Assignment 4 Report

## Ziyang Lin

1.Models Choices

In the ‘Funniness\_Estimation\_3rd\_version’, I experiment with the pre-trained language models *BERT(base-uncased)*, *ALBERT(base-v2)*, *ROBERTA(base)*, *ELECTRA*, *XLNET(base-cased)* provided by Huggingface and each of them has a head classification layer.

In the ‘Funniness\_Estimation\_4th\_version’, I carry out a trial in which a pre-trained *ALBERT(base-v2)* model without the head layer is firstly combined with a linear regression layer to learn the expected task-specific knowledge from a designed fake task using training and validation datasets. After training well for the fake task, the *ALBERT* model will get rid of the linear regression layer and add a new initialised classification layer to its head for the real classification task.

2.Preprocessing

In the ‘Funniness\_Estimation\_3rd\_version’, there are three versions of code designed to tackle the data preprocessing (the normal version, the cut-headlines version and the punctuation removal version respectively).

3.Hyperparameters

The below table shows all hyperparameters involving in the model training phrase:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Batch size | Number of Epoch | Learning Rate | Fine-tuning Rate | Epsilon | Warmup Ratio | Weight Decay |

There are also two versions of code for the general hyperparameters setting and I put them into two dictionaries. One applies weight decay to all parameters other than bias and layer normalization terms and uses a general learning rate. Another uses weight decay for all parameters while there are learning rate and the fine-tuning rate for the optimization of the head layer and the optimization of pretrain LM respectively.

4.Optimizer & Learning Rate Scheduler

The optimizer and learning rate scheduler I use are ‘AdamW’

and ‘get\_linear\_schedule\_with\_warmup’ from the Huggingface respectively.

For the ‘get\_linear\_schedule\_with\_warmup’ scheduler, the parameter ‘num\_training\_steps’ is computed by ‘[number of batches] x [number of epochs]’ and the parameter ‘num\_warmup\_steps’ is calculated using ‘[warmup ratio] x [num\_training\_steps]’

5.Experiment Results

Funniness\_Estimation\_3rd\_version:

The 1st log:

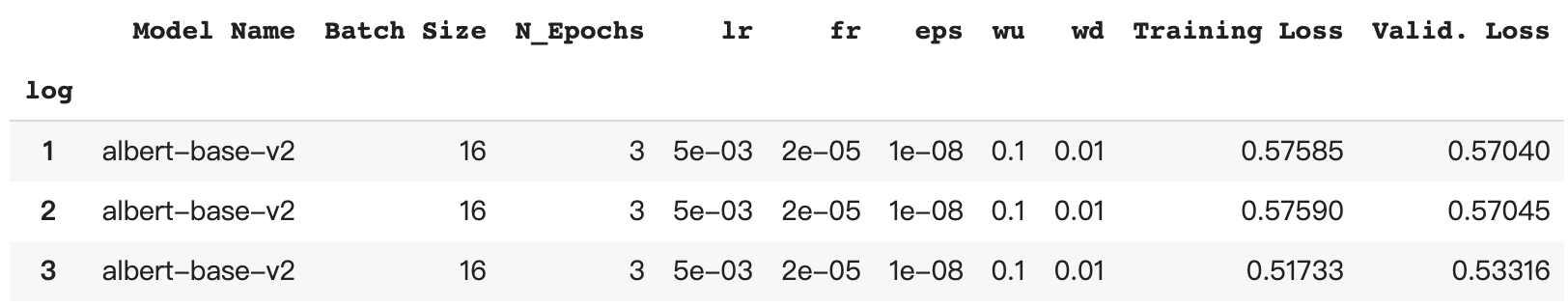
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Model Name | Batch Size | N\_Epochs | lr | eps | wus | wd | Training Loss | Training Accur. | Valid. Loss | Valid. Accur. | Testing Accur. | wu |
| electra | 16 | 3 | 5.00E-05 | 1.00E-08 | 500 | 0.01 | 0.9040663524 | 56.01575809 | 1.017055559 | 45.19988739 | 50.11415525 |  |
| electra | 16 | 3 | 3.00E-05 | 1.00E-08 | 500 | 0.01 | 0.7519826203 | 67.64267462 | 1.077755947 | 46.07263514 | 50.11415525 |  |
| bert-base-uncased | 16 | 3 | 3.00E-05 | 1.00E-08 | 500 | 0.01 | 0.7519826203 | 67.64267462 | 1.077755947 | 46.07263514 | 50.11415525 |  |
| bert-base-uncased | 32 | 3 | 3.00E-05 | 1.00E-08 | 500 | 0.01 | 0.7409755295 | 68.22916667 | 1.09057613 | 46.98168563 | 50.83713851 |  |
| albert-base-v2 | 32 | 3 | 3.00E-05 | 1.00E-08 |  | 0.01 | 0.9251348149 | 54.42389457 | 0.9430917228 | 51.98035208 | 51.63622527 | 0.1 |
| albert-base-v2 | 32 | 4 | 3.00E-05 | 1.00E-08 |  | 0.01 | 0.7931970478 | 67.60841837 | 0.9970703101 | 54.06294449 | 50.41856925 | 0.1 |
| albert-base-v2 | 16 | 3 | 3.00E-05 | 1.00E-08 |  | 0.01 | 0.9442128167 | 51.98679728 | 0.9520759715 | 51.91441442 | 50.0761035 | 0.1 |
| albert-base-v2 | 16 | 3 | 1.00E-05 | 1.00E-08 |  | 0.01 | 0.8710818268 | 62.6362862 | 0.9773926723 | 50.02815317 | 49.6194825 | 0.1 |
| electra | 32 | 3 | 3.00E-05 | 1.00E-08 |  | 0.01 | 0.9156131631 | 55.71853741 | 0.9853870836 | 46.29045165 | 50.64687976 | 0.1 |
| **albert-base-v2** | **32** | **4** | **3.00E-05** | **1.00E-08** |  | **0.01** | **0.9587668154** | **46.11607143** | **0.960945814** | **43.51662518** | **51.78843227** | **0.1** |

The 2nd log:

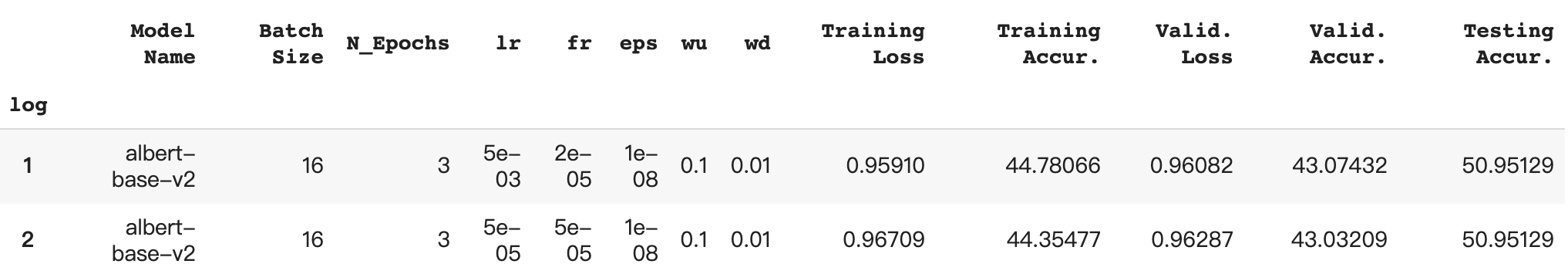
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Model Name | Batch Size | N\_Epochs | lr | fr | eps | wu | wd | Training Loss | Training Accur. | Valid. Loss | Valid. Accur. | Testing Accur. |
| xlnet | 32 | 4 | 3.00E-05 | none | 1.00E-08 | 0.1 | 0.01 | 0.4545018434 | 83.26955782 | 1.459015717 | 43.38993599 | 48.89649924 |
| xlnet | 32 | 2 | 8.00E-03 | 3.00E-05 | 1.00E-06 | 0.3 | 0.01 | 1.041007528 | 43.85841838 | 0.9648312193 | 43.24768846 | 50.95129376 |
| xlnet | 32 | 2 | 5.00E-03 | 2.00E-05 | 1.00E-08 | 0.1 | 0.01 | 1.007680466 | 44.6237245 | 0.9634933729 | 44.73906474 | 48.89649924 |
| albert-base-v2 | 32 | 2 | 5.00E-03 | 2.00E-05 | 1.00E-08 | 0.1 | 0.01 | 1.013667674 | 44.53869048 | 0.9644417126 | 43.24768846 | 50.95129376 |
| albert-base-v2 | 32 | 2 | 5.00E-03 | 2.00E-05 | 1.00E-08 | 0.1 | 0.01 | 0.9626696132 | 44.72151361 | 0.9640179244 | 43.24768846 | 50.95129376 |
| albert-base-v2 | 32 | 3 | 5.00E-03 | 2.00E-05 | 1.00E-08 | 0.1 | 0.01 | 0.9720912097 | 47.04719389 | 0.9662280614 | 43.24768846 | 50.95129376 |
| albert-base-v2 | 32 | 4 | 3.00E-05 | none | 1.00E-08 | 0.1 | 0.01 | 0.8882992945 | 57.72534014 | 0.9931740898 | 43.24768846 | 50.95129376 |
| albert-base-v2 | 32 | 2 | 3.00E-05 | none | 1.00E-08 | 0.1 | 0.01 | 0.9605611593 | 48.11437075 | 0.9521363063 | 51.54027383 | 50.41856925 |
| albert-base-v2 | 32 | 2 | 3.00E-05 | none | 1.00E-08 | none | 0.01 | 0.96823891 | 44.85331633 | 0.9655782927 | 43.24768846 | 50.91324201 |
| albert-base-v2 | 32 | 2 | 3.00E-05 | none | 1.00E-08 | 0.3 | 0.01 | 0.9561010386 | 47.67431974 | 0.9605669492 | 47.07725817 | 50.87519026 |
| albert-base-v2 | 32 | 3 | 3.00E-05 | none | 1.00E-08 | 0.1 | 0.01 | 0.9421355432 | 50.55484695 | 0.9653435607 | 46.58828233 | 50.87519026 |
| roberta | 32 | 3 | 3.00E-05 | none | 1.00E-08 | 0.1 | 0.01 | 0.9555197334 | 47.1492347 | 0.9625042894 | 46.23488623 | 48.85844749 |
| roberta | 32 | 3 | 3.00E-05 | none | none | 0.06 | 0.01 | 0.9589914738 | 45.75467687 | 0.9630940758 | 45.79703059 | 49.04870624 |
| roberta | 16 | 4 | 1.00E-05 | none | none | 0.06 | 0.1 | 0.8042444122 | 62.08475298 | 1.023312278 | 46.59346848 | 49.543379 |
| bert-base-uncased | 32 | 4 | 1.00E-05 | none | none | 0.06 | 0.1 | 0.5708310699 | 80.08078231 | 1.118884071 | 48.38415719 | 51.56012177 |
| bert-base-uncased | 16 | 4 | 1.00E-05 | none | none | 0.06 | 0.1 | 0.3808206013 | 86.85051107 | 1.378147476 | 48.60641892 | 50.2283105 |
| bert-base-uncased | 32 | 4 | 1.00E-05 | none | none | 0.01 | 0.1 | 0.5582021983 | 81.05867347 | 1.135324642 | 49.24431008 | 50.68493151 |
| bert-base-uncased | 32 | 4 | 5.00E-06 | none | none | 0.01 | 0.1 | 0.8313606623 | 64.90221088 | 0.996364724 | 49.23097439 | 49.88584475 |
| bert-base-uncased | 32 | 4 | 1.00E-05 | none | none | 0.06 | 0.1 | 0.5205045101 | 83.46088435 | 1.132244302 | 47.69736843 | 50.68493151 |

Funniness\_Estimation\_4th\_version:

Fake task log:



Real task log:



In the ‘Funniness\_Estimation\_3rd\_version’, the control variable method is applied when tuning the parameters. The best performance is achieved when using an albert-base-v2 model with the hyperparameters displayed on the above “The 1st log” table which are marked by red. For the different preprocessing methods, the cut-headlines version and the punctuation removal version have the same performance as the normal one except that doing cut-headlines will reduce the training time for a single epoch. One last thing we need to pay attention to is that no matter which one of these models is chosen to implement and whatever the parameters settings are, the test accuracy is just in the range of about 45% to 50%, which probably means that the ways I choose to tackle this domain of task is inappropriate.

In the ‘Funniness\_Estimation\_4th\_version’, when executing the training for the real task, the loss for the training dataset and the loss for the validation dataset remain almost unchanged throughout the epochs even for the different hyperparameters settings. There are two possible reasons for this, one is that there is something wrong with my code, another is the fake task just brings the model into the area where local minima is just nearby.