Documentation: Assignment InfoVision

1. Data pre-processing steps:

* Removed the other category data as it was insignificant in comparison to positives and negatives, also doesn’t correspond to neutral.
* Shuffled the dataset to add randomization, to reduce any chance of bias based on ids etc.
* BERT doesn't work with raw text. Need to pre-process the data:
* Add special tokens to separate sentences and do classification
* Pass sequences of constant length (introduce padding)
* Create array of 0s (pad token) and 1s (real token) called attention mask
* Check for special tokens using example sentence.
* Check for max token length, assign max\_token\_length based on observation

1. Model architecture and parameters:

I’ve used BERT base cased pretrained model from huggingface

Last hidden state is of shape [1, 32, 768]

Pooled output is of shape [1, 768]

Last hidden layer has 768 neurons

For sentiment analysis just added a new linear layer with 768 neurons to train for the given classes.

Added dropout of 0.3 for the pooled layer, as output for forward propagation.

Then added a softmax layer for the output.

1. Training process and hyperparameters:

For training used AdamW optimizer keeping other hyperparameters same, just kept learning rate as 2e-5 as one of the standard from paper.

Kept number of epoch as 10

1. Evaluation results and analysis:

precision recall f1-score support

negative 0.81 0.81 0.81 1117

positive 0.77 0.77 0.77 939

accuracy 0.79 2056

macro avg 0.79 0.79 0.79 2056

weighted avg 0.79 0.79 0.79 2056

Analysis:

Accuracy is not upto the mark but a good measure to start with.

Precision and Recall is decent and f1-score holds firm for the predictions for this model.

This training process is compute intensive and is not under free-tier so, couldn’t do much tuning to improve it, I hit my quota limit.

1. Instructions for using the deployed model (if applicable):

Unfortunately the model is not deployed.

If I had enough credits on my aws accounts I would have deployed the model and created an endpoint.

This public endpoint could have been used to run predictions over api calls.

Code References:

* [BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding](https://arxiv.org/abs/1810.04805)
* [L11 Language Models - Alec Radford (OpenAI)](https://www.youtube.com/watch?v=BnpB3GrpsfM)
* [The Illustrated BERT, ELMo, and co.](https://jalammar.github.io/illustrated-bert/)
* [BERT Fine-Tuning Tutorial with PyTorch](https://mccormickml.com/2019/07/22/BERT-fine-tuning/)
* [How to Fine-Tune BERT for Text Classification?](https://arxiv.org/pdf/1905.05583.pdf)
* [Huggingface Transformers](https://huggingface.co/transformers/)
* [BERT Explained: State of the art language model for NLP](https://towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-f8b21a9b6270)
* Book: getting things done with pytorch