CSCI 5980 (F22) HW1: Replication of Text Classifier

Github: HarBatt/CSCI_5980_HW1

Saturday 15th October, 2022

Contents

1 Introduction

1.1 Description

The goal of this homework is to replicate of Text Classifier using the models and framework provided by Huggingface. I am working with my team on sentiment analysis so I chose this task for homework 1. For sentiment classification problem, I used DistilledBERT Model proposed by [1] which is a smaller and faster version of BERT (Deep Bidirectional Transformers) proposed by [2]. V. Sanh et al, used Knowledge Distillation technique to train a student BERT model which is smaller, reduced with similar architecture like BERT and by using larger BERT model as a teacher. They got a significant performance increase interms of speed while retaining most of the performance on downstream tasks. They also quoted this gives an edge on BERT when we are working on a low resource environments like edge devices or agents. I used the pre-trained DistilledBERT model provided **here** by Huggingface for this homework.

1.2 Dataset

We were also provided links to the datasets for corresponding tasks, I used SST2 dataset by Stanford which is also included in Huggingface dataset library. It has 67349 train, 872 validation, and 1821 test samples in it as a JSON file with an index, sentence and a label.

2 Performance

I trained the model for 10 epochs using Train data and evaluated on Evaluation dataset, given below is the figure for train loss vs steps.

2.1 Training Loss vs Steps

I used WandB chart to monitor the training process and the loss vs step chart can be found **here**, and is given below as Figure 1. More information can be found **here**.

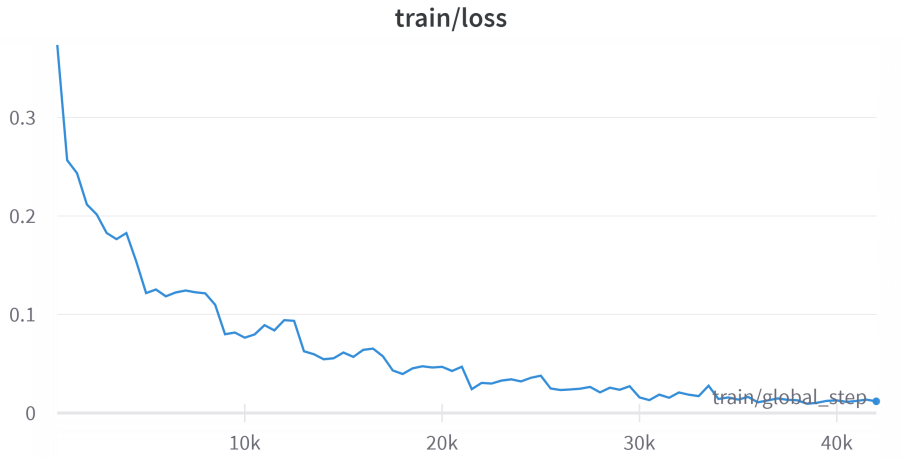

Figure 1: CrossEntropy loss on Y-Axis vs Number of steps on X-Axis

2.2 Performance Comparision

Unfortunately, the testset for SS2 dataset is unlabelled but I used some part of the evaluation dataset for testing and I got an accuracy of 0.91, precision: 0.90, recall: 0.92, f1: 0.91 and the performance matches with the leaderboard results and DistilledBERT performance reported in the paper which is accurcy of 91.3 %.

2.3 Training and Inference Time

It took me 28 minutes to train the model for 10 epochs at 27.25 it/s on Google Colab, and the average compute time for classifying a sample is an upper-bound of 0.018-0.02 seconds.

3 Observations

Unfortunately, the testset for SS2 dataset is unlabelled but I used some part of the evaluation dataset for testing and I got an Accuracy of 0.91, Precision around 0.90, Recall around 0.92, and F1 score of 0.91 and the performance matches with the leaderboard results and DistilledBERT performance reported in the paper which is accuracy of 91.3 %.

3.1 Hyperparameters

I used the a learning rate of 5×10^{-5} as fine-tuning requires low learning rate as per the articles that I read on Stack Overflow and Huggingface discussion board. I used batch size of 16 due to the GPU memory constraints, and batch size of 32 was used in the official implementation. The weight decay was set to 0.01 for good convergence. I tried different hyperparameter settings like higher learning rate and batch size, the performance was relatively lower or remained same from this setting.

3.2 Incorrect Samples

Given below are some of the sentences and wrongly assigned classes by the model. Here G.T expands to ground truth.

No.	Sentence	Prediction	G.T
1	teen movies have really hit the skids .	Positive	Negative
2	good film , but very glum .	Negative	Positive
3	this flick is about as cool and crowd-pleasing as a documentary can get .	Negative	Positive
4	what better message than ‘ love thyself ’ could young women of any size receive ?	Negative	Positive
5	no telegraphing is too obvious or simplistic for this movie .	Positive	Negative
6	as unseemly as its title suggests . .	Negative	Positive
7	you won’t like roger , but you will quickly recognize him .	Positive	Negative
8	but it still jingles in the pocket ..	Negative	Positive
9	sam mendes has become valedictorian at the school for soft landings and easy ways out .	Positive	Negative
10	manages to show life in all of its banality when the intention is quite the opposite .	Positive	Negative

After looking at these examples, we can notice that the model sometimes fails to capture the underlying tone of the speaker, sentence 7 indicates a warning sign but the model did not capture that. But the model did really well in some of the ambigious sentences like sentence 8, as even the human reader might consider it to be negative. We can observe an underlying hidden message in most of these sentences which the model failed to capture. Usually a human would end with a positive statement when they start with sentence 6 and sentence 5 suggests there is a prior expectation on the movie which the model fails to understand.

3.3 Potential Modeling or Representation Ideas to Improve the Errors

My team and I are working on looking at the sentence from a parse tree perspective where the dependency relations between the words in a sentence might come in play when we are looking at the underlying structure. Our plan is to use a graph based self supervision framework to pre-train a graph neural network to capture structural information from the dependency trees.

References

[1] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, “Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter,” 2019.

[2] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding.”