**Assignment 4 Report**

Bhavatarini M P - 1PI14IS023

Samrat K S - 1PI14IS053

Problem Statement:

Implementing the Vector Space Model for Scoring and displaying top k documents for a query.

Dataset : **cran.all.1400**

Steps:

1. Tokenisation : Converting the data into tokens
2. Pre-Processing : Converting the tokens into lower case, removing the top words.
3. Stemming : Stem the tokens into root form
4. Index Construction :
   1. Constructed using Dictionary in python

Index - >

Term -> Document Frequency ->

{

doc\_id -> term\_frequency

}

5) Calculating TF - IDF weights :

term freq = 1 + log(tf)

idf = log(N/df)

tf-idf = (1 + log(tf)) \* log(N/df)

Calculating tf-idf using **ltc.ltn** ranking schema where cosine normalisation is done only for documents.

6) Ranking the documents:

After calculating the score for all the documents and the given query, top k (k==10) should be returned.

**Output:**

1. **experimental results on hypersonic viscous interaction**

[26, 1299, 323, 570, 1253, 1395, 525, 63, 333, 305]

1. **properties of impact pressure probes in free molecule flow**

[906, 183, 10, 1139, 405, 1227, 1151, 355, 356, 1257]

1. **manufacturing and maintainance of ideally sharp leading edges and noses is practically impossible**

[211, 900, 1196, 918, 1317, 1267, 337, 544, 1022, 167]

1. **why does the compressibility transformation fail to correlate the high speed data for helium and air**

[502, 1176, 1026, 68, 271, 343, 1022, 389, 340, 376]

1. **can increasing the edge loading of a plate beyond the critical value for buckling change the buckling mode**

[862, 1069, 1023, 1026, 642, 31, 915, 15, 735, 1177]

