Course Project
CS6370: Natural Language Processing

Naveen Vakada (CS20S012)
Arup Das (CS20S016)

Limitations of the Vector Space Model:

The limitations of the current search engine primarily lies in the algorithm used to find the relevant documents as per the user query. From our observations until now:

1. The current search engine fails to classify documents containing synonymous words or documents with the same context as relevant documents.
2. Further, the current search engine suffers from the problem of polysemy.
3. It does not consider the co-occurrence relation between the terms in similar documents.

We attempt to improve the current state of the art of the search engine using the below hypotheses:

Candidate Hypothesis 1(H_1): *Latent Semantic Indexing* instead of just TF-IDF vectorial representation of documents can better handle the problem of synonymy and co-occurrence relation.

Reason: In LSA, it is assumed that terms used in similar contexts have similar meanings and hence are synonymous. We are producing a k-rank approximation of the original term-document matrix. This alleviates the problem of synonymous terms and partially the problem of polysemy as the low rank approximation combines the dimensions of terms having similar meanings.

Candidate Hypothesis 2(H_2): Using *BM25* (Best Match 25) as a ranking function instead of TF-IDF and cosine similarity will produce better relevant results.

Reason: The BM25 formula is as given below:

$\Sigma_{t \epsilon q} IDF(t) * \frac{(k_1 + 1) * tf(t, d)}{k_1[(1 - b) + \frac{b*dl}{avdl}] + tf(t, d)} * \frac{(k_2 + 1) * tf(t, q)}{k_2 + tf(t, q)}$ where $IDF(t) = log(\frac{N}{df(t)})$

In the presence of full relevance judgement, $IDF(t) = log(\frac{\frac{r_t + 0.5}{R - r_t + 0.5}}{\frac{df(t) - r_t + 0.5}{N - df(t) - R + r_t + 0.5}})$

The TF-IDF implementation produces documents relevant to the query irrespective of user's relevance whereas BM25 takes into account the total number of relevant documents while calculating the score. This makes BM25 more efficient based on the relevant documents list.

Candidate Hypothesis 3: Using *Query Expansion* along with Candidate Hypothesis 1 **(H_{31})** or 2 **(H_{32})** will improve the search results.

Reason: Since Query Expansion will introduce additional tokens or phrases to a query ultimately enriching the user's initial query with synonymous words and words which represent a similar context. Hence Query expansion will significantly help in improving the recall.

Candidate Hypothesis 4(H_4): A *convex combination* of Query Expansion with Candidate Hypothesis 1 and Query Expansion with Candidate Hypothesis 2 will produce documents with greater precision and recall than the current search engine.

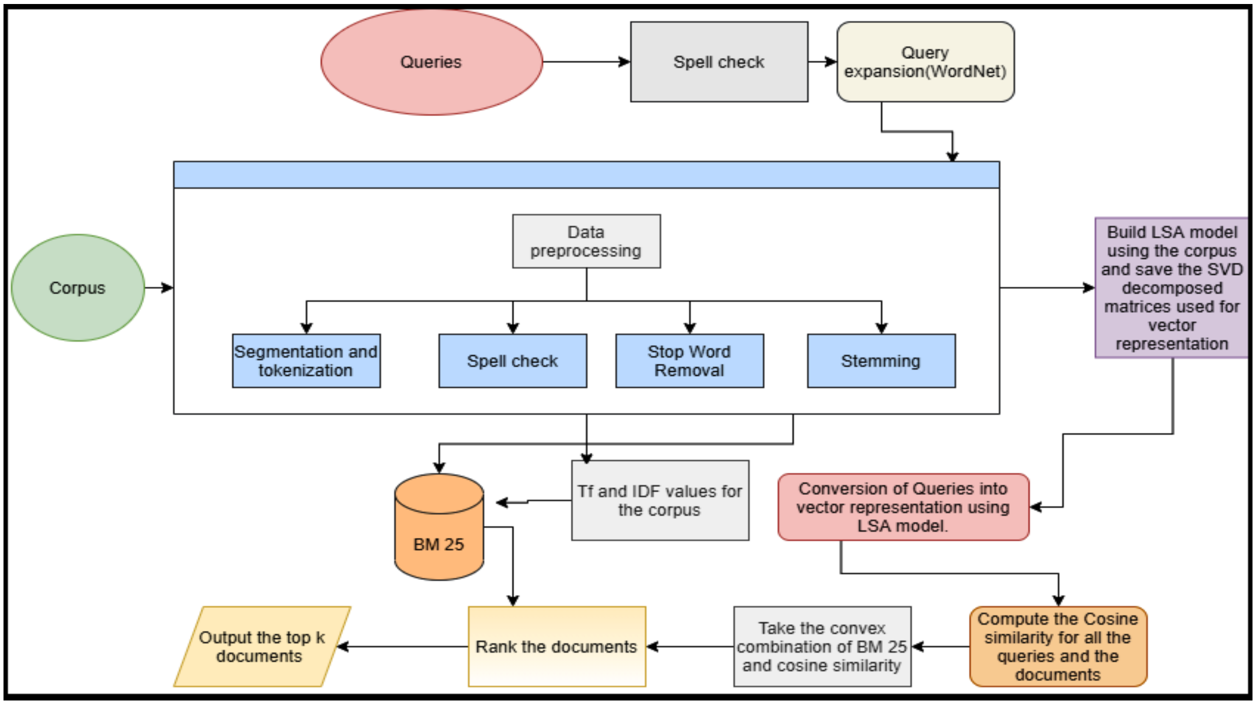
Reason: Formula for convex combination is given by: $\boldsymbol{\alpha} H_{31} + (1 - \boldsymbol{\alpha}) H_{32}$ where $\boldsymbol{\alpha} \epsilon [0, 1]$.

Using the idea of ensemble methods where a combination of multiple models produce more accurate results than a single model, our objective here is to combine the LSA model and the BM25 model where the weightage of combination is dictated by the degree of usefulness of the models in the given evaluation metric. Let a, b be the scores of the given evaluation metric then $\boldsymbol{\alpha} = \frac{a}{a+b}$ and $1 - \boldsymbol{\alpha} = \frac{b}{a+b}$. Such a

procedure would ensure us to use a top down knowledge of the relative strengths of both models instead of parameter tuning which is a bottom up approach agnostic to such top knowledge.
Below **algorithm** enlists the steps that would be adopted to realise the above four candidate hypothesis.

1. Pre-process the corpus(Cranfield Dataset) and the query.
2. Expand the query using Query Expansion with the help of WordNet.
3. Compute the Term Frequency(TF) and Inverse document Frequency (IDF) of the documents in the corpus.
4. Build the LSA model by computing the SVD decomposed matrices U, Σ and V^T.
5. Convert the query into vector representation using the LSA model.
6. Compute the cosine similarity and the BM25 scores.
7. Perform the convex combination of the two scores obtained in step 6.
8. Rank the documents in non-increasing order of scores obtained in step 7.
9. Output the top k results.

The below diagram shows the architecture of the proposed approach:



Queries
Spell check
Query expansion(WordNet)
Corpus
Data preprocessing
Segmentation and tokenization
Spell check
Stop Word Removal
Stemming
Build LSA model using the corpus and save the SVD decomposed matrices used for vector representation
BM 25
Tf and IDF values for the corpus
Conversion of Queries into vector representation using LSA model.
Output the top k documents
Rank the documents
Take the convex combination of BM 25 and cosine similarity
Compute the Cosine similarity for all the queries and the documents

Fig. 1 Architecture of Proposed Approach

Hypothesis Testing methods:

The evaluation metric for the candidate hypotheses are Precision@K, Recall@K, MAP@K, NDCG@K for K=10. We use Hypothesis Testing of Means of two Normal Populations with unknown variances.

Null Hypothesis (H_0): Evaluation metric of a candidate hypothesis does not give a better score when compared to that of the current search engine.

Alternate Hypothesis (H_A): Evaluation metric of a candidate hypothesis gives a better score when compared to that of the current search engine.

References:

1. https://nlp.stanford.edu/IR-book/html/htmledition/latent-semantic-indexing-1.html
2. https://nlp.stanford.edu/IR-book/html/htmledition/okapi-bm25-a-non-binary-model-1.html
3. https://nlp.stanford.edu/IR-book/pdf/09expand.pdf
4. http://www.r-5.org/files/books/computers/algo-list/statistics/probability/Sheldon_M_Ross-Introduction_to_Probability_and_Statistics-EN.pdf