Bastien LE CHENADEC
bastien.le-chenadec@eleves.enpc.fr

Lab session # 4
ALTEGRAD 2023

11/14/23

1 Question 1

We perform the same investigation as in the question 3 of the previous lab. We don't count bias and layer norm parameters, nor do we count the parameters of the model head. The only additional parameters compared to the transformer in the previous lab are $(n_{\text{source\_positions}} + 2) \times d_{\text{model}}$ positional encoding parameters. The theoretical number of parameters is thus :

$$d_{\text{vocab}} \times d_{\text{model}} + (n_{\text{source\_positions}} + 2) \times d_{\text{model}} + n_{\text{layers}} \times (2 \times d_{\text{ff}} \times d_{\text{model}} + 4 \times d_{\text{model}}^2) \tag{1}$$

In our case,

- $d_{\text{vocab}} = 32000$
- $d_{\text{model}} = 512$
- $d_{\text{ff}} = 512$
- $n_{\text{source\_positions}} = 256$
- $n_{\text{layers}} = 4$

Which yields $\boxed{22807552}$ parameters.

2 Accuracies on the Roberta model

2.1 Using fairseq

For each seed the best validation accuracy is reached after 5 epochs (1125 training steps). The corresponding test accuracies are as follows :

	Seed 0	Seed 1	Seed 2
Pretrained	79.9%	80%	83.5%
Random	67.5%	68.1%	67.4%

Which yields a mean test accuracy of 81.1% (± 0.02%) for the pretrained model and 67.7% (± 0.003%) for the randomly initialized model.

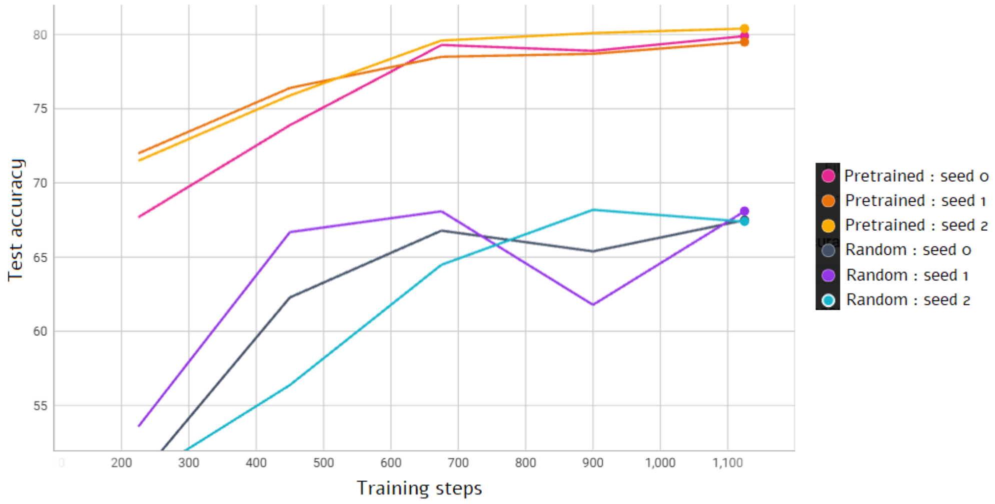
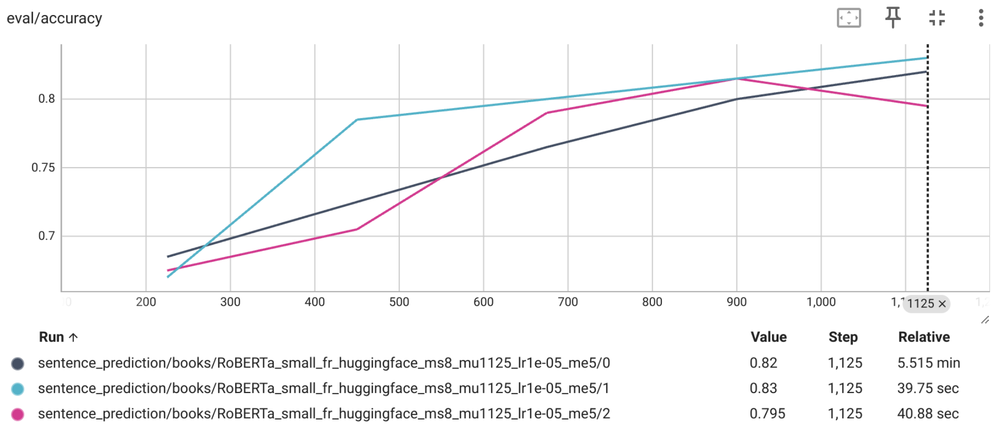


Figure 1: Test accuracy of the six models

As expected, the pretrained models perform much better than the randomly initialized ones.

2.2 Using HuggingFace



	Run ↑	Value	Step	Relative
●	sentence_prediction/books/RoBERTa_small_fr_huggingface_ms8_mu1125_lr1e-05_me5/0	0.82	1,125	5.515 min
●	sentence_prediction/books/RoBERTa_small_fr_huggingface_ms8_mu1125_lr1e-05_me5/1	0.83	1,125	39.75 sec
●	sentence_prediction/books/RoBERTa_small_fr_huggingface_ms8_mu1125_lr1e-05_me5/2	0.795	1,125	40.88 sec

Figure 2: Validation accuracy of three models

We get the following test accuracies on the three seeds:

- 79.9%
- 79%
- 81.1%

Which is consistent with the results we get using fairseq.

3 Question 2

The parameters used in LoraConfig are :

- **r** : Rank of the fined-tuned matrices A and B. We want it to be small to reduce the number of trainable parameters, but big enough that the model can still be fine-tuned efficiently.
- **lora_alpha** : A scaling factor applied to the matrix BA. In general, the real scaling factor is $\frac{\alpha}{r}$ so we want α to not depend on r. α is useful to avoid retuning hyperparameters when changing r. [1]
- **target_modules** : The layers we want to fine-tune. In our case, we fine-tune the attention heads.
- **lora_dropout** : The dropout rate applied to the fine-tuned matrices A and B. This is useful to avoid overfitting.
- **bias** : Type of bias applied to the matrices A and B. In our case we don't use any bias.
- **task_type** : The type of task to fine tune on. This parameter is inherited from the parent class PeftConfig. In our case, we use "CAUSAL_LM" which is sentence prediction from a prompt.

References

[1] Edward Hu, Yelong Shen, Philip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. 2021.