Praktikum 1 – Report

Objective: Find a way to improve the existing project with focus on LLM and or PCMs.
The current project encompasses:

- Tokenization
- Lemmatization
- Part-of-Speech Tagging
- Named Entity Recognition
- Dependency Parsing
- Translation into English
- Transliteration between Latin and Cyrilic
- Linking Local and online Definition of the words
- Visualization of Dependency Tree

Using classla pipeline for Serbian language.

Research for potential upgrades/additions

YugoGPT:

- Online web-app does not work.
- Initial attempts to make it work locally failed, because my GPU is not strong enough to run the model.
- Successfully installed a weaker model of YugoGPT called YugoGPT-Q4_K_M-GGUF. Although it works, my initial tests have shown poor performance. The model sometimes repeats the questions multiple times as an answer or gives me an answer completely out of context. However, it seems that this can improved by playing around with parameters.
- Successfully installed YugoGPT on Google Colab, however, I do not see any difference between the “weaker” model and this one. Maybe I am missing something?
- I could potentially implement this model in my web application. For example, it could be used for providing the same outputs like classla pipeline, which could be used for comparisons. Additionally, I could implement error detector, sentence generator or a chatbot.

- Idea – Why force YugoGPT when I could use lightweight version of Mistral, Gemma or Llama 3? Requires additional research and testing. Potentially also requires fine-tuning.

Speech-to-Text Integration

- I could add an additional feature in my app to directly input the text by using the microphone of the device.
- Realized by using Vosk API for Serbian Language OR whisper.cpp
- Add confidence score

Improved Named Entity Linking

- Current project link local and online definitions of the words. I could additional also link Wikipedia pages.
- DBpedia Spotlight could help me with automatic Wikipedia linking
- Maybe display tooltips with Wikipedia summaries when an entity is detected?

Text Summarization

- There is a possibility for summarizing Serbian texts using sumy python library. I tested it and it works.
- According to ChatGPT I could also use BERTSUM or PEGASUS for summarization, however, it would be much better if the model was fine-tuned for Serbian language.

BERTić

- Transformer-based language model designed for Serbian/Croatian/Bosnian/Montenegrin.
- Can be used for the same tasks that my current project does, however, it is claimed that this model has a higher accuracy than Classla.
- It could be used for POS tagging, NER, **Geolocation Prediction**, **Commonsense Casual Reasoning.**
- Could be used to make comparisons between Classla and itself
- Maybe also a possibility to use it as a chatbot?
- Geolocation Prediction could be used as an addition to my webapp

Sentiment Analysis

- Used to determine the emotional tone of the input

- There seems to be multiple fine-tuned BERT models for Serbian, like setu4993/LaBSE and EMBEDDIA/crosloengual-bert, which could be used for this.
- There is also an option for Rule-based sentiment analysis, like textblob-sr

Additional Visualizations

- Named Entity Recognition (NER) Heatmap (Highlights people, locations, organizations)
- Part-of-Speech (POS) Sunburst Chart (To see the distribution)
- Word Cloud of Most Common Words (Quickly visualizes frequent words)
- Interactive NER Entity Timeline (Good for stories and fairy tales)
- Dependency Graph Evolution (How structure changes when translating between languages)