NLP Course. Assignment 3

Russian Nested Named Entities

Report

Name: Adelina Kildeeva

Email: a.kildeeva@innopolis.university

CodaLab nickname: adelyumi

Link on the GitHub repository: https://github.com/adelyumi/NLP_assignments

Solution Finding

I used various online resources to find a solution. Firstly, I studied the topic of NER and how it can be solved in different ways. While searching for information, I came across the Spacy library, which provides many possibilities in the field of language processing.

First (Baseline) Solution

I decided to download the Russian model "ru_core_news_lg" and immediately apply it to the test data.

The big limitation of this model is that it defines only 3 types of entities: person (PER), organization (ORG) and location (LOC). Nevertheless, I decided to see how the model works and see what percentage of entities it covers.

I applied the model to each text and changed the type labels to those given in the dataset (PER -> PERSON etc.). The final score obtained by this solution is **0.06**. It's not 0, but the solution is not flexible at all, so I decided to train a custom model.

Second (Main) Solution

To implement the second solution, I used the following tutorial: https://www.newscatcherapi.com/blog/train-custom-named-entity-recognition-ner-model-with-spacy-v3, mainly to preprocess the data for spacy models and use configurations.

First, I preprocessed the data. A special DocBin format is used for spacy models. Each named entity was processed with char_span method and was skipped if it is None or starts or ends with a space (since spacy models are sensitive to spaces and will generate errors if they are present).

Next, I downloaded the configuration file from the configuration generator on the official spacy website. The Transformer model does not support Russian, so I used the word2vec model.

```
python -m spacy init fill-config base_config.cfg config.cfg
```

And one more command to train the model (I used GPU on Google Colab).

```
python -m spacy train config.cfg --output ./ --paths.train ./training_data.spacy --paths.dev ./training_data.spacy --gpu-id=0
```

The training process lasted 50 epochs and about 40 minutes. Next, I paused the training.

Finally, I used the best model to predict named entities and recorded the results in the needed format.

This solution scored **0.52** in CodaLab.

Some ideas for improvement of this solution:

- use some pre-computed vectors (e.g. ru_core_news_lg)
- split the training dataset into train and eval (in the tutorial the model was trained without splitting, but I think it can improve the model performance)

Conclusions

I tried using the spacy library to recognize nested named entities in Russian. Pre-trained (default) models are ill-suited for this problem because they are trained for very general purposes. But with the help of the library we can quickly and easily train our own model, which shows decent results.

Notebooks with code for each solution and all necessary files can be found in the github repository.