	0.7963
(10,9)	0.8464	0.7689
(9,8)	0.8455	0.7703
(8,7)	0.8196	0.7252

Table 8

Overall performance of different hidden layers pair with BERT parameter frozen percentage = 75%

Two Hidden Layers Pair	*Accuracy*	*F1*
(12,11)	0.8509	0.7836
(11,10)	0.8384	0.7542
(10,9)	0.8304	0.7424
(9,8)	0.7080	0.5027
(8,7)	0.7107	0.5127

Table 9

Confusion matrix - BERT layer 12 + 10 with parameter frozen percentage of 0%

	Negative_Pred	*Positive_Pred*	*Neutral_Pred*
Negative	149	25	22
Positive	8	688	32
Neutral	15	48	133

Table 10

Overall performance of different methods on the SemEval 2014

	Method	*Accuracy*	*F1*
Wang et al.	R-GAT	0.8330	0.7608
Wang et al.	R-GAT+BERT	0.8660	0.8135
Ours	Fine-tune BERT, Single Hidden Layer	0.8625	0.7984
Ours	Fine-tune BERT, Two Hidden Layers	0.8661	0.8095