**Information Retrieval Systems**

**Lab Practical and date** – Practical 5, Wednesday 16th September 2020

**Name and Roll Number**- Het Shah, 17BIT103

**Practical Objective**- Implement SVD algorithm on text data.

* Represent a term-document matrix in lower rank approximation.
* Give a query to the matrix, and return the rank of the similar documents.

**Steps Involved**

● The Tokenization task was performed on an input file that consisted of at least 10 documents (i.e. corpus) from a minimum of two domains(1st domain is of movies and the other is banking system).

● After that, the Stop Words Removal in the same file list is been created to remove all the helping verbs, punctuation, etc in the document.

● In the third step, Stemming is being performed on the statement to bring the words back to their root form by removing the certain suffixes like ‘er’, ’ed’, ’ing’, ’tion’ etc.

● Finally, the case conversion methods were applied to sentences to convert from lower case to upper and vice-versa

● Document term matrix was printed in which the x axis was the words while the y axis consisted off the documents. If a particular word was present in that matrix then it would be 1 or else it would be 0

● The inbuild SVD function of NUMPY was used to convert the document term matrix to the lower rank approximation form.

**Python Package Used**

* We used inbuild python data structures such as Dictionary and arrays and to take the input we used the I/O operations from the files
* We used NUMPY to reduce the rank of the matrix

**Sample Input/Output**

The input is in the form of corpus consisting of 10 text files by the name of main.py

The output is presented in the form of document term matrix in the form of output.pdf. After each step the output the corpus of words are printed as well.

The document term matrix and its reduced form is also printed in the console

**Conclusion**

In this practical, we manually coded all the steps for text preprocessing and took into consider 10 different documents consisting of 2 domains. In the end, I also printed the document matrix for the same while was then represented in the lower rank approximation form.