

Analysis Report

(Amit Panwar, M22AIE202)

I am working on a task to train original Bert model with some change known as Adapter Bert.

In this model authors of the paper “Parameter-Efficient Transfer Learning for NLP” have used a methodology known as Adapter Bert where original Bert model is modified with some additional parameters and training model with less data while achieving the similar benchmark performance.

Original works from this paper claims to produce results as below on MRPC task:

***** Eval results *****

eval_accuracy = 0.85784316

eval_loss = 0.48347527

global_step = 573

loss = 0.48347527

I have forked original GitHub repository and created my own branch as below:

Original repo: <https://github.com/google-research/adapter-bert>

Assignment repo: <https://github.com/amitpanwarIndia/adapter-bert-assignment>

Original Adapter Bert repo is supported only with TensorFlow version 1.15; however, I had to upgrade and modify the code to latest TensorFlow version using **tf_upgrade_v2 script** provided by TensorFlow and by doing some code changes to python files.

Original Algorithm:

Original Adapter Bert works with MRPC task; it trains, eval and test on this task with following inputs, it adds parameter to original Bert as a feedforward adapter layer with a bottleneck.:

--task_name=MRPC \

--do_train=true \

--do_eval=true \

--data_dir=/content/GlueData/MRPC \

--vocab_file=/content/uncased_L_12_H_768_A_12/vocab.txt \

```
--bert_config_file=/content/uncased_L_12_H_768_A_12/bert_config.json \  
--init_checkpoint=/content/uncased_L_12_H_768_A_12/bert_model.ckpt \  
--max_seq_length=128 \  
--train_batch_size=32 \  
--learning_rate=3e-4 \  
--num_train_epochs=5.0 \  
--output_dir=/tmp/adapter_bert_mrpc/
```

GitHub colab file: https://github.com/amitpanwarIndia/adapter-bert-assignment/blob/master/AML_Assignment_2.ipynb

When I trained upgraded code in google colab then I got below results:

***** Eval results *****

eval_accuracy = 0.67741936

eval_loss = 0.6297616

global_step = 636

loss = 0.6298113

Modified Algorithm:

Modified Adapter Bert works with WNLI task; it trains, eval and test on this task with following inputs, it adds parameter to original Bert as a feedforward adapter layer with a bottleneck.:

```
--task_name=WNLI \  
--do_train=true \  
--do_eval=true \  
--data_dir=/content/GlueData/WNLI \  
--vocab_file=/content/uncased_L_12_H_768_A_12/vocab.txt \  
--bert_config_file=/content/uncased_L_12_H_768_A_12/bert_config.json \  
--init_checkpoint=/content/uncased_L_12_H_768_A_12/bert_model.ckpt \  
--max_seq_length=128 \  

```

```
--train_batch_size=32 \  
--learning_rate=3e-4 \  
--num_train_epochs=5.0 \  
--output_dir=/tmp/adapter_bert_wnli/
```

GitHub colab file: https://github.com/amitpanwarIndia/adapter-bert-assignment/blob/master/Latest_AML_Assignment_2_Modified.ipynb

When I trained upgraded code in google colab then I got below results:

```
***** Eval results *****
```

```
eval_accuracy = 0.5633803
```

```
eval_loss = 0.6919873
```

```
global_step = 99
```

```
loss = 0.6919852
```

Analysis: When I ran both results then those looks similar in terms of loss and accuracy, however original Adapter Bert was slightly better performing than currently produced results. The reason for this might be upgrade we had to do of TensorFlow from 1.15 to 2.2 where newer version is supporting old code with help of compatible library not actual functions of Tensorflow 2.2.

Working demo can be seen in GitHub ipynb files as result.