Assignment 4

Sreemanti Dey & Si Siddhanth Raja

28 November 2022

1 CNN

Train accuracy: 90.23684210526316 %

Test accuracy: 13.543859649122808 %

Time taken for training: 3.5 hours

Workflow: We made 2 classes, one for handling the loading of data and the other one for carrying out the training. The layers were made as given in the pdf and finally we reported the train and test accuracy on the non competitive dataset.

Model details :

1. Batch size = 2
2. Learning rate = 0.001
3. Number of epochs = 4
4. Size of images = 224 x 224

Observations:

1. The training was done on GPU hence this was quite faster, than training on CPU.
2. We resized the images and ran the code, the code gave similar accuracy as the unresized version of it. Resizing of images made it faster to run.
3. We used SGD optimizer.

2 RNN

Train Accuracy: 58.5672514619883 %

Test Accuracy: 44.6140350877193 %

Time taken for training: 15 min

Workflow: We made 2 classes, one for handling the loading of data and forming the glove embeddings and the other one for carrying out the training. We used the glove model as given in the pdf and added the layers as mentioned, and finally reported the train and test accuracy on the non competitive dataset.

Model details :

1. Batch size = 20
2. Learning rate = 0.01
3. Number of epochs = 50
4. Sentence length = 15

Observations:

1. The training was done on GPU hence was much faster compared to CPU.
2. We used SGD optimizer (tried Adam as well, but SGD worked better).
3. Training time was much less compared to CNN part, since we are using text here instead of images.

3 Competitive

Training accuracy: 95.2134422675692 22%

Test accuracy (as seen on the public leaderboard): 65.576%

Model is Bert-large-uncased.

Model description : Bert is the state of the art language model for various NLP tasks. It applies bi-directional training of transformer, a popular attention model, to language modelling. Among different variations of Bert, we found uncased versions perform better than cased ones, since genre will not depend on casing of the words. Also, Bert large performs much better than Bert base.

Training the language model in BERT is done by predicting 15% of the tokens in the input, that were randomly picked. These tokens are pre-processed as follows — 80% are replaced with a "[MASK]" token, 10% with a random word, and 10% use the original word.

Features of Bert-large-uncased:

1. 24-layer
2. 1024 hidden dimension
3. 16 attention heads
4. 336M parameters

Workflow: We learned about the different architectures that are available for nlp tasks and found Bert to be the best, then we added some warmup steps, experimented with different optimizers, tweaked the parameters and finally reported the competitive test predictions.

Model training parameters:

1. Batch size = 50
2. Learning rate = 5e-5
3. Optimizer = AdamW
4. Max length = 64
5. Number of epochs = 5

Pre-processing/Augmentation:

1. We added the datasets - train and non_comp_test to train our data.
2. We used 90-10 split of training data for validation dataset.

3. We added number of warmup steps to be 600, so that our learning rate increases for some steps, then decreases later on (as we reach closer to the minima), but got lower accuracy so we removed this.

4. We tried removing stopwords and lemmatizing but got lower accuracy so we removed this.

Trials done:

1. We tried a lot of models including all possible varieties of Bert, Albert, Distilbert, xlnet, RoBerta, GPT2, Spectra, their cased and uncased versions, base and large versions.

2. We ran on a huge set of parameters.

   (a) Batch sizes were varied from 16 to 100.

   (b) Learning rates were varied among 1e-2, 1e-3, 1e-4, 1e-5, 3e-5, 5e-5, 6e-5.

   (c) Optimizers were varied among SGD, Adam and AdamW.

   (d) Max sequence length were varied among 32, 64, 72, 85, 128.

   (e) Warmup steps were varied among 0, 600, 900.

   (f) Number of epochs were varied among 2, 3, 4, 5, 6, 7, 8.

   (g) We checked with 80-20 and 90-10 split for validation dataset.

Thus, we tried tuning all possible hyperparameters to get the best accuracy.