Matias Etcheverry
matias.etcheverry9@gmail.com

Lab session # 1
ALTEGRAD 2022

10/31/22

1 Question 1

The role of a transformer is to predict the next token from a set of token. Thus, it is really important that the transformer can't access future tokens to predict it. However, when we construct the datasets, we don't respect this constraint. Actually, the role of the square mask is to prevent the attention mechanism of the transformer to focus on the future tokens. The mask puts weights of -infinity to these tokens, so they can't be minimized.

With the current architecture, the transformer doesn't have access to the relative position of the tokens. To prevent this, we simply add a position encoding whose role is to give a piece of information on each token. This positional encoding relies on 4 principles:

- each word receives a unique encoding in a sentence
- the step between 2 words should always remains the same
- the positional encoding should adapt on any sentence of any length

2 Question 2

We are currently aiming at 2 differents tasks:

- language prediction where our transformer predicts for the next word
- classification where our transformer classifies a sentence in bad or good

We first need to train our transformer on the language prediction task for it to understand language before moving to classification. Thus, the classification head helps us to switch between the 2 cases. For every growing subset of tokens in the input,

- it will output a vector of length `number of token` when predicting the next token. The argmax of the output will be the predicted token
- it will output a vector of length 2 when predicting good or bad classification.

3 Question 3

Let's compute the number of trainable parameters of the body of the transformer.

- The encoder has `num_embeddings` $\times$ `embedding_dim` parameters, ie `ntokens` $\times$ `nhid` trainable parameters
- The positional embedding has 0 trainable parameters.
- The transformer encoder layer:
  - self attention mechanism of $3 \times$ `nhid` $\times$ `(nhid + 1)` parameters for projecting Q, V, K and `nhid` $\times$ `(nhid + 1)` parameters for projecting the result.
  - 2 linears layers of each `nhid` $\times$ `(nhid + 1)` parameters (weights and bias)
  - 2 normalization layers of each `nhid` $\times$ `(nhid + 1)` parameters (weights and bias)

Thus, the body of the transformer has `ntoken` $\times$ `nhid` $+ 8 \times$ `nhid` $\times$ `(nhid + 1)`.
The classification head has only a linear layer with `nhid` $\times$ `(nclasses)` trainable parameters.
In the end, we have:

- `ntokens` $\times$ `nhid` $+ 8 \times$ `nhid` $\times$ `(nhid + 1)` $+$ `nhid` $\times$ `(ntokens)` $= 20\ 332\ 000$ trainable parameters for the language learning task
- `ntokens` $\times$ `nhid` $+ 8 \times$ `nhid` $\times$ `(nhid + 1)` $+$ `nhid` $\times 2 = 10\ 332\ 200$ trainable parameters for the classification task

4 Question 4

Figure 1 shows the accuracy curves from 2 different models. The task for the 2 models is to predict wheither a sentence has a good or a bad tone. One model was trained from scratch while the other model was pretrained from another task.

The first thing we can notice is that the 2 models don't reach their maximum of accuracy within the same number of epochs. The models trained from scratch begins with a 50% accuracy because it has never seen text before. It has an exponential learning curve. The learning curve also flattens at 77% of accuracy from the 7^{th} epoch.
On the other hand, the pretrained model already starts with a high accuracy as it already understands text. The learning curve flattens at 80% of accuracy from the 5^{th} epoch.
In the end, the pretrained model converges faster towards a better model.

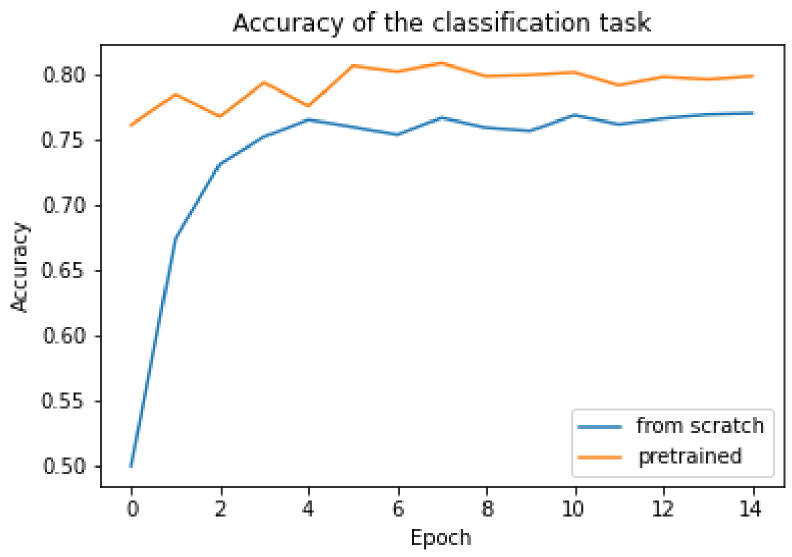

Figure 1: Accuracy of a pretrained model and a model trained from scratch on the first 15 epochs

5 Question 4

The main problem of our approach is that the model can only learn from left to right. Each piece of information can only be used for future understanding. This is due to the unilateral mask. However, in the article [1], the authors proposed a new mask, where all the context is used for prediction. This context comes from the right and left parts of the word to predict. This new mask enabled to reach new state-of-the-art performances on GLUE dataset for instance.
Figure 2 shows the different behaviors of the unilateral and bilateral masks. The model trained with the bilateral mask is therefore stronger in logic as he can modify its knowledge based on future.

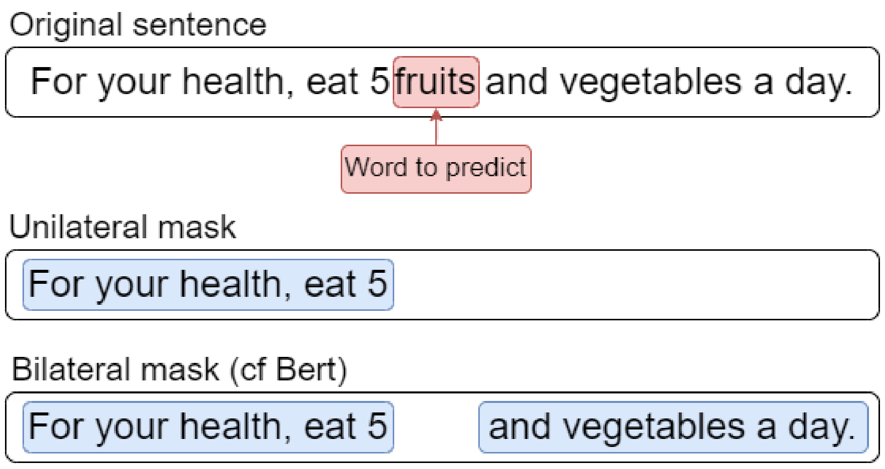

Figure 2: Unilateral and bilateral masks

References

[1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810s.04805, 2018.