**Objective**: Practice the creation of Stored Procedure to apply Similarity algorithms for ranking documents using SQL

# Exercise 1: Using Cosine Similarity algorithm

We have given 5 document titles (considered as 5 documents for the purpose of this lab) and index terms that are significant in our collection. Fill the document – term matrix with frequency (𝑡𝑓𝑖,𝑗) of occurrences of each of the listed terms in the 5 documents

Given the following 5 document titles, fill the document/title-term matrix (table below) with the frequency (𝑡𝑓𝑖,𝑗) of occurrences of each of the listed terms in the 5 document titles.

𝑡𝑒𝑟𝑚 𝑓𝑟𝑒𝑞𝑢𝑒𝑛𝑐𝑦 (𝑡𝑓𝑖,𝑗), 𝑖𝑠 𝑡ℎ𝑒 𝑓𝑟𝑒𝑞𝑢𝑒𝑛𝑐𝑦 𝑜𝑓 𝑜𝑐𝑐𝑢𝑟𝑒𝑛𝑐𝑒 𝑜𝑓 𝑡𝑒𝑟𝑚 𝑡𝑖 𝑖𝑛 𝑑𝑜𝑐𝑢𝑚𝑒𝑛𝑡 𝑑𝑗

# a. Document -Term Matrix Table

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **DocID** | **Title/Term** | **computer** | **digital** | **information** | **security** | **system** |
| 1 | Understanding the digital world: what you need to know about computers, the Internet, privacy, and security | 1 | 1 | 0 | 1 | 0 |
| 2 | Management information systems and databases | 0 | 0 | 1 | 0 | 1 |
| 3 | The Information Systems Security Officer Guide: Establishing Security and Managing a Cyber Security Program | 0 | 0 | 1 | 3 | 1 |
| 4 | Computer security, Computer science and Computer systems | 3 | 0 | 0 | 1 | 1 |
| 5 | Database : models, language, design, system | 0 | 0 | 0 | 0 | 1 |

To help you with calculating cosine similarity among documents, we have provided you the following. Understand the purpose of each. b. Create a schema named **textsearch** in your mysql DBMS.

1. Create the **DocTermFreq** table in the **textsearch** database using the script given below. This table will be used to store frequency count of appearance of the term in the document’s title. For each document record (docID) and term (term), corresponding frequency count value is stored in the in the column "frequency" of the table **DocTermFreq**

|  |  |  |
| --- | --- | --- |
| docID int | term  Varchar(100) | frequency  int |

DocTermFreq

CREATE TABLE DOCTERMFREQ

( DOCID INT,

TERM VARCHAR(100),

FREQUENCY INT,

CONSTRAINT PRIMARY KEY (DOCID, TERM) );

1. Create the following stored procedure named **InsertFreq** using the script given below to populate the frequency count for every term based on its appearance in every document.

CREATE PROCEDURE INSERTFREQ(IN DOCID INT, IN TERM VARCHAR(100), IN FREQ INT)

INSERT INTO DOCTERMFREQ (DOCID, TERM, FREQUENCY)

VALUES (DOCID, TERM, FREQ);

1. Using data from the prepared document/title-term table in part a, call the stored procedure **InsertFreq** to populate the doc term frequency count for all the 5 documents.

Eg: calling the stored procedure to populate frequency count for document 1

call insertfreq(1,'computer',1); call insertfreq(1,'digital',1); call insertfreq(1,'system',0); call insertfreq(1,'information',0); call insertfreq(1,'security',1);

1. Write a stored procedure named **get\_similarity** that takes in two input parameters doc1\_ID(type:int) and doc2\_ID(type:int), and one output parameter sim\_val(type:double) to calculate the cosine-similarity value for any 2 records (corresponding to 2 documents i.e. DocIDs).

𝑣𝑎 ∙ 𝑣𝑏

𝑠𝑖𝑚𝑖𝑙𝑎𝑟𝑖𝑡𝑦(𝑣𝑎, 𝑣𝑏) = |𝑣𝑎||𝑣𝑏|

Refer to week 12 - text\_search slides. Hint: Use variables to store the intermediate values 𝑣𝑎 ∙ 𝑣𝑏 , |𝑣𝑎| 𝑎𝑛𝑑 |𝑣𝑏| in variables by using [Select col1, col2, col3 into var1, var2, var3 from table1].

1. Given a query Q of terms (system, computer, information). Using **get\_similarity** stored procedure, calculate the similarity of query Q against all 5 document records.

Tick the document that is most similar to the query Q. (Hint: your query may also be placed as a record in the DocTermFreq table).

|  |  |  |
| --- | --- | --- |
| **DocNo** | **cosine-similarity value of Document with query Q** | **Tick the Most Similar Document to Query Q** |
| 1 | 0.333 |  |
| 2 | 0.8165 | ✔ |
| 3 | 0.3483 |  |
| 4 | 0.6963 |  |
| 5 | 0.5774 |  |

# Exercise 2: Document Search using TF-IDF Weighting algorithm

1. Calculate the 𝑑𝑓𝑖 and IDF 𝑖𝑑𝑓𝑖 = 𝑙𝑜𝑔2 (𝑑|𝐷𝑓|𝑖) for all the terms. *D denotes the set of all documents and* 𝑑𝑓𝑖*refers to the number of documents containing index term* 𝑘𝑖*.* Leave your answer in the form of 𝑙𝑜𝑔2 (𝑑|𝐷𝑓|𝑖), do not use a calculator.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Term** | **computer** | **digital** | **information** | **security** | **system** |
| 𝑑𝑓𝑖 | 2 | 1 | 2 | 3 | 4 |
| 𝑖𝑑𝑓𝑖 |  |  |  |  |  |

1. For the top 3 most similar documents to query Q from result of question g, fill the document/title-term matrix (table below) with the *tf-idf weighting*

|𝐷|

(𝑤𝑖,𝑗) = 𝑡𝑓𝑖,𝑗 𝑖𝑑𝑓𝑗 = 𝑡𝑓𝑖,𝑗 𝑙𝑜𝑔2 ()

𝑑𝑓𝑖

𝑁𝑜𝑡𝑒: 𝑙𝑜𝑔2(2.5) = 1.322, 𝑙𝑜𝑔2(1.25) = 0.322, 𝑙𝑜𝑔2 () = 0.737

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **DocNo** | **Title/Term** | **computer** | **digital** | **information** | **security** | **system** |
| 2 | Management information systems and databases | 0 | 0 |  | 0 |  |
| 4 | Computer security, Computer science and Computer systems |  | 0 | 0 |  |  |
| 5 | Database : models, language, design, system | 0 | 0 | 0 | 0 |  |

1. Repeat step c, d and e to create a new weighted (TF-IDF) term frequency table **varDocTermFreq**, a new **insertVarFreq** stored procedure and insert the values for the top 3 documents using the new stored procedure.

1. Given two documents *d*1 and *d*2, their similarity can be measured by the dot product of their tf-idf weights. Write a new stored procedure **get\_weighted\_similarity** to calculate the similarity value for any 2 records (corresponding to 2 document IDs).

𝑁

𝑠𝑖𝑚𝑖𝑙𝑎𝑟𝑖𝑡𝑦(𝑑1, 𝑑2) = ∑ 𝑤𝑗1𝑤𝑗2

𝑗=1

1. Given the same query Q of keywords (system, computer, information),fill the document/title-term matrix (table below) with the *tf-idf weighting* for query Q. Using **get\_weighted\_similarity** stored procedure, calculate the similarity of query Q against the top 3 document records in the table **varDocTermFreq**, fill the table below and tick the document that is the most similar to the query Q.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **DocNo** | **Title/Term** | **computer** | **digital** | **system** | **information** | **security** |
| Q | system,  computer, information |  | 0 |  |  | 0 |

|  |  |  |
| --- | --- | --- |
| **DocNo** | **Similarity value of Document with query Q** | **Tick the Most Similar Document to Query Q** |
| 2 | + = 1.8514 |  |
| 5 | × + = 5.3467 | ✔ |
| 4 | = 0.1036 |  |