Information Retrieval CS6322.001 Assignment 1 (Tokenization & Stemming)

SHIVAM GUPTA (NET ID:- SXG190040)

Part 1:

Tokenization

- The number of tokens in the Cranfield text collection are: 201520
- The number of unique words in the Cranfield text collection are: 8514
- The number of words that occur only once in the Cranfield text collection are: 3551
- The 30 most frequent words in the Cranfield text collection and their respective frequency are: *[('the', 18682), ('of', 11312), ('and', 5693), ('a', 5283), ('to', 4372), ('in', 4235), ('is', 4107), ('for', 3265), ('are', 2425), ('with', 2082), ('by', 1695), ('at', 1591), ('that', 1565), ('on', 1552), ('flow', 1412), ('be', 1269), ('an', 1258), ('as', 1101), ('this', 1079), ('from', 1067), ('pressure', 1021), ('which', 968), ('number', 905), ('results', 873), ('it', 851), ('mach', 725), ('boundary', 722), ('was', 698), ('theory', 693), ('method', 634)]*
- The average number of word tokens per document are: 143.94

Description of Tokenization:

Tokenization is basically extracting different words from documents. As Cranfield documents collection contains the documents which are of XML type. So, I have used XML.dom(Document Object Model) Python API to read all the documents by parsing all the Cranfield documents. I have considered only the TEXT tag part in the documents which is useful for tokenization and removed all the other SGML tags(like <DOCNO>, <TITLE>, <AUTHOR>, <BIBLIO>) and extracted the TEXT part from the documents. I have removed all the punctuations and the special characters (!"#$%&'()*+,-./:;<=>?@[\]^_`{|}~). The documents contain a lot unnecessary spaces which are also removed. I have also removed all the numerics and extracted the tokens by splitting by space in the string.

Some ***important Information*** regarding ***tokenization*** are:

1) *The program handles*:

- All the words are converted into lower-case.
- The program removes all the dashes between the words and splits into two different words.
- The Possessives (') are also removed using the remove_punct_special_chars function created in the program.
- Acronyms containing . are also removed and converted into a single token.

2) *Data Structures used are*:

Lists, Dictionaries (hash maps) and sets. Lists are used for storing the tokens. Hash map for storing the corresponding stem words. Sets are used for storing the distinct tokens (without duplicates)

Note: The total time taken for tokenization and Stemming from the Cranfield Document Collection is: 1.6 seconds.

Part 2:

Stemming

- The number of distinct stems in the Cranfield text collection are: 5839
- The number of stems that occur only once in the Cranfield text collection are: 2428
- The 30 most frequent stems in the Cranfield text collection and their respective frequency information are:

  [('the', 18682), ('of', 11312), ('and', 5693), ('a', 5283), ('to', 4372), ('in', 4235), ('is', 4107), ('for', 3265), ('ar', 2426), ('with', 2082), ('on', 1818), ('by', 1695), ('flow', 1598), ('at', 1591), ('that', 1565), ('be', 1366), ('an', 1258), ('number', 1236), ('pressur', 1180), ('as', 1101), ('thi', 1079), ('result', 1072), ('from', 1067), ('it', 1029), ('which', 968), ('effect', 839), ('method', 820), ('solut', 785), ('theori', 780), ('boundari', 750)]

- The average number of word-stems per document are: 143.94

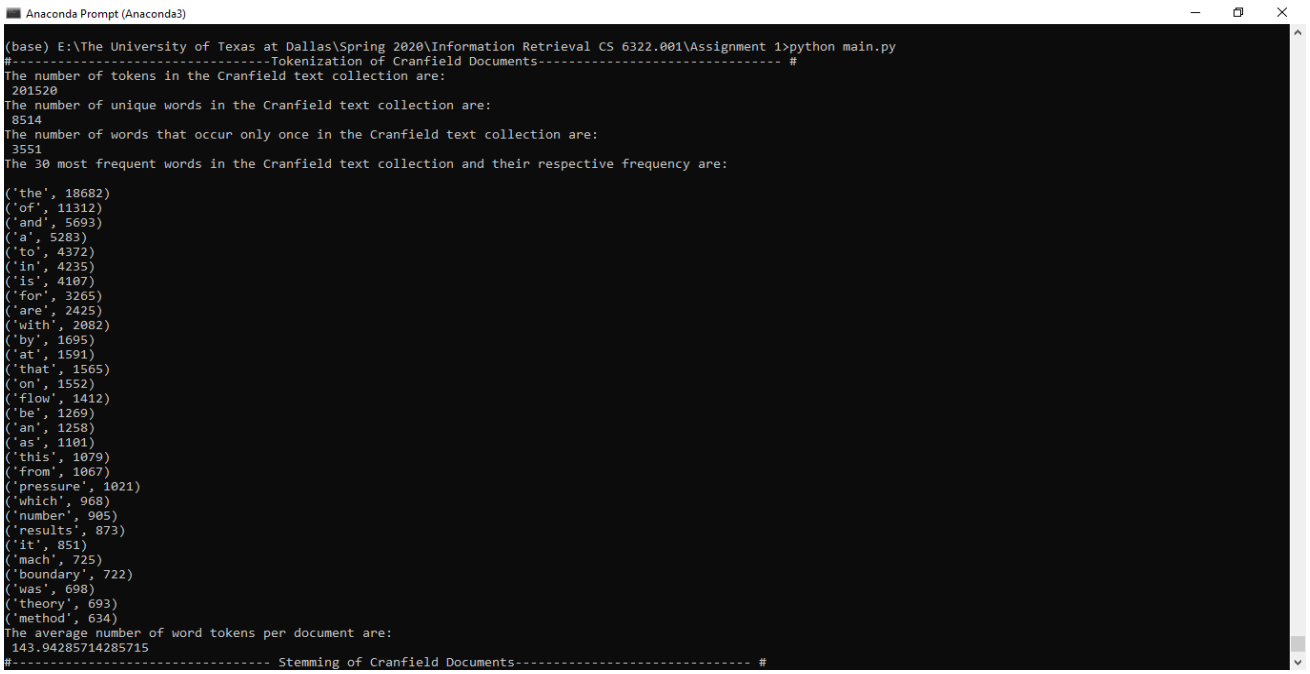
Description of Stemming :

Stemming is the process of reducing the tokens and converting it into stems. In stemming, I have used the open-source porter stemmer algorithm provided by tartarus.org, the link is: https://tartarus.org/martin/PorterStemmer/python.txt. I have used this python file as a library and imported and used the stem function from it to get the stem words for each token.

In the end, I have also created a dictionary which contains the tokens and their respective stem words.

Note: The total time taken for tokenization and Stemming from the Cranfield Document Collection is: 1.6 seconds.

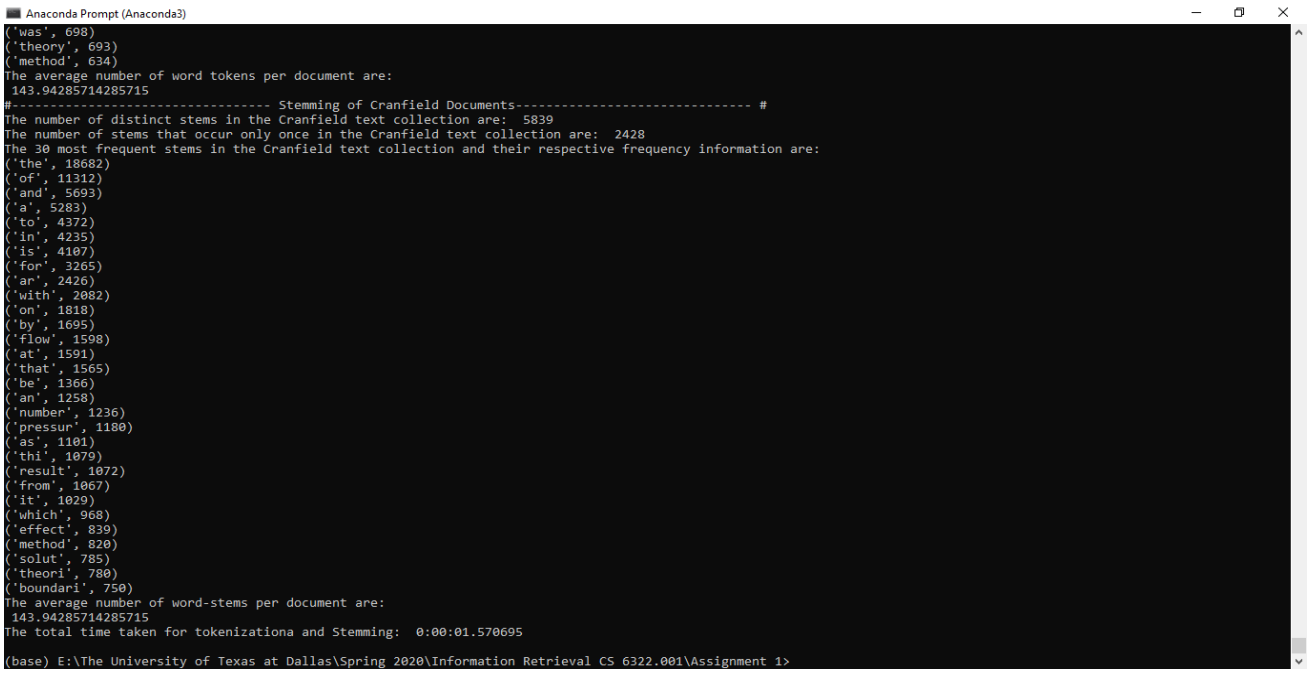
Program Execution and Outputs Screenshots



```
Anaconda Prompt (Anaconda3)

(base) E:\The University of Texas at Dallas\Spring 2020\Information Retrieval CS 6322.001\Assignment 1>python main.py
#---------------------------------Tokenization of Cranfield Documents------------------------------ #
The number of tokens in the Cranfield text collection are:
 201520
The number of unique words in the Cranfield text collection are:
 8514
The number of words that occur only once in the Cranfield text collection are:
 3551
The 30 most frequent words in the Cranfield text collection and their respective frequency are:

('the', 18682)
('of', 11312)
('and', 5693)
('a', 5283)
('to', 4372)
('in', 4235)
('is', 4107)
('for', 3265)
('are', 2425)
('with', 2082)
('by', 1695)
('at', 1591)
('that', 1565)
('on', 1552)
('flow', 1412)
('be', 1269)
('an', 1258)
('as', 1101)
('this', 1079)
('from', 1067)
('pressure', 1021)
('which', 968)
('number', 905)
('results', 873)
('it', 851)
('mach', 725)
('boundary', 722)
('was', 698)
('theory', 693)
('method', 634)
The average number of word tokens per document are:
 143.94285714285715
#---------------------------------- Stemming of Cranfield Documents------------------------------ #
```



```
Anaconda Prompt (Anaconda3)

('was', 698)
('theory', 693)
('method', 634)
The average number of word tokens per document are:
 143.94285714285715
#---------------------------------- Stemming of Cranfield Documents------------------------------ #
The number of distinct stems in the Cranfield text collection are:  5839
The number of stems that occur only once in the Cranfield text collection are:  2428
The 30 most frequent stems in the Cranfield text collection and their respective frequency information are:
('the', 18682)
('of', 11312)
('and', 5693)
('a', 5283)
('to', 4372)
('in', 4235)
('is', 4107)
('for', 3265)
('ar', 2426)
('with', 2082)
('on', 1818)
('by', 1695)
('flow', 1598)
('at', 1591)
('that', 1565)
('be', 1366)
('an', 1258)
('number', 1236)
('pressur', 1180)
('as', 1101)
('thi', 1079)
('result', 1072)
('from', 1067)
('it', 1029)
('which', 968)
('effect', 839)
('method', 820)
('solut', 785)
('theori', 780)
('boundari', 750)
The average number of word-stems per document are:
 143.94285714285715
The total time taken for tokenizationa and Stemming:  0:00:01.570695

(base) E:\The University of Texas at Dallas\Spring 2020\Information Retrieval CS 6322.001\Assignment 1>
```