

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Programme | : | **B.Tech** | Semester | : | **Win Sem 22-23** |
| Course | : | **Web Mining** | Code | : | **CSE3024** |
| Faculty | : | **Dr.Bhuvaneswari A** | Slot | : | **L3+L4** |
| Date | : | **09-01-2023** | Marks | : | **10 Points** |

**Exercise 3: PART A**

**Inverted Index Creation and Searching**

1. Build the inverted index for the following documents:

ID1 : Selenium is a portable framework for testing web applications

ID2 : Beautiful Soup is useful for web scraping

ID3: It is a python package for parsing the pages

ID4: Java programming can be used for web applications

ID5: scraping web and crawling web is useful

1. Search following words using the inverted index
   1. Selenium AND web
   2. Soup
   3. Python OR java
   4. Web AND craw

**HINTS:**

We will first pre-process the documents, and then split the documents into tokens or words. The pre-processing includes conversion to lower case, removal of numbers and other special characters, and stop word removal. After this we tokenise this document using the **NLTK** library.

After this we then take all these words after tokenisation and form a inverted index, which will include the word, and a list of occurrences in all the documents, and in each entry of this array would include the document number, number of times this word has occurred in this document, and also a list of offset position where the word occurs in the document.This map developed in the above stated process is our **Inverted Index** map for the documents.

**PART B**

**Boolean and Vector Model, TF-IDF, Similarity Measures**

Consider the following documents.

Doc 1 : Information Retrieval Systems is used with database systems

Doc 2 : Information is in Storage

Doc 3 : Digital Speech can be used in Synthesis and Systems

Doc 4 : Speech Filtering, Speech Retrieval systems are applications of Information Retrieval

Doc 5: Database Management system is used for storage

1. Perform the text pre-processing of the given documents.
2. Construct a Boolean Model for the vocabulary by considering documents 1, 2, 3,4 and 5.
   1. Retrieve the documents for the Boolean query “Information Retrieval Synthesis” using simple match. (Rank the documents in the order of relevance)
   2. Retrieve the documents for the Boolean query “Database Retrieval Storage” using weighted match. (Rank the documents in the order of relevance)
3. Construct a vector space model to build the term weights. Compute the TF-IDF and identify the most important terms across the documents.
   1. Rank all the documents in the collection for the query “Speech Systems”? (Rank the documents in the order of relevance)
4. Compute the cosine similarities between docs 1 and docs 2
5. Compute Dice Co-efficient between docs 3 and docs 4.
6. Compute the Jaccard co-efficient between docs 4 and docs 5.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*