Running the application –

1. Before starting our application we need to start the Stanford Core NLP server. So after downloading the JAR, we run the following command –

java -mx4g -cp "\*" edu.stanford.nlp.pipeline.StanfordCoreNLPServer -annotators "tokenize,ssplit,pos,lemma,parse,sentiment" -port 9000 -timeout 30000;

This starts the server at localhost:9000. We will then start an nlp instance in our main python code and would use that to send our sentences to the server for parsing.

nlp = StanfordCoreNLP('http://localhost:9000')

1. After that we run our flask application which contains the back-end logic and the User Interface of our application.

* For running Task-2, we run the following command python3 flaskpage.py

That starts our app on localhost:5000. We open it up on a web browser.

Also a folder is created called CorpusFilesWithoutFeatures that contains the bag of words files for each question in corpus.

* For running Task-3, we run the following command – python3GeneratingCorpusWithFeatures.py

This generates the feature documents in the CorpusFIlesWithFeatures directory.

After that we run the following command - python3 flaskpage1.py

That starts our app on localhost:5001. We open it up on a web browser.

* For running Task-4, we run the following command python3 flaskpage2.py

That starts our app on localhost:5002. We open it up on a web browser.

Figure 1: Front page of our Task-2 Application

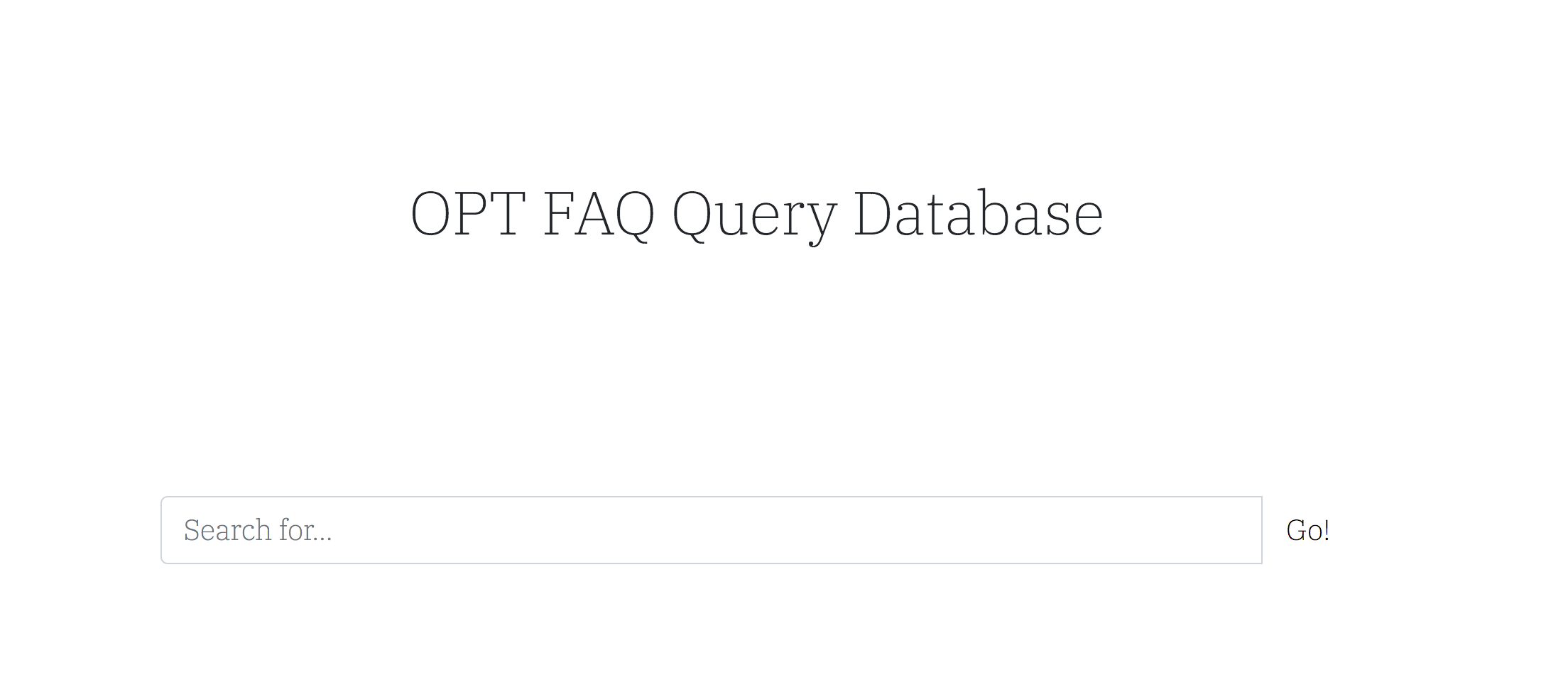


Figure 2: Entering the query and getting relevant results

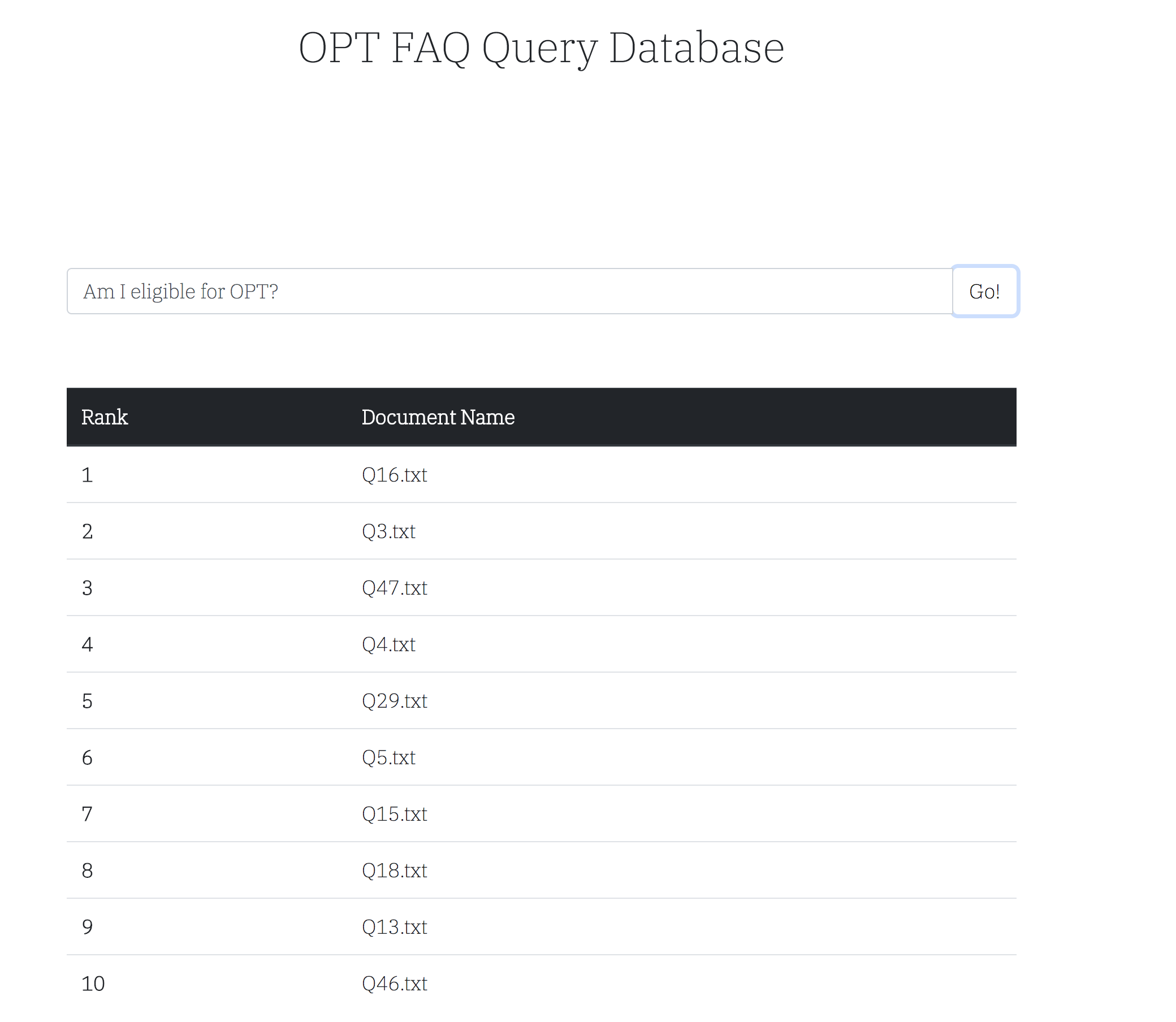


Figure 3: Clicking the document will open the question details

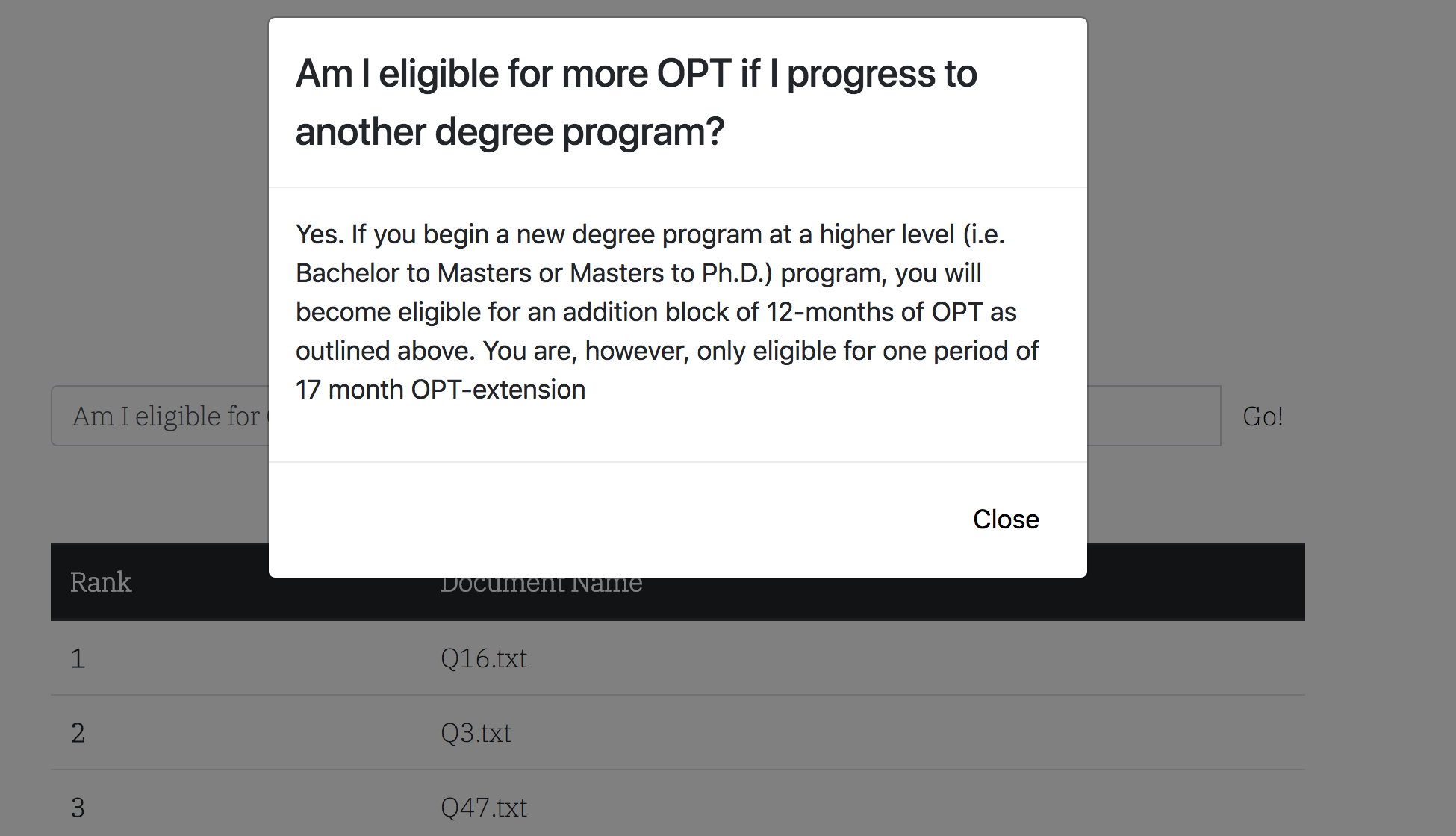


Figure 4: Front page of our Task-3 Application

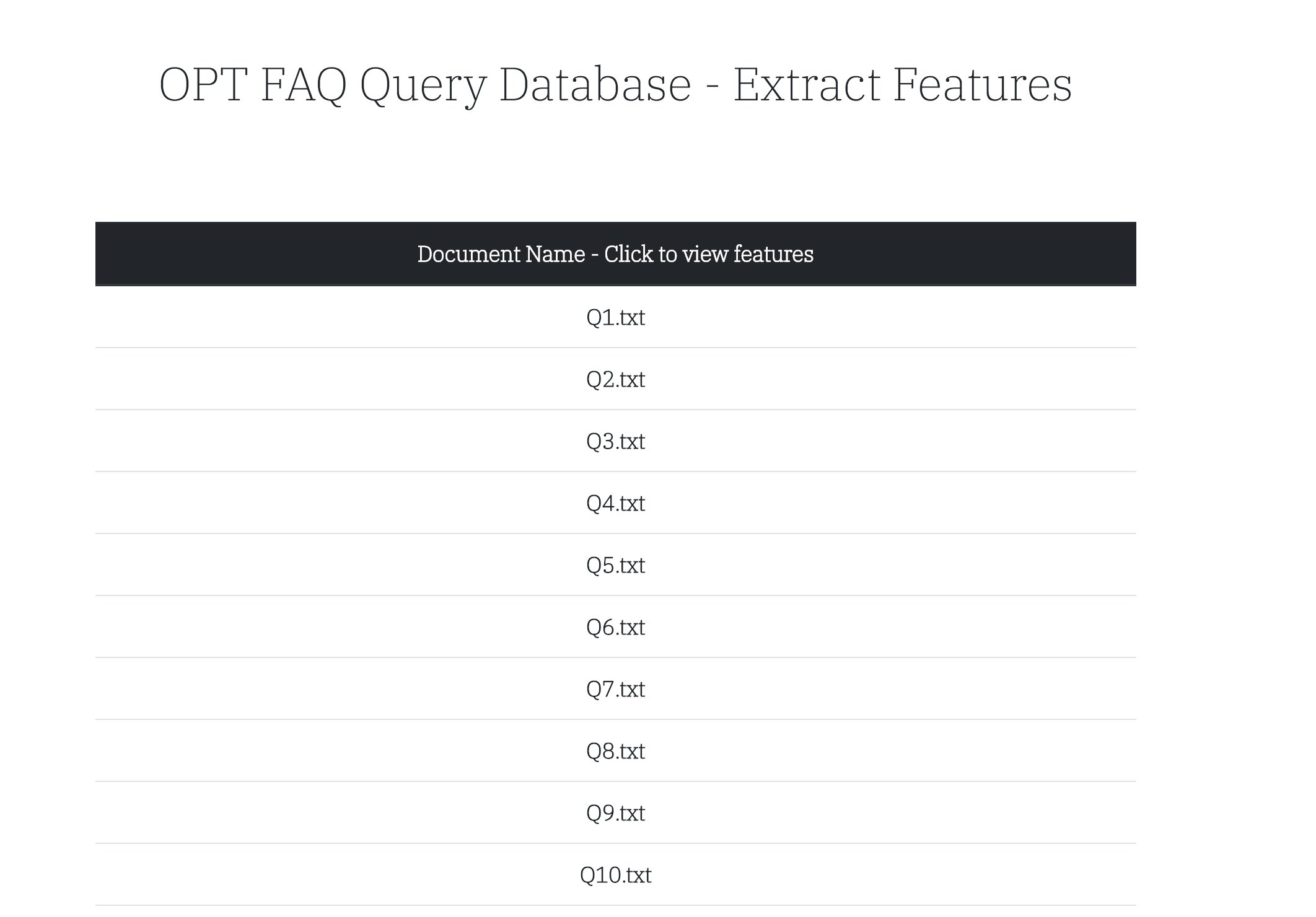


Figure 5: Clicking the document will show you the features extracted

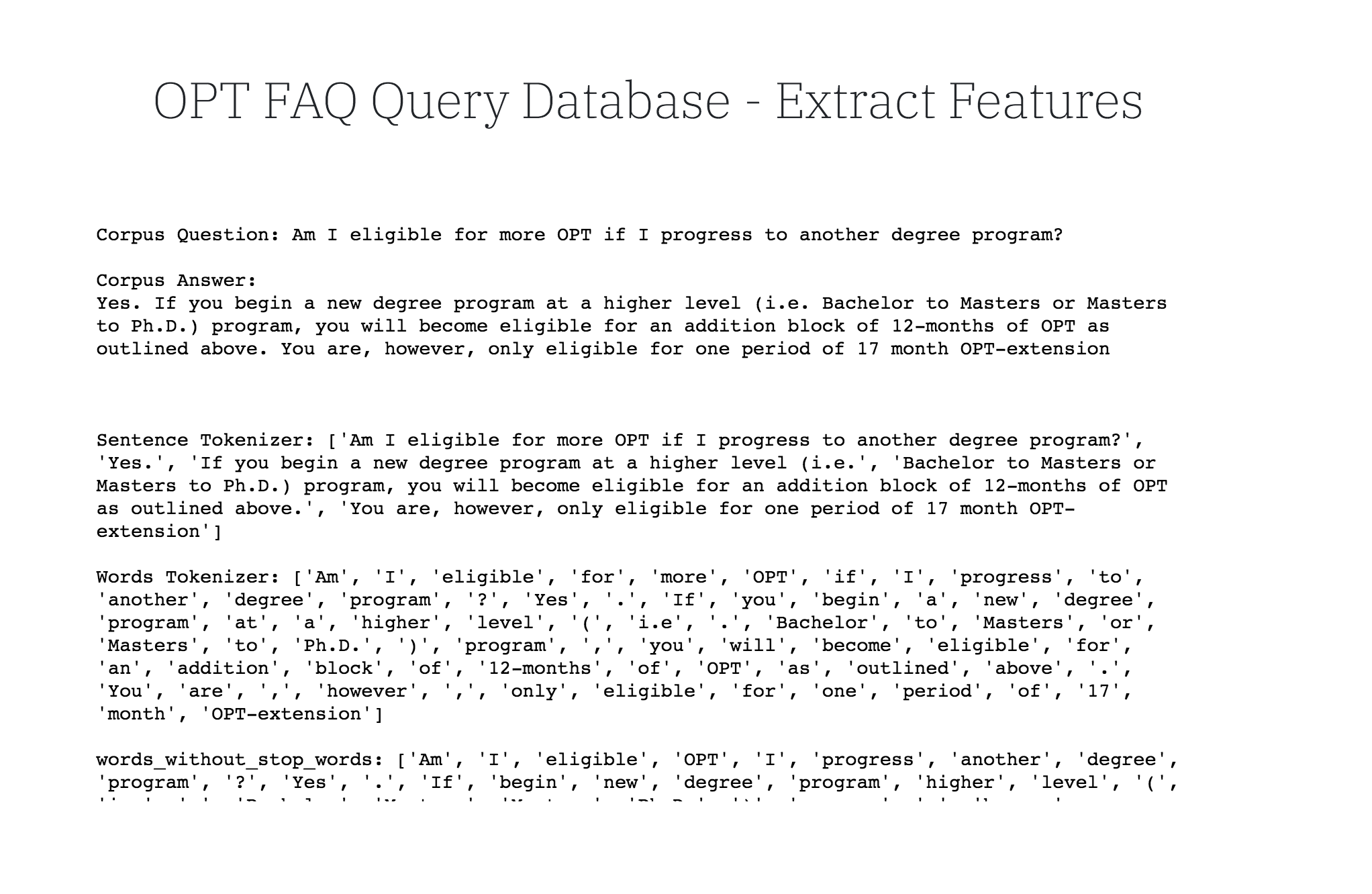


Figure 6: Front-end of our Task-4

