Design Document

The programming language used in the implementation of the Visualization Tool was Python. The first major part of the project was generating relations and statistics from the data files provided. To store the generated data, our team decided to use CSV files instead of other database packages to keep our product lightweight. Two libraries were used for building relations: Natural Language Toolkit for the natural language processing (NLP) and Pandas for organizing and generating the CSV files.

The data file was filtered to remove unnecessary characters such as new line characters (\n) and section number headings. To obtain information about the entire data file, the data was tokenized and stop words were filtered out, resulting in a data structure containing the filtered tokens. For obtaining information about each section, the original data needed to be parsed to find each section. The data files provided contained tags which indicated the end of sections of text and the location of figures in the text. Using the section tags, the data file was split into an array that contained each section's content. The filtered sections were tokenized, and stop words were removed. Using this final data of filtered tokens and sections, a frequency distribution could be generated to obtain CSVs.

Multiple different CSV files were generated in the initial NLP. The frequency of words was obtained for the entire document and each section. A set of important words could be selected, and the frequency of those words for the document and section could be generated. The total word count for each section was generated. Using NLP, all nouns from the data file were obtained and the frequency of those nouns were recorded for the entire file. Finally, the location of figures and graphs in the data file were recorded with the section number.

After the CSVs have been generated, it created the ability to plot the information that was inside the CSVs. Although multiple different CSV files were generated, only four CSV files could be integrated for the final visualization. One CSV file is a small CSV indicating which sections have associated graphs or figures which is called "Section files". Also, a large CSV that will contain all words and their frequency per section called "Section word statistics". The third CSV is a file that is similar to the former but shows total word frequency for the entire document called "Document word statistics". The last CSV that could be generated is a CSV that contains the number of total words occurring in each section called "Section word count". If one of the four CSV files are produced, then a graph could be generated based on the information in the CSV file.

The Python library Matplotlib was used to graph the information provided in the CSV files. The five different graphs are frequency per section, frequency comparison, multiple word frequency, multiple word frequency alternative and section word frequency. The frequency per section creates a bar graph showing the frequency of a given word per section in which that word appears. The frequency comparison creates a line graph comparing the frequency of several words for the entire document. The multiple word frequency creates a line graph with one line per indicated word comparing frequencies for each section in which any of the words appear.

The multiple word frequency alternative creates the same visual as the multiple word frequency, but it uses a bar graph instead of a line graph. The section word frequency creates a line graph comparing the total word occurrence of each section.

Furthermore, the GUI provides an easy way to find file paths to provide data and choose where to store the CSV file that is generated. Also, a button is available next to each graph type that could be clicked on to generate the specific graph. There is also a small window that describes the different CSV and graphs that could be generated. In the small window there is also a couple instructions on how to use the program.