Assignment IV: Argument Generation

Installing Dependencies:

- Assuming you have a working ``Python`` library and a ``pip`` tool:

* Install `jupyter` using the command: ``pip install jupyterlab``
* Install `pandas` using the command: ``pip install pandas``
* Install `numpy` using the command: ``pip install numpy``
* Install `nltk` using the command: ``pip install nltk``

Input and Output files:

To run the project please run the file: argument_generation.ipynb
Input file name should be: valid_data.json located in the same path as the notebook.
The output data will be named: predicted.json located in the same path as the notebook.

Goal of the Assignment:

The goal of this assignment is to generate a text for a set of given arguments, which represents the conclusion of the argument.

The output file should have the following format:

```
{
"<id_1>": “conclusion 1”,
"<id_2>": “conclusion 2”,
"<id_3>": “conclusion 3”
}
```

Features

TF-IDF:

We calculated TF-IDF by calculating tf score and idf score, and then multiplying them.

By using TF-IDF we are taking into account semantic features of the words in each sentence, because TF-IDF actually tells us how important each sentence is based on the importance of its words.

Claim Lexicon:

We search in each sentence of the argument and for how many words that exists also in the lexicon, we add a value to the sentence.

The idea behind this decision is that the conclusion of an argument should be an argumentative sentence.

We found the used lexicon from: https://github.com/webis-de/sigir20-extractive-snippet-generation-for-arguments from the paper “Extractive Snippet Generation for Arguments” (Alshomary et al.)

Strategy

We chose an unsupervised, extractive method, in which we identify and extract the **two most important sentences** from the text and form a summary with them.

- **Preprocessing:**

We removed special characters from the text and also cleaned the text. We also removed all the stopwords and lemmatized the words to be able to apply POS tagging.

- **The most important sentences:**

Based on the value we get from TF-IDF and also the score we get from checking the words of each sentence towards a claim lexicon (https://github.com/webis-de/sigir20-extractive-snippet-generation-for-arguments), we evaluate the importance of the sentences in an argument. All calculated values are normalized to have an objective score.

In the end we select the two sentences with the highest value and add them in the original ordering to the summary.