Mercier Marine
marine.mer98@gmail.com

Lab session # 3
ALTEGRAD 2021

11/25/21

1 Question 1

A square attention mask define which tokens are visible. Since the self-attention layers in nn.TransformerEncoder are only allowed to look at previous words in the sentence, tokens in future position have to be hidden by a mask. PositionalEncoding gives information about relative or absolute position of tokens in the sentence. It helps the model to find the closest tokens and to give them more importance.

2 Question 2

The head of the model define it's task, therefore we have to replace the classification head in order to do language modeling. The goal of the algorithm change, in classification the model does sentiment analysis and classify the sentence accordingly, while in language modeling, the model is creating a sentence. The size of the output and target changes, in language modeling the target is a batch of sequence, while in classification it is a batch of labels, and only the vector representing the last token is given in the output.

3 Question 3

The classification head is a linear layer with 200 inputs and 2 outputs, therefore the number of trainable parameters of the classification head is equal to (number of inputs + 1) time the number of outputs. Therefore the classifier has $(nhid + 1) * 2 = 402$ trainable parameters (we have to add the number of output one extra time for biases). The head for the language modeling task is a linear layer with 200 inputs and 50 001 outputs, therefore this head has 10 050 201 trainable parameters.

The model base has 10 968 200 trainable parameters (cf. code). The encoder has $ntokens * nhids = 10000200$ trainable parameters. There are $nlayers$ TransformerEncoderLayer. Each TransformerEncoderLayer is composed of :

- A multihead with $nhead * nhid * (nhid + 1) * 2$ trainable parameters.
- Two normalization layers with $2 * nhid$ trainable parameters each.
- Two linear layers with $nhid * (nhid + 1)$ trainable parameters each.

Therefore the TransformerEncoderLayers have a total of $nlayers * nhid * ((nhid + 1) * (2 + nhead * 2) + 4) = 10968200$ trainable parameters.

4 Question 4

We can see that the validation accuracy of the pretrained model is high, even at the first epoch while the accuracy of the model trained from scratch is starting from zero. The scores of the pretrained model are better than those of the model trained from scartch. Indeed the pretrained model has been trained on more data.

5 Question 5

The language modeling objective used in this notebook is limited by the fact that future tokens are masked. The solution in [1] uses representation coming from the left and right context, unlike our left-to-right model.

References

[1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.