пощупать много токенайзеров, поискать статьи по токенизации (их эффективности). К следующей неделе небольшое literature review нужно предоставить

Tokenizers

There are 3 main types of tokenizers:

- **Character level**
  - **+ Very small vocabulary**
  - **+ No out of vocabulary**
  - **- What is the meaning of a character? No meaning of words**
  - **- Tokenized sequences are very long (1 vs n tokens)**
- **Subword level**
  - **+ Seems the most appropriate. Handles out of vocabulary, splits words,**
  - **+ OK vocabulary size**
  - **+ Mitigates data sparsity**
  - **+ SOTA models based on this (GPT)**
  - **- expensive to train**

- **Word level**
  - **- Out of vocabulary words**
  - **- Different meaning of same words (boy, boys)**
  - **+ simple and fast**
  - **+ Easy to customize**

Also there are some statistical or dictionary based tokenizers, but I just consider it as a word level tokenizers: https://iq.opengenus.org/tokenization-in-nlp/

Examples of popular subword tokenizers are

- **WordPiece**
- **BPE**
- **Byte-Level BPE (slight modification to the BPE, allows to avoid using of <unk> token)**

Fast vocabulary transfer paper: Interesting approach. It should be useful for the stage of COMBINING the tokenizer results together. Additionally, this FVT is used in some context, area. I mean that this approach is tested in some concrete, not deep areas. We want to build something more general, but, of course, this FVT should be tested in my work.

Another problem that I investigate is that outcomes of the tokenizer will be of different shapes. It strongly depends on the tokenizer and I need to create some solution for this.

Yet another problem is different sizes of the same text tokenized in different languages (cost of training, size of tokenized text (up to 15 times for the same text).

This is the issue with multilingual LLMS, so, we need to focus maybe on single language model or find a way how to deal with that problem (do not found yet).

As for me, subword tokenizers seem to be the most appropriate for our task. I see that subword tokenizers are used for training almost every SOTA models (GPT, BERT). I want to investigate further what may be the outcome of combining such tokenizers together, how this can influence the performance of the model.

Overall, this is all that I investigate during the first week and I am ready to work with some new tasks.