**DESIGN OF INFORMATION RETRIEVAL MODEL**

**ARCHITECTURE:**

The retrieval system was aimed at high recall and ranking amongst the returned documents. The query is processed to return documents by the Boolean Retrieval. This results in 100% recall. Then the Vector Space Model is used to rank the documents on the basis of tf-idf score. The ranks generated displays top 20 results in an effort to increase relevance of documents for user and subsequently, precision.

The corpus selected was TUAW. It has 104 documents and total words of the order 5\*105 , with 105 unique words.

**PHASE A: DATASET CONSTRUCTION**

**STEMMING AND NORMALIZATION**

The input file data is stemmed using Porter’s Algorithm in ANSI C language.

**TOKENIZATION AND INDEX CONSTRUCTION**

The inverted index is created using maps data structure. The obvious choice was to consider struct data type. This required memory of the order 104 \* 105 since we need frequency of every word in each document. This was an expensive choice in terms of memory storage. Hence, the alternative chosen is maps data structure, We used nested maps i.e map[i][j]=k represents k is the frequency of ith word in jth document. This reduces the memory occupied. The words are tokenized using in-built package in boost library.

**RETRIEVAL- BOOLEAN & VECTOR SPACE MODEL- RANKING**

The query is processed to obtain documents via Boolean retrieval which are further processed to calculate the tf-idf score to give top 20 ranked documents.

**PHASE B: QUERY PROCESSING**

**QUERY SUGGESTION-EDIT DISTANCE**

The entered query is read and then suggestions are given in case the query is mistyped. This is done by calculating edit distance of query terms with respect to terms in dataset. The selected query is then finally used for retrieval.

**RESULT DISPLAY**

The terms in the query are also processed(stemmed, normalize and tokenized). The matching documents from the dataset are then returned in the form of DocIDs.