Documentation

Text slicing project base of the context limit size.

Steps of the project:

- Measure length of documents[*]
- pass to chat gpt if its length was below the limit size[*]
- slice the document if its size is bigger that the limit size with the aims of the split_into_slices function in nlp_utils.py file[*]
  - tokenize the input text in cluding:
    - Tokenization
    - stopword remova;
    - lemmatization
  - calculate the length of tokenized text
  - claculate the number of slices and size of each slice base of the input text size and contex limit size
  - calculate the start index and end index of sliced text and slice it
  - check the text is overlap with previose sliced text
  - check the similarity of text with previose sliced text
  - if it does not have overlap and was similar changing the start index
- checking the overlap
  - make the text lower case
  - find all overlapping sequences of a certain length
- checking similarity
  - vectorization and calculation of cosine distance with two kind of approach (defualt one isTfidfVectorizer which consider the count of occurrences weighted on the length of the document):
    - CountVectorizer
    - TfidfVectorizer
    - Also another approach is to check the similarity meaning of the word in the text
  - calculate cosine distance and compare it to the threshold

Project Structure:

- config file: include the config data like context limit size
- llm file: contains the code which are responsible to connecting to the chat gpt api.
- main file: the start point of application
- npl utils file: contains all the method which are relatied to the text processing like;
  - word tokenizing
  - checking similarity
  - checking overlapping text
  - slipiting the text

Running Project

In windows run these commands in rout of project: * pip install -r requirements.txt * py ./main.py

In Linux base systems: * pip3 install -r requirements.txt * python3 ./main.py