GALAGAIN Calvin
calvin.galagain@gmail.com





# 1 Question 1

To count the parameters, we will look at what the base_architecture() function creates based on the BERT [1] model.
To start, we have an Embedding block that will first tokenize our data and then pass through a positional Encoding layer. Which gives $n_{token} \times n_{hidden} + n_{positional} \times n_{hidden}$ parameters.

To calculate the parameters of the Encoder and Decoder blocks, we will look at each sub-block:

- An "MHA" block needs to do 4 linear calculations. 3 are simple projections and 1 is a concatenation. Considering the biases, we have here $4 \times (n_{hidden}^2 + n_{hidden})$ parameters.
- A "Feed Forward" block is a network that has two linear layers matrix transformations. Considering the biases, we have here $2 \times (n_{hidden}^2 + n_{hidden})$ parameters.

We have 4 layers of Transformers which brings a total result of: $4 \times (6 \times (n_{hidden}^2 + n_{hidden})) + n_{token} \times n_{hidden} + n_{positional} \times n_{hidden}$
With $n_{hidden} = 512$, $n_{token} = 32000$ and $n_{positional} = 258$, we have $24 \times (512 \times 512 + 512) + 32000 \times 512 + 258 \times 512 = 22819840$ parameters.

# 2 Question 2

We used a $RoBERTa_{small}^{fr}$ model with the two libraries. The use of this model is thus brought differently. Fairseq [2] offers many translation and language scripts for custom training. And Huggingface [3] seeks to offer models based on pre-trained transformers or specialized models for research and real-world problems.
Our model was brought in two different ways. With Fairseq, we have tokenize and binarize the data so that the model can understand what we want from it. With Hugging Face, everything was automatic. What makes me lean in favor of Hugging Face, is the fact that with an automatic preprocessing, we cannot make mistakes either on the value or on the output format of our processing.
The advantage of Hugging Face is to be able to put the raw .json as input and the results are optimal.

# References

[1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018.

[2] Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. fairseq: A fast, extensible toolkit for sequence modeling. *CoRR*, abs/1904.01038, 2019.

[3] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. Huggingface's transformers: State-of-the-art natural language processing. *CoRR*, abs/1910.03771, 2019.