# NLP. Assignment 3. Report

*Darya Zhuravleva*
*d.zhuravleva@innopolis.university*
*CodaLab: zhur_a_vleva*
*GitHub: link*

**NB**: the assignment was solved in Colab, so all the paths are to my Google Drive and need to be changed if you want to check code

## Solution 1. Spacy (folder solution_spacy)

This method finds and processes named entities in a text corpus by utilizing Spacy, a potent natural language processing package. The solution involves several key steps:

- To start, use the *read_file* method to load the training data from a JSONL file. After reading every line in the file and parsing it as JSON, this method outputs a list of dictionaries, each of which represents a text entry along with its corresponding named entities

- To deal with overlapping entities, the *update_entities* method eliminates entities that have already been processed. Using the start and end indices of each entity as a guide, this function iterates across the collection looking for overlaps. In order to prevent future overlaps, entities that do not overlap with any previously processed entity are added to a new list and have their indices marked as processed

- To prepare the data for training, each text entry's NERs are sorted according to length before being formatted so that Spacy can use them. Spacy's *ru_core_news_lg* model, which is tailored for Russian text, is used to train the model. Iterating through several epochs to enhance the model's performance, the training procedure entails updating the model with examples taken from the prepared data.

- Using a test dataset, the model is trained to identify named entities. The model's predictions are filtered to only contain entities that are included in the training set after the test data is handled in a manner akin to that of the training data. This guarantees that the output of the model is consistent and pertinent to the training set of data

- A list of dictionaries containing the identified entities and their matching IDs is prepared and saved to a JSONL file. The processed entities for every text entry in the test dataset are included in this file, which functions as the final output

## Solution 2. Dictionary (folder solution_dict)

Using a dictionary-based method to handle overlapping entities and combine them depending on predetermined constraints, the solution is made to analyze and recognize named entities inside text input. The solution involves several key steps:

- The data is loaded using the *read_file function*, and then the *count_ners* function is used to process the data in order to identify and count named entities

- The most common NER for each word is found by running the *most common ner* function over the dictionary. The text is divided into tokens using the *tokenize* function, and these tokens are then compared to the dictionary to find possible entities

- Overlapped or adjacent entities of the same kind are merged by processing the identified entities. To do this, the tokenized text is iterated through and the positions of the entities are compared to see if they should be classified as independent entities or combined

- The last stage entails putting the combined entities and their textual locations in an organized manner. To accomplish this, a list of dictionaries with the combined NER predictions and the matching ID from the original data are created

- The processed data is written to a JSONL file, which is subsequently compressed for storage and ease of access

## Best Solution

| | **Spacy** | **Dictionary** |
|---|---|---|
| **F1 score** | 0.32 | 0.26 |