Project 5 (Sequence and Sentiment Classification using Transformers)

Part 1 - Named Entity Recognition using BERT

1.1 When initializing the BertForTokenClassification-class with BERT-base you should get a warning message. Explain why you get this message.

> *'Error: Some weights of the model checkpoint at bert-base-german-cased were not used when initializing BertForTokenClassification:*
> *This IS expected if you are initializing BertForTokenClassification from the checkpoint of a model trained on another task or with another architecture*
> *(e.g. initializing a BertForSequenceClassification model from a BertForPreTraining model).*
> *This IS NOT expected if you are initializing BertForTokenClassification from the checkpoint of a model that you expect to be exactly identical*
> *(initializing a BertForSequenceClassification model from a BertForSequenceClassification model).*
> *Some weights of BertForTokenClassification were not initialized from the model checkpoint at bert-base-german-cased and are newly initialized:*
> *['classifier.bias', 'classifier.weight'].*
> *You should probably TRAIN this model on a down-stream task to be able to use it for predictions and inference.'*

We added the received error message here in order to make clear what we are talking about.
This error message is received because the BERT model 'bert-base-german-cased'
that we want to use for our BertForSequenceClassification model has been pre-trained on another architecture (BertForPreTraining). This means that the layers of our target architecture (BertForSequenceClassification), which are not present in our base architecture (BertForPreTraining) will be randomly initialized. This layers of our base model (BertForPreTraining), which are not part of our target architecture (BertForSequenceClassification) will be discarded. This means that the model has to be fine tuned on a down stream task in order to learn the proper weights in the randomly intialized layers.

1.2 Which model performed best on the evaluation set?.

Description	Approx Micro F1	Approx Macro F1
Model fine-tuned with 1000 sentences (and non-frozen embeddings)	0.9064	0.3462
Model with 3000 sentences (and non-frozen embeddings)	0.9301	0.4655
Model with 3000 sentences (and frozen embeddings)	0.9047	0.1903

=> We can see that the model being fine tuned using 3000 sentences with non frozen base model parameters achieves the best performance.

1.3 Are there differences between f1-micro and f1-macro score? If so, why?.
There are big differences between micro and macro f1. The macro f1 metric is computed by independently computing the f1 score for each class/ label seperately and then taking the average (In this case all classes are weighted equally). The micro f1 is computed by weighting the f1 score of every class by the number of samples supporting this class (In this case the weight of a class is poportional to its number of samples). The big negative deviation between micro and macro f1 just says that the model is performing well for frequent classes of labels while performing worse for at least one minority class. For example it might able to correctly label people with 'PER' and others with 'O' but it might be bad in correctly labeling 'ORG'.
If the frequency of 'PER' and 'O' is much higher than the 'ORG' label entity, this leads to a deviation in micro and macro f1 (Like we observerd).

1.4 How large is the performance gap between 1'000 and 3'000 sentences for finetuning?.
There is a performance gap between a BERT model being fine tuned on 1000 and 3000 sentences of approx 3% micro F1 and approx 12% macro F1. This performance difference is quite significant. Therefore more fine tuning sampels certainly improved the performance. It would be interesting how far you can increase the models performance by increasing the amount of data used for fine tuning.

1.5 Is it better to freeze or not to freeze the embeddings?
Since the performance of the model being trained on 3000 sentences with non frozen embeddings exceeds the performance of the reference model, it is better to not freeze the embeddings. This makes sense since the warning from 1.1 basically notifies us that the base model needs to be fine tuned at first before using it. However, if the weights of the base model are frozen then this fine tuning only happens in the final layer. Therefore, for this task and base model it is better not to freeze the embeddings. However, this depends on the specific base model and scenario.

Part 2 - Resource Limited Competition: Sentiment Analysis

In Part 2, we are expected to submit a fine-tuned Transformer model for the IMDB Binary sentiment classification task.

2.1 Please have a statement in your report as to why and what hypothesis led you to choose this architecture. Did your results support the hypothesis? Why/Why not?

We chose and experimented with 5 different pre-trained models.
The reasons why we chose them are listed below.
The following tables showes their performances before fine-tuning.

model_name	accuracy	f1_macro	f1_micro
distilbert-base-uncased-finetuned-sst-2-english	0.8075	0.806671	0.8075
echarlaix/bert-base-uncased-sst2-acc91.1-d37-hybrid	0.7055	0.686775	0.7055
gchhablani/bert-base-cased-finetuned-sst2	0.8220	0.821986	0.8220
roberta-base	0.5000	0.333333	0.5000
siebert/sentiment-roberta-large-english	0.8880	0.887986	0.8880

Model 1:
The "distilbert-base-uncased-finetuned-sst-2-english", is based on the DistilBERT base model, which is the distilled version of the BERT base model and is later fine-tunned one the Stanford Sentiment Treebank(SST). The Stanford Sentiment Treebank consists of sentences from movie reviews and human annotations of their sentiment.
According to this paper(https://arxiv.org/abs/1910.01108), compared to the original BERT model, the distilled version pre-trains a smaller general-purpose language model and is able to reduce the size of a BERT model by 40%. Since we are doing a comparatively simple binary classification, we thought this lightweight and efficient version could fit our task well.

Model 2:
Is called "echarlaix/bert-base-uncased-sst2-acc91.1-d37-hybrid". We think this model is interesting because it uses a pruning method. Some attentions heads are removed. There are pros and cons regarding this methods. It can avoid over-fitting but potentially lowers the accuracy, which is proven in the later result.

Model 3:.
Is called "gchhablani/bert-base-cased-finetuned-sst2". This model is a fine-tuned version of bert-base-cased on the GLUE SST2 dataset. We included it because it achieved a high accuracy of 0.92 on the sst dataset.

Model 4:
The "roberta-base", is introduced in this paper: https://arxiv.org/pdf/1907.11692.pdf.
According to this paper, this model has improved the BERT in the following 4 aspects: (1) training the model longer, with bigger batches, over more data; (2) removing the next sentence prediction objective; (3) training on longer sequences; and (4) dynamically changing the masking pattern applied to the training data. They also collect a large new dataset (CC-NEWS) of comparable size to other privately used datasets, to better control for training set size effects.

Model 5:
is called “siebert/sentiment-roberta-large-english”. This model is a fine-tuned checkpoint of RoBERTa-large (Model 4). It enables reliable binary sentiment analysis for a larger range of types of English-language text. This model is already fine-tuned and has a high accuracy before training. However, since it is large and it is the fifth model. Our GPU is out of memory in the end.

The models performance after fine-tuning is shown in the table below.

model_name	accuracy	f1_macro	f1_micro
distilbert-base-uncased-finetuned-sst-2-english	0.8385	0.838466	0.8385
echarlaix/bert-base-uncased-sst2-acc91.1-d37-hybrid	0.8425	0.842398	0.8425
gchhablani/bert-base-cased-finetuned-sst2	0.8555	0.855461	0.8555
roberta-base	0.8825	0.882499	0.8825

Conclusion

Besides the model 5, “siebert/sentiment-roberta-large-english”, which is fine-tuned on a large dataset, we found that the prediction accuracy of Model 4, the “roberta-base”, increased the most and achieved the best performance (micro and macro f1 approx 0.88). Compared to BERT, which is the base model of our models 1, 2 and 3, the RoBERTa is pre-trained longer and with bigger batches and more data and longer sequence. We can’t tell if them removing the next sentence prediction objective significantly enhanced the model. The use of a dynamical masking pattern definitely helps.