Emotion Detection from Text using BERT

Submitted by:

Savan Amitbhai Visalpara (sxv180069)

Code:

https://github.com/savan77/EmotionDetectionBERT

Content

Introduction

This project was done as part of the Natural Language Processing class. The goal here is to train a neural network to classify emotions in a given text. One text might have a multi emotions in it. So, this is a multi-label classification model. This report explains the model used in this project and also reports the metrics.

Data Preparation

The data was provided in the json files. However, my code expects data to be in the csv files. So, I first had to convert json files into csv files. This can be done by running following command.

```
python .\data_generator.py --file=D:\UTD\Assignment\NLP\project\nlp_test.json --csvfile=D:\UTD\Assignment\NLP\project\nlp_test.csv
```

This will generate the csv files and store it in the provided path. It was also came to notice that there is a class imbalance in provided training data. Due to the tight timeline, we couldn't explore more in this domain. But we can solve this problem either by oversampling minority class or providing weights to the loss function.

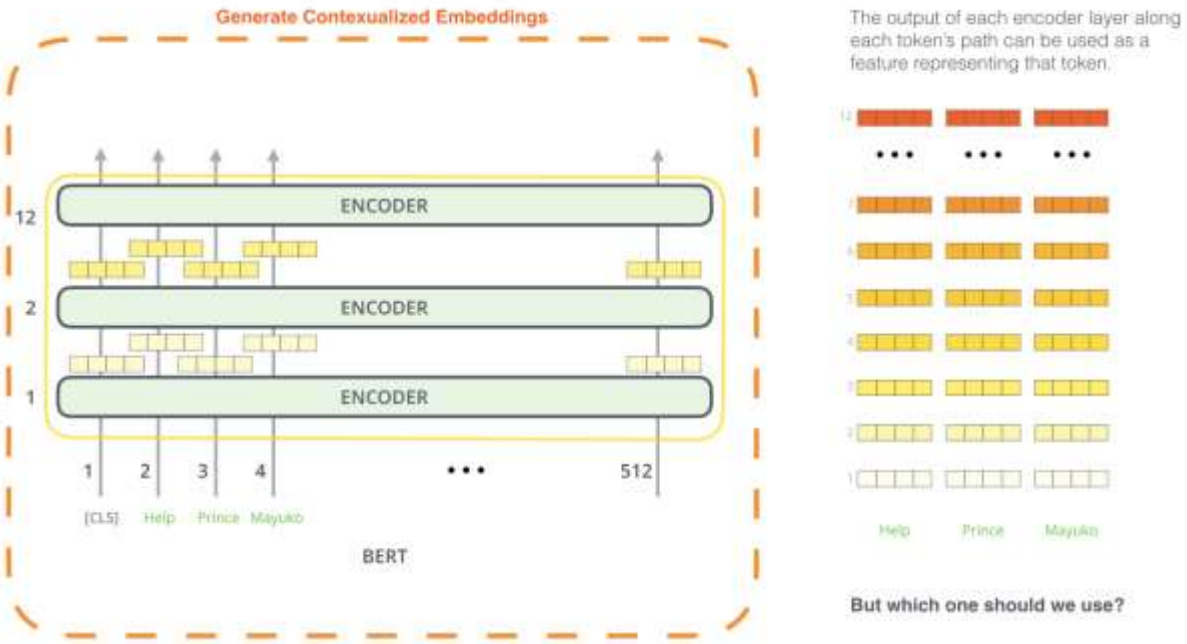
Model

For this project, we chose to train a BERT model on provided train set. BERT stands for Bidirectional Encoder Representations from Transformers and is the current buzz word in the field of natural language processing. We use the Bert Base model which has twelve layers in Transformer Encoder stack.

We used uncased model since they generally perform better than cased model.

Below graph represents basic flow of BERT model.

As we can see, there is a stack of Encoders inspired by the Transformers. This stack of encoders generates an encoding vector that can be passed to **task-specific** module or layer (i.e classification layer) to generate the output.



Training

To train this model, we used a library called fast-bert, which is a wrapper around famous transformers library from HuggingFace. There were certain parameters that we needed to find to create train the best possible model. The main hyper-parameters that we were interested in are : warmup_steps, learning_rate, schedule_type, and epochs.

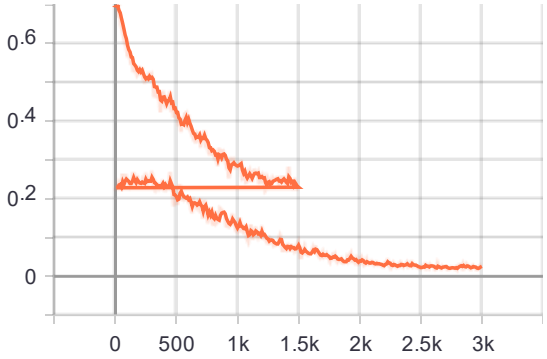
- Warmup steps: warmup steps defines how to increase/decrease the learning rate over time. This also depends on the scheduler. Basically, depending upon the scheduler, the scheduler linearly increased the learning from 0 to 1 over warmup_steps.
- Learning rate: Learning rate defines the size of the step to be taken in the opposite direction of the gradient. We tried with multiple learning rate and found that 2e-3 works the best.
- Scheduler Type: Scheduler type defines how to increase/decrease the learning rate over time. We found that wamup_cosine_hard_restarts works the best.

Training can be started by running following command.

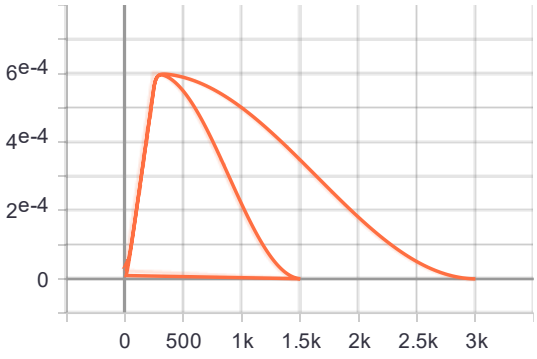
```
python train_bert.py --epochs=20
```

Below graphs show loss and learning rate for various hyper parameters.

1. Warmup steps = 250 Learning rate = 6e-4 Epochs =20

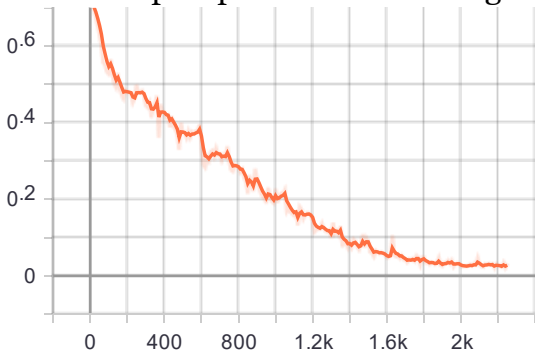


[Loss]

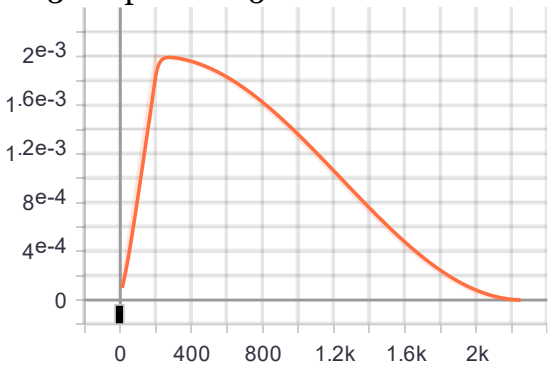


[Learning Rate]

2. Warmup steps = 200 Learning rate = 2e-3 Epochs =15

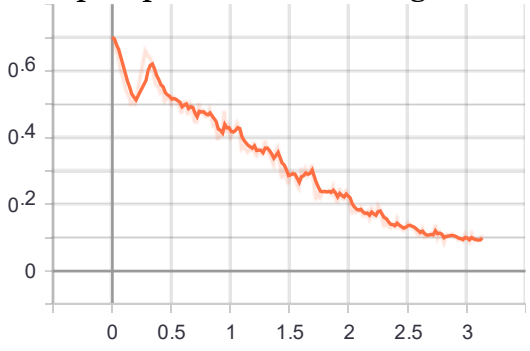


[Loss]

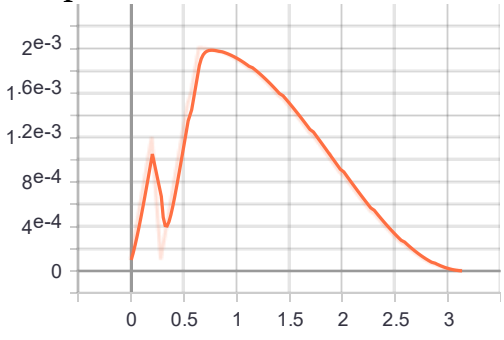


[Learning Rate]

3. Warmup steps = 200 Learning rate = 2e-3 Epochs =10

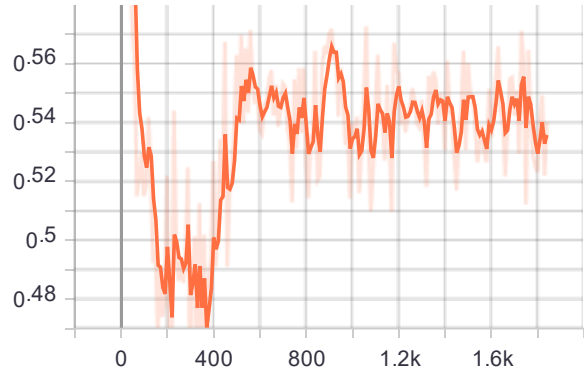


[Loss]

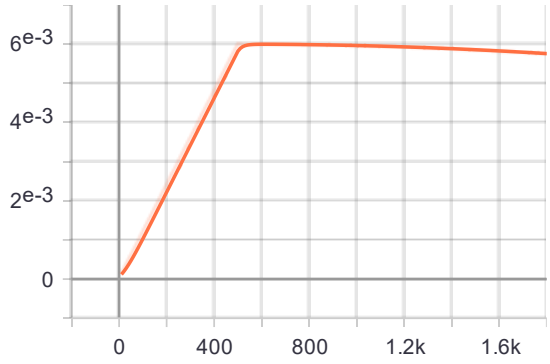


[Learning Rate]

4. Warmup steps = 500 Learning rate = 6e-3 Epochs =15



[Loss]



[Learning Rate]

Evaluation

Evaluation can be done by running following command. Pretrained model can be downloaded from here:
https://utdallas.box.com/s/sqqb0n9qe7txb6j3725aiz76gwlmszuw

python inference.py --test_csv=D:\\UTD\\Assignment\\NLP\\project\\nlp_test.csv --evaluation=True

Below table shows our evaluation on train and test set per class.

Emotion	**Precision**	**Recall**	**F1-Score**
Anger	0.97	0.95	0.96
Anticipation	0.98	1.00	0.99
Disgust	0.97	0.96	0.97
Fear	1.00	1.00	1.00
Joy	0.93	0.93	0.93
Love	1.00	0.73	0.85
Optimism	0.91	1.00	0.95
Pessimism	0.98	0.94	0.96
Sadness	0.99	0.92	0.95
Surprise	0.98	0.92	0.95
Trust	0.95	0.88	0.91
Neutral	0.92	1.00	0.96
Average	**0.98**	**0.97**	**0.97**

Following table shows information about test set.

Emotion	Precision	Recall	F1-Score
Anger	0.53	0.67	0.59
Anticipation	0.67	0.68	0.68
Disgust	0.62	0.78	0.69
Fear	0.69	0.72	0.71
Joy	0.50	0.27	0.35
Love	0.32	0.37	0.34
Optimism	0.37	0.59	0.45
Pessimism	0.44	0.73	0.55
Sadness	0.42	0.53	0.47
Surprise	0.55	0.38	0.45
Trust	0.12	0.12	0.12
Neutral	0.37	0.37	0.37
Average	**0.60**	**0.62**	**0.55**

We also found that various thresholds gave different accuracy. In general, 0.0017 worked the best.

References

https://medium.com/huggingface/introducing-fastbert-a-simple-deep-learning-library-for-bert-models-89ff763ad384

https://arxiv.org/abs/1810.04805

https://github.com/kaushaltrivedi/fast-bert