Bachelor Thesis – Preparation

Ivan Cvetanovic

Python Libraries for NLP for Serbian language:

a) SrbAI - https://github.com/Serbian-AI-Society/SrbAI
Transliteration (between cyrilic and latin), Stemmer, Spell checker, Part-of-speech tagging, negation-handling. I could not find what corpora it uses, but I suspect that it is the same corpora from classla, because I tested it, and it works for wide range of words.

b) Classla - https://pypi.org/project/classla/
This library is partially used in SrbAI for POS tagging, and offers functions like: Tokenization and sentence splitting, Lemmatization, Dependency Parsing and Named entity recognition. Uses CLASSLA-web.sr Copus, which uses data found online, like news, websites etc.

c) Stanza StanfordNLP Python Package - https://stanfordnlp.github.io/stanza/
Production of constituency parsed trees, like those found in the Penn Treebank. This can give us a hierarchical, tree-like representation of the sentence's grammatical structure.

Additional Corpora: multext_east - consists of POS-tagged version of George Orwell's book 1984 in Serbian; SETimes-SR.

Based on the findings above, here is an idea about how the milestones could look like:

Milestone 1

- Website with a simple design and one entry field in the middle.
- Entry field supports the input of a single word.
- Entry field supports both Latin and Cyrillic input. If input is in latin, a cyrilic text will also appear under the input field. Example:

```
[ ]  from srbai.Alati.Transliterator import transliterate_cir2lat, transliterate_lat2cir

     lat = transliterate_cir2lat("Текст на ћирилици. ")
     print(lat)
     cir = transliterate_lat2cir("Tekst na latinici. ")
     print(cir)

     Tekst na ćirilici.
     Текст на латиници.
```

- Upon entry, the word is checked for spelling. If the word is wrong, a suggestion will appear under the input field, otherwise it continues with the output. Example:

```
[ ]  from srbai.SintaktickiOperatori.spellcheck import SpellCheck
     sc = SpellCheck('sr-latin') #postoji opcija i #sr-cyrilic za ćirilicu /usr/local/lib/python3.9/dist-packages/srbai/Resursi/Recnici/Serbian (Latin).dic
     word = "predetori"
     correction = sc.spellcheck(word)
     if correction:
         print(f"Did you mean '{correction}'?")
     else:
         print("No close match found.")

     Did you mean 'predatori'?
```

- Stem of the word is output. Example for the whole sentence input:

```
[ ]  from srbai.SintaktickiOperatori.stemmer_nm import stem_str, stem_arr

     sent = stem_str("Jovica je išao u školu. Marija je dobra devojka.")
     print(sent)

     [nltk_data] Downloading package punkt to /root/nltk_data...
     [nltk_data]   Unzipping tokenizers/punkt.zip.
      jovi jesam isx u sxkol . marij jesam dobr devoj .
```

- Find if the word is a noun, verb, adjective, etc. Example:

```
[ ]  from srbai.SintaktickiOperatori.POS_tagger import POS_Tagger

     pt = POS_Tagger()
     tags = pt.tag('Jovica je išao u školu. Marija je dobra devojka.')
     print(tags)

     [nltk_data] Downloading package punkt to /root/nltk_data...
     [nltk_data]   Unzipping tokenizers/punkt.zip.
     [('Jovica', b'N-msn'), ('je', b'Vcr3s'), ('išao', b'Vmp-sm'), ('u', b'Sa'), ('školu', b'N-fsa'), ('.', b'Z'), ('Marija', b'N-fsn'), ('je', b'Vcr3s'), ('dobra', b'Agpfsn'), ('devojka', b'N-fsn'), ('.', b'Z')]
```

- Here is a table provided by SrbAi for the above output:
  - * N - Noun
  - * V - Verb
  - * A - Adjective
  - * P - Pronouns
  - * S - Suggestions

    - * Z - Punctuation mark
    - * C - Conjunction
    - * M - Number
    - * R - Attachments,
    - * I - Exclamations
    - * Q - Words
    - * Y - Abbreviations
    - * X - Other

- I could insert the legend above into the simple website and then use these abbreviations in case I do not have enough space to write full words.

- Part-of-speech Tagging for the word. Example for the whole sentence:

```
from srbai.NER.NER_classla import NER_classla

ner = NER_classla()
ners = ner.perform_NER("Jovica je išao u školu u Beogradu")
print(ners)
```

```
INFO:classla:Use device: cpu
INFO:classla:Loading: tokenize
INFO:classla:Loading: pos
INFO:classla:Loading: lemma
INFO:classla:Loading: depparse
INFO:classla:Loading: ner
INFO:classla:Done loading processors!
[{'text': 'Jovica', 'NER': 'B-PER', 'dep_rel': 'nsubj'}, {'text': 'je', 'NER': 'O', 'dep_rel': 'aux'},
```

- Output is happening in a small table, where there will be two rows and multiple columns. It will not look like a simple text in the above examples.

Milestone 2

- Enable input of whole sentences.
- Spell checker can now give multiple suggestions of incorrect words.
- Enabling the output of information of multiple words; one word in each row in the table.

- Add Dependency Tree with simple textual representation with the possibility to make a nicer graphical representation. Example:

```
nlp = stanza.Pipeline(lang='sr')
doc = nlp("Мачка седи на столици.")
print(doc.sentences[0].print_dependencies())

```

```
('Мачка', 0, 'root')
('седи', 1, 'flat')
('на', 1, 'flat')
('столици.', 1, 'flat')
None
```

Milestone 3

- Improve GUI of the website.
- Enable Profanity Detection (?) -> I could not find anything for Serbian language, I would need to do this manually by creating profanity lexicon.
- Decide the structure of the thesis
- Check for additional literature according to the structure.
- Write the Thesis

Based on the information above, here are the potential names for my Bachelor's Thesis:

- A Web-Based Morphological Analyzer for Serbian
- Implementation of a Web-Based System for Morphological Analysis of Serbian
- Serbian Morphological Analysis and Visualization: A Web-Based Approach with NLP

Here is some literature, which may prove useful for my thesis:

- A Survey of Resources and Methods for Natural Language Processing of Serbian Language - https://arxiv.org/pdf/2304.05468
- An overview of resources and basic tools for the processing of Serbian written texts - https://www.researchgate.net/publication/228523124_An_overview_of_resources_and_basic_tools_for_the_processing_of_Serbian_written_texts
- Resources and Methods for Named Entity Recognition in Serbia - https://infoteka.bg.ac.rs/ojs/index.php/Infoteka/article/view/509
- The Serbian language in the digital age - https://link.springer.com/book/10.1007/978-3-642-30755-3
- Corpora issues in validation of Serbian WordNet. - https://link.springer.com/chapter/10.1007/978-3-540-39398-6_19
- Composite tense recognition and tagging in Serbian. - https://aclanthology.org/W03-2908.pdf
- WaC – web corpora of Bosnian, Croatian and Serbian - https://aclanthology.org/W14-0405.pdf
- Lemmatization and morphosyntactic tagging of Croatian and Serbian - https://aclanthology.org/W13-2408.pdf