IRE Assignment 2 Report

Name - Pranav Gupta

Roll no. - 2021101095

Task 1 and 2: Preprocessing and Tokenisation (Stemming)

In this task, the text is prepared for retrieval. Punctuation, special characters, and numbers
are removed using the regular expression [ˆa-zA-Z]. This makes the text contain only
alphabetic letters and spaces. Then, the text is separated into tokens, which are essentially words, by splitting the text on the basis of a space. This process is done in order to obtain raw data.

Following this, Porter Stemming Algorithm is used to reduce words to their base or root form. It is a very popular method in NLP to reduce words to their root form. It applies heuristic rules to remove common suffixes from words, simplifying them for analysis. For example, ″playing″ becomes ″play″ after stemming. In this task, the tokens were stemmed using the PorterStemmer from nltk.stem. This gives the root forms of the tokens. An option was given as to whether to perform stemming or not or whether to remove stopwords from the corpus or not.

Task 3: Computation of tf-idf

In this task, the Term Frequency (tf) and TF-IDF (tf-idf) are computed for each document.
Subsequently, the top p stems in each document are compared based on both tf and
tf-idf. These two metrics are essential in information retrieval and text mining, aiding in
understanding the importance of terms within a document and across a collection.

Term Frequency (tf)

tf measures how often a given term appears in a specific document. It is calculated as:

tf(ti, dj) = freq(ti, dj)/summation over all k(freq(ti, dk)) where, tf(ti, dj) is the term frequency of term t in document d and freq(ti,dj) is the number of occurrences of the term ti in document dj. The purpose of tf is to normalize the term frequency by dividing it by the length of the document (i.e. the number of terms in the document). This normalization removes biasness from longer documents so that longer documents don't have a natural advantage in the sense that they will have more occurrences of different terms in the corpus.

Inverse Document Frequency (idf)

idf measures the importance of a term in the entire corpus. It is calculated as:

idf(ti) = log(N/nti) where, idf(t) is the Inverse Document Frequency of term ti, N is the total number

of documents in the corpus and nti is the number of documents containing term ti. idf gives higher weight to terms that are rare across the entire corpus and lower weight to common terms because rarer words add more value to the meaning and representation of the corpus.

Term Frequency-Inverse Document Frequency (tf-idf)

tf-idf combines tf and idf to assess the importance of a term within a document and the

entire corpus. It is calculated as: tf-idf(t, d) = tf(t, d) × idf(t)

The tf-idf value reflects how significant a term is within a document (tf) while considering

its rarity across the entire corpus (idf). High tf-idf values are obtained for terms that are

frequent within a document but relatively uncommon across the corpus.

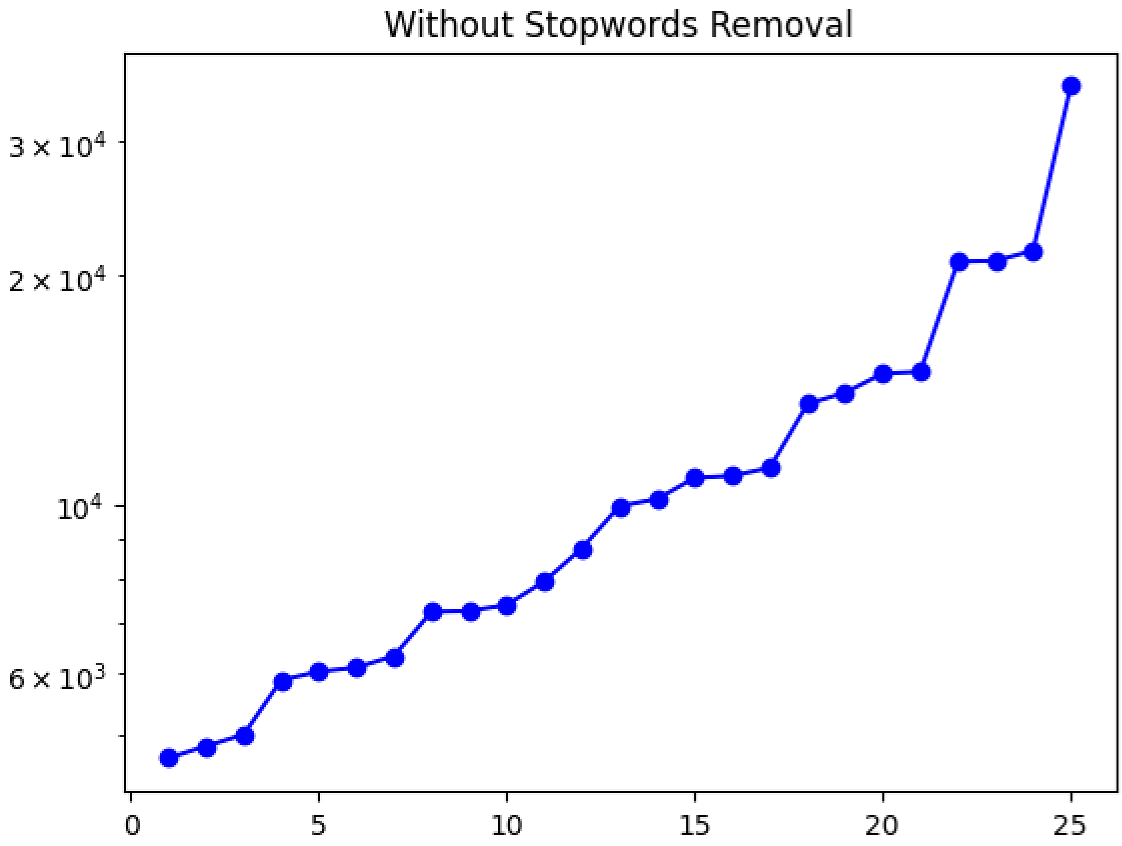
Stopwords removal

Stopwords are commonly used words in a language (such as "the," "is," "in," "and," etc.) that are often considered unimportant for understanding the main content of a text. They are typically filtered out in natural language processing

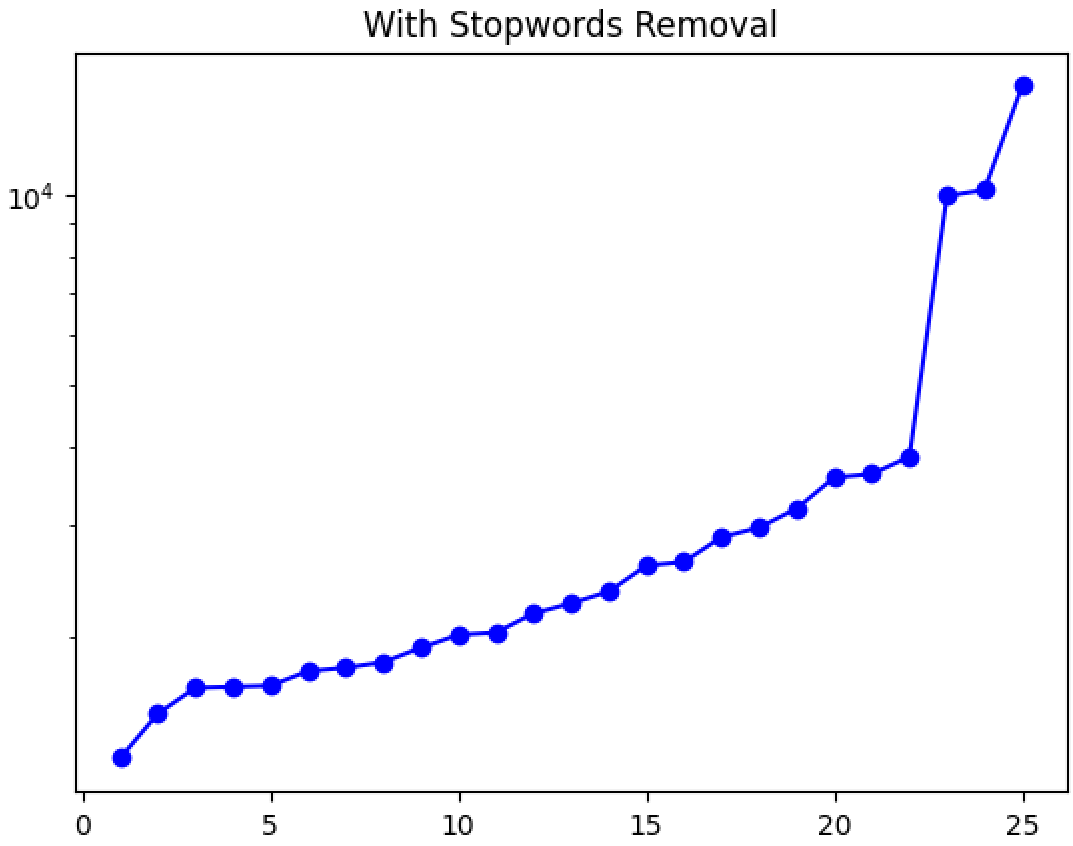
(NLP) tasks because they don't carry much meaning or contribute to the analysis of the text.

For example, in text mining and search engines, removing stopwords helps focus on the significant words in a document, making text processing more efficient.

The english.stop file (of the previous assignment) containing stopwords of English Language was loaded. Changes were only to the existing class where a check was written if we have to remove stopwords or not. Then, the stopwords were removed from all the documents as well as queries so that they do not interfere with the learning of the model. Following are the graphs for the frequency distributions obtained before and after performing stopwords removal.



Frequency Distribution of words before Stopwords removal



Frequency Distribution of words after Stopwords removal

Top-25 Stems in the Vocabulary before Stopwords Removal and their respective frequencies: {'the': 35477, 'to': 21567, 'you': 20917, 'a': 20865, 'of': 14950, 'insur': 14876, 'is': 14038, 'and': 13569, 'that': 11187, 'your': 10918, 'for': 10852, 'polici': 10179, 'life': 9963, 'in': 8769, 'be': 7935, 'are': 7390, 'it': 7258, 'if': 7241, 'have': 6331, 'or': 6117, 'can': 6041, 'will': 5889, 'with': 4993, 'not': 4824, 'as': 4653}

Top-25 Stems in the Vocabulary after Stopwords Removal and their frequencies: {'insur': 14876, 'polici': 10179, 'life': 9963, 'compani': 3845, 'term': 3612, 'coverag': 3566, 'rrb': 3184, 'premium': 2970, 'pay': 2869, 'lrb': 2621, 'year': 2586, 'plan': 2352, 'death': 2254, 'benefit': 2172, 'rate': 2026, 'cash': 2009, 'agent': 1916, 'question': 1818, 'amount': 1782, 'health': 1760, 'time': 1669, 'cover': 1662, 'cost': 1656, 'person': 1507, 'age': 1282}

We notice that now the frequencies of the 25 most occurring words has significantly reduced. It is because the most commonly occurring words are mostly the stopwords themselves and removing them gives an opportunity to the lesser frequency words to appear in the learning of the model. Also, be observing the Top p-stems in the corpus after stopwords removal, words that do not contribute anything to the learning process have been removed which results in better generalisation.

Task 4: Implementation of Probabilistic and LSI(Latent Semantic Indexing) Models

Probabilistic Model

The key components and steps involved in this process include:

This ProbabilisticModel class implements an iterative refinement process to compute document-query similarities using a probabilistic model. Here is a brief summary of its key components and steps:

1. Initialization:
    - The constructor initializes key variables like document/query preprocessing, their TF-IDF values, word occurrence probabilities, and similarity scores between documents and queries. These values are already stored in the form of an object which is an instance of the above defined Preprocessing class.
    - Probabilities of a term being in either the relevant or non-relevant documents are initialized to 0.5. This is done because, initially, we are not aware which document in the corpus is relevant document for the query, and it is believed that the likelihood of a term lying in any document is 0.5 only.
    - The parameter p determines the number of most frequent word stems in the corpus to consider, and the number of refinement iterations is defined as the value 25.
    - The parameter V determines the number of relevant documents to be considered for each query given as input by the user, whose value is set as 15 for the assignment.

2. P-stems:

   - Following the above, we find the top p stems which are occurring in the dataset depending upon the cumulative frequency if each term in the corpus.

3. Similarity evaluation of each query and document:

   - This function computes the similarity between a specific document and a query. The vectors used for this task is the term-document matrix (TF-IDF Matrix), which is already constructed in the above parts. The similarity score is evaluated using the TF-IDF values of shared words and the log-ratio of relevant and non-relevant probabilities.

4. Finding most similar documents for each query:

   - This method computes similarity scores between all document-query pairs. It then identifies the V (defined as a Hyperparameter, whose value is set as 15 for the assignment) most similar documents for each query and updates the word probabilities based on these documents.

5. Probabilities Updation:

   - This step updates the probabilities of a document being relevant or non-relevant for each term occurring in the corpus based on the most similar documents to the queries. It uses the number of occurrences of the most frequent stems to compute new probabilities. Probabiilties are updated as follows:

$$P(t_i|R) := \frac{|V_i| + 0.5}{|V| + 1}, P(t_i|\bar{R}) := \frac{n_i - |V_{i|} + 0.5}{N - |V| + 1}.$$

6. Iterative Refinement:

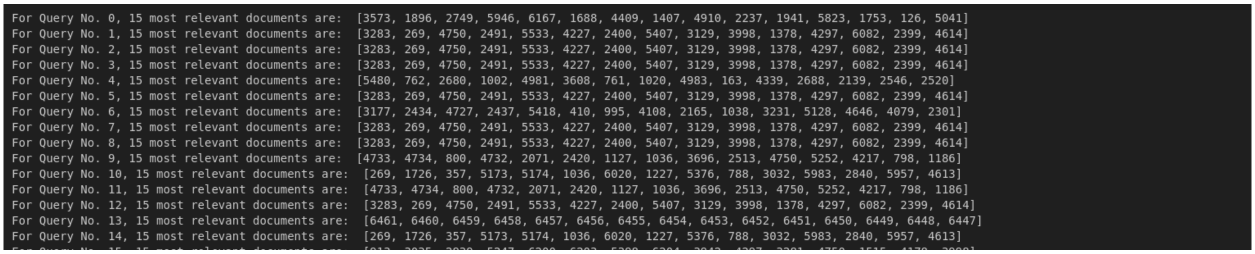
   - This step essentially performs the above steps in a loop for num_iters times (which is set as 10 in the code).

In essence, the model refines document-query similarity by iteratively adjusting word relevance probabilities based on the most similar documents found in each iteration.

- Vector Query Comparison: The similarity between the user’s query vector and each document’s vector is calculated using Similarity Function defined as below:

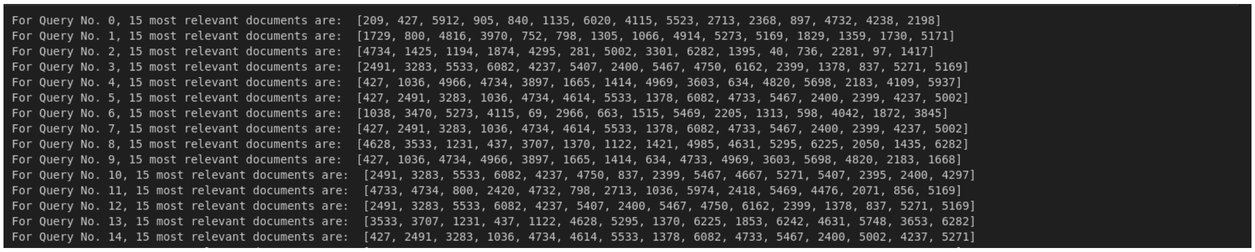
$$sim(d_j, q) = \sum_{i=1}^{k} w_{iq} \cdot w_{ij} \cdot \left(\log \frac{P(t_i|R)}{1 - P(t_i|R)} + \log \frac{1 - P(t_i|\bar{R})}{P(t_i|\bar{R})} \right),$$

Output: The the top ’V’ documents that exhibit the highest similarity with the user’s query are identified. Here the value of V is also a hyperparameter and is fixed at a value of 15 for the Assignment. These documents are considered the most relevant matches to the query. If a document has a similarity score of zero, it is typically excluded from the results, as it does not match the query’s content. The document numbers (since documents are only the sentences in the documents.json file) along with their similarity measures are returned as the final result.



```
For Query No. 0, 15 most relevant documents are:  [3573, 1896, 2749, 5946, 6167, 1688, 4409, 1407, 4910, 2237, 1941, 5823, 1753, 126, 5041]
For Query No. 1, 15 most relevant documents are:  [3283, 269, 4750, 2491, 5533, 4227, 2400, 5407, 3129, 3998, 1378, 4297, 6082, 2399, 4614]
For Query No. 2, 15 most relevant documents are:  [3283, 269, 4750, 2491, 5533, 4227, 2400, 5407, 3129, 3998, 1378, 4297, 6082, 2399, 4614]
For Query No. 3, 15 most relevant documents are:  [3283, 269, 4750, 2491, 5533, 4227, 2400, 5407, 3129, 3998, 1378, 4297, 6082, 2399, 4614]
For Query No. 4, 15 most relevant documents are:  [5480, 762, 2680, 1002, 4981, 3608, 761, 1020, 4983, 163, 4339, 2688, 2139, 2546, 2520]
For Query No. 5, 15 most relevant documents are:  [3283, 269, 4750, 2491, 5533, 4227, 2400, 5407, 3129, 3998, 1378, 4297, 6082, 2399, 4614]
For Query No. 6, 15 most relevant documents are:  [3177, 2434, 4727, 2437, 5418, 410, 995, 4108, 2165, 1038, 3231, 5128, 4646, 4079, 2301]
For Query No. 7, 15 most relevant documents are:  [3283, 269, 4750, 2491, 5533, 4227, 2400, 5407, 3129, 3998, 1378, 4297, 6082, 2399, 4614]
For Query No. 8, 15 most relevant documents are:  [3283, 269, 4750, 2491, 5533, 4227, 2400, 5407, 3129, 3998, 1378, 4297, 6082, 2399, 4614]
For Query No. 9, 15 most relevant documents are:  [4733, 4734, 800, 4732, 2071, 2420, 1127, 1036, 3696, 2513, 4750, 5252, 4217, 798, 1186]
For Query No. 10, 15 most relevant documents are:  [269, 1726, 357, 5173, 5174, 1036, 6020, 1227, 5376, 788, 3032, 5983, 2840, 5957, 4613]
For Query No. 11, 15 most relevant documents are:  [4733, 4734, 800, 4732, 2071, 2420, 1127, 1036, 3696, 2513, 4750, 5252, 4217, 798, 1186]
For Query No. 12, 15 most relevant documents are:  [3283, 269, 4750, 2491, 5533, 4227, 2400, 5407, 3129, 3998, 1378, 4297, 6082, 2399, 4614]
For Query No. 13, 15 most relevant documents are:  [6461, 6460, 6459, 6458, 6457, 6456, 6455, 6454, 6453, 6452, 6451, 6450, 6449, 6448, 6447]
For Query No. 14, 15 most relevant documents are:  [269, 1726, 357, 5173, 5174, 1036, 6020, 1227, 5376, 788, 3032, 5983, 2840, 5957, 4613]
```

Top 15 Documents retrieved for the First 15 queries in the corpus with Stopwords removal



```
For Query No. 0, 15 most relevant documents are:  [209, 427, 5912, 905, 840, 1135, 6020, 4115, 5523, 2713, 2368, 897, 4732, 4238, 2198]
For Query No. 1, 15 most relevant documents are:  [1729, 800, 4816, 3970, 752, 798, 1305, 1066, 4914, 5273, 5169, 1829, 1359, 1730, 5171]
For Query No. 2, 15 most relevant documents are:  [4734, 1425, 1194, 1874, 4295, 281, 5002, 3301, 6282, 1395, 40, 736, 2281, 97, 1417]
For Query No. 3, 15 most relevant documents are:  [2491, 3283, 5533, 6082, 4237, 5407, 2400, 5467, 4750, 6162, 2399, 1378, 837, 5271, 5169]
For Query No. 4, 15 most relevant documents are:  [427, 1036, 4966, 4734, 3897, 1665, 1414, 4969, 3603, 634, 4820, 5698, 2183, 4109, 5937]
For Query No. 5, 15 most relevant documents are:  [427, 2491, 3283, 1036, 4734, 4614, 5533, 1378, 6082, 4733, 5467, 2400, 2399, 4237, 5002]
For Query No. 6, 15 most relevant documents are:  [1038, 3470, 5273, 4115, 69, 2966, 663, 1515, 5469, 2205, 1313, 598, 4042, 1872, 3845]
For Query No. 7, 15 most relevant documents are:  [427, 2491, 3283, 1036, 4734, 4614, 5533, 1378, 6082, 4733, 5467, 2400, 2399, 4237, 5002]
For Query No. 8, 15 most relevant documents are:  [4628, 3533, 1231, 437, 3707, 1370, 1122, 1421, 4985, 4631, 5295, 6225, 2050, 1435, 6282]
For Query No. 9, 15 most relevant documents are:  [427, 1036, 4734, 4966, 3897, 1665, 1414, 634, 4733, 4969, 3603, 5698, 4820, 2183, 1668]
For Query No. 10, 15 most relevant documents are:  [2491, 3283, 5533, 6082, 4237, 4750, 837, 2399, 5467, 4667, 5271, 5407, 2395, 2400, 4297]
For Query No. 11, 15 most relevant documents are:  [4733, 4734, 800, 2420, 4732, 798, 2713, 1036, 5974, 2418, 5469, 4476, 2071, 856, 5169]
For Query No. 12, 15 most relevant documents are:  [2491, 3283, 5533, 6082, 4237, 5407, 2400, 5467, 4750, 6162, 2399, 1378, 837, 5271, 5169]
For Query No. 13, 15 most relevant documents are:  [3533, 3707, 1231, 437, 1122, 4628, 5295, 1370, 6225, 1853, 6242, 4631, 5748, 3653, 6282]
For Query No. 14, 15 most relevant documents are:  [427, 2491, 3283, 1036, 4734, 4614, 5533, 1378, 6082, 4733, 5467, 2400, 5002, 4237, 5271]
```

Top 15 Documents retrieved for the First 15 queries in the corpus without Stopwords removal

The Probabilistic Model in Information Retrieval (IR) offers several advantages, especially when applied to tasks such as document-query similarity and ranking. Some key advantages are:

1. Relevance-based Ranking:

- The model focuses on relevance by calculating the probability that a document is relevant to a query. This allows the system to rank documents in order of their likelihood to be relevant, leading to more accurate search results.

2. Bayesian Foundation:

- The probabilistic model is based on Bayesian inference, allowing the incorporation of prior knowledge (such as term frequencies) and continuous updates of probabilities as new data or iterations come in. This provides a theoretically sound foundation for document ranking.

3. Dynamic Adaptation:

- As demonstrated in the iterative refinement process, probabilistic models can adapt dynamically through iterative updates. This allows the system to improve its relevance predictions based on feedback, making it flexible and improving results over time.

4. Term Independence Assumption:

- While simplifying assumptions (e.g., that terms are independent) are made in probabilistic models, this allows the model to remain computationally efficient. Despite this simplification, the model can still yield highly relevant results.

In summary, the probabilistic model is advantageous for IR tasks due to its relevance-based ranking, adaptability, term-independence assumption for simplifying calculations, and efficiency in handling large datasets.

Latent Semantic Indexing (LSI) Model:

The LatentSemanticIndexing (LSI) class applies **Latent Semantic Analysis (LSA)** to documents and queries using Singular Value Decomposition (SVD) for dimensionality reduction, allowing it to capture latent relationships between words and documents. Here's a brief summary of the main steps involved:

1. Initialization:

- The constructor initializes key variables like document/query preprocessing, their TF-IDF values, word occurrence probabilities, and similarity scores between documents and queries. These values are already stored in the form of an object which is an instance of the above defined Preprocessing class.
- The parameter p determines the number of most frequent word stems in the corpus to consider, and the number of refinement iterations is defined as the value 25.
- The parameter V determines the number of relevant documents to be considered for each query given as input by the user, whose value is set as 15 for the assignment.

2. P-stems:

- Following the above, we find the top p stems which are occurring in the dataset depending upon the cumulative frequency if each term in the corpus.

3. Forming Joint TF-IDF Matrix for Documents and Queries:

- This method constructs a **TF-IDF matrix** for both documents and queries, using the Top P-stems found above to build a reduced representation of the data.

4. SVD Factorisation:

- This method performs Singular Value Decomposition (SVD) on the TF-IDF matrix, reducing its dimensionality to k latent factors. It reconstructs a lower-dimensional approximation (M_k) of the original matrix.

$$M = S \cdot \Delta \cdot D^T.$$

$$M_l = S_l \cdot \Delta_l \cdot D_l^T$$

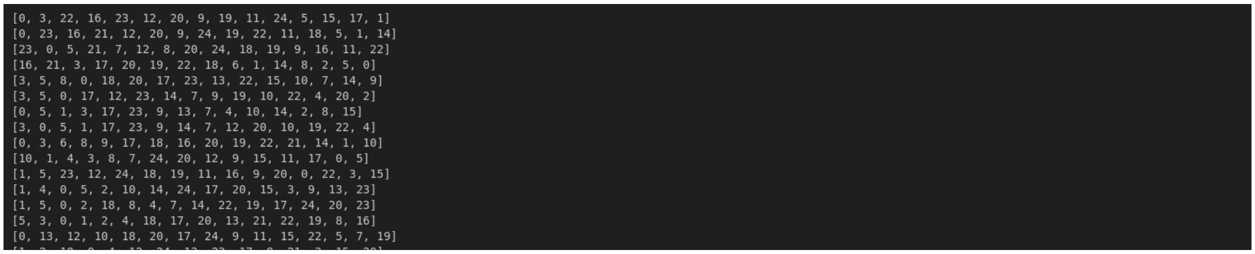
where M_l is the reconstructed matrix in the reduced space.

5. Similarity:

- This method computes the similarity between queries and documents in the **reduced space** (after SVD). It returns the top V most similar documents for a given query by sorting their similarity scores. The similarity between documents i and j is obtained by considering the (i, j)th term in the reconstructed matrix in the reduced dimension space. It is due to the reason that the resultant matrix was formed by the matrix multiplication $x^1.x^T$ which effectively takes the cosine similarity between the ith and jth documents.

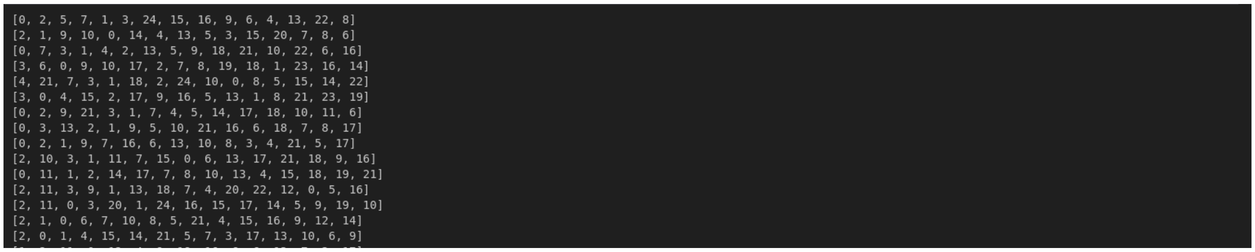
6. Fitting of Model:

- The main pipeline, where the model first identifies the most frequent word stems, constructs the TF-IDF matrix, and then applies SVD to reduce the matrix to the latent semantic space.



```
[0, 3, 22, 16, 23, 12, 20, 9, 19, 11, 24, 5, 15, 17, 1]
[0, 23, 16, 21, 12, 20, 9, 24, 19, 22, 11, 18, 5, 1, 14]
[23, 0, 5, 21, 7, 12, 8, 20, 24, 18, 19, 9, 16, 11, 22]
[16, 21, 3, 17, 20, 19, 22, 18, 6, 1, 14, 8, 2, 5, 0]
[3, 5, 8, 0, 18, 20, 17, 23, 13, 22, 15, 10, 7, 14, 9]
[3, 5, 0, 17, 12, 23, 14, 7, 9, 19, 10, 22, 4, 20, 2]
[0, 5, 1, 3, 17, 23, 9, 13, 7, 4, 10, 14, 2, 8, 15]
[3, 0, 5, 1, 17, 23, 9, 14, 7, 12, 20, 10, 19, 22, 4]
[0, 3, 6, 8, 9, 17, 18, 16, 20, 19, 22, 21, 14, 1, 10]
[10, 1, 4, 3, 8, 7, 24, 20, 12, 9, 15, 11, 17, 0, 5]
[1, 5, 23, 12, 24, 18, 19, 11, 16, 9, 20, 0, 22, 3, 15]
[1, 4, 0, 5, 2, 10, 14, 24, 17, 20, 15, 3, 9, 13, 23]
[1, 5, 0, 2, 18, 8, 4, 7, 14, 22, 19, 17, 24, 20, 23]
[5, 3, 0, 1, 2, 4, 18, 17, 20, 13, 21, 22, 19, 8, 16]
[0, 13, 12, 10, 18, 20, 17, 24, 9, 11, 15, 22, 5, 7, 19]
```

Top 15 Documents retrieved for the First 15 queries in the corpus with Stopwords removal



```
[0, 2, 5, 7, 1, 3, 24, 15, 16, 9, 6, 4, 13, 22, 8]
[2, 1, 9, 10, 0, 14, 4, 13, 5, 3, 15, 20, 7, 8, 6]
[0, 7, 3, 1, 4, 2, 13, 5, 9, 18, 21, 10, 22, 6, 16]
[3, 6, 0, 9, 10, 17, 2, 7, 8, 19, 18, 1, 23, 16, 14]
[4, 21, 7, 3, 1, 18, 2, 24, 10, 0, 8, 5, 15, 14, 22]
[3, 0, 4, 15, 2, 17, 9, 16, 5, 13, 1, 8, 21, 23, 19]
[0, 2, 9, 21, 3, 1, 7, 4, 5, 14, 17, 18, 10, 11, 6]
[0, 3, 13, 2, 1, 9, 5, 10, 21, 16, 6, 18, 7, 8, 17]
[0, 2, 1, 9, 7, 16, 6, 13, 10, 8, 3, 4, 21, 5, 17]
[2, 10, 3, 1, 11, 7, 15, 0, 6, 13, 17, 21, 18, 9, 16]
[0, 11, 1, 2, 14, 17, 7, 8, 10, 13, 4, 15, 18, 19, 21]
[2, 11, 3, 9, 1, 13, 18, 7, 4, 20, 22, 12, 0, 5, 16]
[2, 11, 0, 3, 20, 1, 24, 16, 15, 17, 14, 5, 9, 19, 10]
[2, 1, 0, 6, 7, 10, 8, 5, 21, 4, 15, 16, 9, 12, 14]
[2, 0, 1, 4, 15, 14, 21, 5, 7, 3, 17, 13, 10, 6, 9]
```

Top 15 Documents retrieved for the First 15 queries in the corpus without Stopwords removal

LSI Model in Information Retrieval (IR) provides many advantages:

1. Captures Synonymy and Latent Relationships:

- LSI identifies latent structures in the data, allowing it to recognize synonyms and related terms that may not be explicitly present in the query or document.

This helps improve retrieval by matching documents to queries even when they use different vocabulary to describe the same concept.

2. Reduces Noise and Sparsity:

- By performing Singular Value Decomposition (SVD), LSI reduces the high-dimensional space of terms to a lower-dimensional latent space. This helps in reducing noise and sparsity that typically arise from infrequently occurring words or noise in the data, leading to more robust document representations.

3. Semantic Generalization:

- Instead of relying on exact word matches, LSI captures the **underlying meaning** or semantics of documents. This enables the model to retrieve documents that may be relevant based on conceptual similarity, not just lexical overlap.

4. Handles Polysemy:

- LSI can also handle polysemy, where a word has multiple meanings, by placing documents with similar meanings in the same latent space. The dimensionality reduction allows the model to disambiguate word meanings based on context.

In summary, the LSI model is advantageous for IR tasks due to synonymity capturing, noise and sparsity reduction, semantic generalization, polysemy and synonymy which leads to better capturing of results.

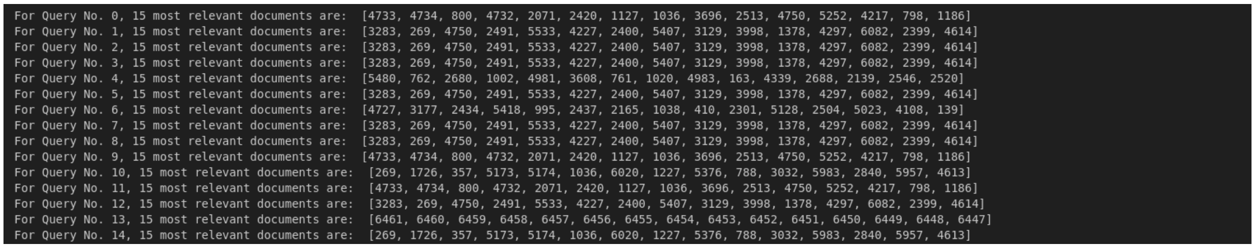
Comparison:

We can see that first of all the documents which are retrieved are different when stopwords are removed, it is because stopwords occupy the top p-stems otherwise, which hampers the overall learning of the model, since now the assumption is that top p-stems convey the most information in the model, which in the first case are stopwords, which by themselves do not convey any meaning and in the second case are different words in the corpus. This is the reason that in queries involving no stopwords (Query No. 2) in the figure, same documents are retrieved, since stopwords do not affect the retrieval process now.

Also, one more thing to note is that the documents retrieved in case of LSI Models are entirely different! It is due to the way the algorithm proceeds for this case. Also, in this case, removing stopwords do not entirely change the retrieval process. This means that stopwords have lesser role to play in case of LSI Model, which concludes the fact that it is a stronger model than Probabilistic model since it is able to capture the features in the corpus to some extent, without even removing the stopwords from the corpus.

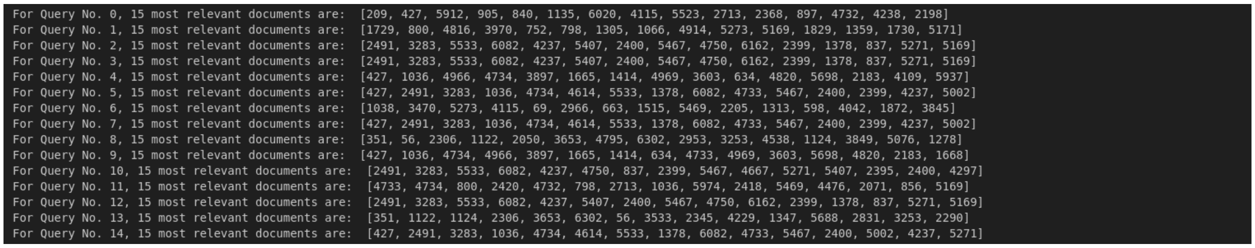
Task 5: Experimentation with Different Hyperparameters

We experiment by varying the value of p, which affects the top p-stems in the corpus and also vary the number of documents retrieved (V) and note the results. Documents retrieved are as follows (all results are shown for only the top 15 queries in the corpus):



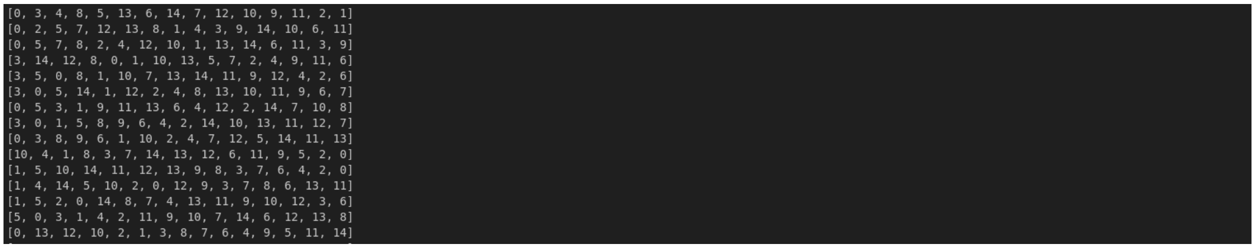
```
For Query No. 0, 15 most relevant documents are:  [4733, 4734, 800, 4732, 2071, 2420, 1127, 1036, 3696, 2513, 4750, 5252, 4217, 798, 1186]
For Query No. 1, 15 most relevant documents are:  [3283, 269, 4750, 2491, 5533, 4227, 2400, 5407, 3129, 3998, 1378, 4297, 6082, 2399, 4614]
For Query No. 2, 15 most relevant documents are:  [3283, 269, 4750, 2491, 5533, 4227, 2400, 5407, 3129, 3998, 1378, 4297, 6082, 2399, 4614]
For Query No. 3, 15 most relevant documents are:  [3283, 269, 4750, 2491, 5533, 4227, 2400, 5407, 3129, 3998, 1378, 4297, 6082, 2399, 4614]
For Query No. 4, 15 most relevant documents are:  [5480, 762, 2680, 1002, 4981, 3608, 761, 1020, 4983, 163, 4339, 2688, 2139, 2546, 2520]
For Query No. 5, 15 most relevant documents are:  [3283, 269, 4750, 2491, 5533, 4227, 2400, 5407, 3129, 3998, 1378, 4297, 6082, 2399, 4614]
For Query No. 6, 15 most relevant documents are:  [4727, 3177, 2434, 5418, 995, 2437, 2165, 1038, 410, 2301, 5128, 2504, 5023, 4108, 139]
For Query No. 7, 15 most relevant documents are:  [3283, 269, 4750, 2491, 5533, 4227, 2400, 5407, 3129, 3998, 1378, 4297, 6082, 2399, 4614]
For Query No. 8, 15 most relevant documents are:  [3283, 269, 4750, 2491, 5533, 4227, 2400, 5407, 3129, 3998, 1378, 4297, 6082, 2399, 4614]
For Query No. 9, 15 most relevant documents are:  [4733, 4734, 800, 4732, 2071, 2420, 1127, 1036, 3696, 2513, 4750, 5252, 4217, 798, 1186]
For Query No. 10, 15 most relevant documents are:  [269, 1726, 357, 5173, 5174, 1036, 6020, 1227, 5376, 788, 3032, 5983, 2840, 5957, 4613]
For Query No. 11, 15 most relevant documents are:  [4733, 4734, 800, 4732, 2071, 2420, 1127, 1036, 3696, 2513, 4750, 5252, 4217, 798, 1186]
For Query No. 12, 15 most relevant documents are:  [3283, 269, 4750, 2491, 5533, 4227, 2400, 5407, 3129, 3998, 1378, 4297, 6082, 2399, 4614]
For Query No. 13, 15 most relevant documents are:  [6461, 6460, 6459, 6458, 6457, 6456, 6455, 6454, 6453, 6452, 6451, 6450, 6449, 6448, 6447]
For Query No. 14, 15 most relevant documents are:  [269, 1726, 357, 5173, 5174, 1036, 6020, 1227, 5376, 788, 3032, 5983, 2840, 5957, 4613]
```

Documents retrieved using Probabilistic Model for p=15 and V=15



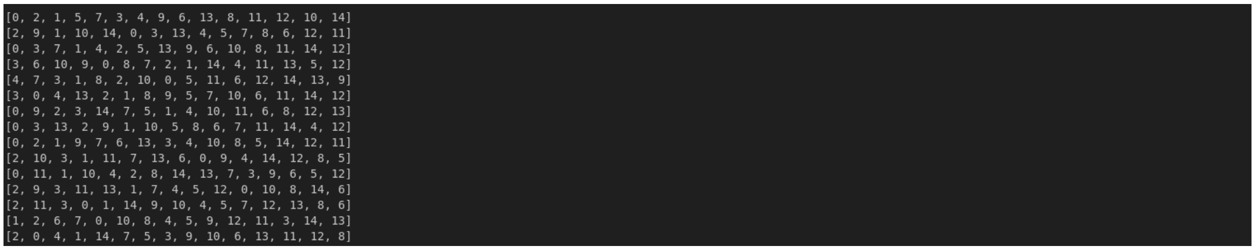
```
For Query No. 0, 15 most relevant documents are:  [209, 427, 5912, 905, 840, 1135, 6020, 4115, 5523, 2713, 2368, 897, 4732, 4238, 2198]
For Query No. 1, 15 most relevant documents are:  [1729, 800, 4816, 3970, 752, 798, 1305, 1066, 4914, 5273, 5169, 1829, 1359, 1730, 5171]
For Query No. 2, 15 most relevant documents are:  [2491, 3283, 5533, 6082, 4237, 5407, 2400, 5467, 4750, 6162, 2399, 1378, 837, 5271, 5169]
For Query No. 3, 15 most relevant documents are:  [2491, 3283, 5533, 6082, 4237, 5407, 2400, 5467, 4750, 6162, 2399, 1378, 837, 5271, 5169]
For Query No. 4, 15 most relevant documents are:  [427, 1036, 4966, 4734, 3897, 1665, 1414, 4969, 3603, 634, 4820, 5698, 2183, 4109, 5937]
For Query No. 5, 15 most relevant documents are:  [427, 2491, 3283, 1036, 4734, 4614, 5533, 1378, 6082, 4733, 5467, 2400, 2399, 4237, 5002]
For Query No. 6, 15 most relevant documents are:  [1038, 3470, 5273, 4115, 69, 2966, 663, 1515, 5469, 2205, 1313, 598, 4042, 1872, 3845]
For Query No. 7, 15 most relevant documents are:  [427, 2491, 3283, 1036, 4734, 4614, 5533, 1378, 6082, 4733, 5467, 2400, 2399, 4237, 5002]
For Query No. 8, 15 most relevant documents are:  [351, 56, 2306, 1122, 2050, 3653, 4795, 6302, 2953, 3253, 4538, 1124, 3849, 5076, 1278]
For Query No. 9, 15 most relevant documents are:  [427, 1036, 4734, 4966, 3897, 1665, 1414, 634, 4733, 4969, 3603, 5698, 4820, 2183, 1668]
For Query No. 10, 15 most relevant documents are:  [2491, 3283, 5533, 6082, 4237, 4750, 837, 2399, 5467, 4667, 5271, 5407, 2395, 2400, 4297]
For Query No. 11, 15 most relevant documents are:  [4733, 4734, 800, 2420, 4732, 798, 2713, 1036, 5974, 2418, 5469, 4476, 2071, 856, 5169]
For Query No. 12, 15 most relevant documents are:  [2491, 3283, 5533, 6082, 4237, 5407, 2400, 5467, 4750, 6162, 2399, 1378, 837, 5271, 5169]
For Query No. 13, 15 most relevant documents are:  [351, 1122, 1124, 2306, 3653, 6302, 56, 3533, 2345, 4229, 1347, 5688, 2831, 3253, 2290]
For Query No. 14, 15 most relevant documents are:  [427, 2491, 3283, 1036, 4734, 4614, 5533, 1378, 6082, 4733, 5467, 2400, 5002, 4237, 5271]
```

Documents retrieved using Probabilistic Model for p=15 and V=15 without Stopwords removal



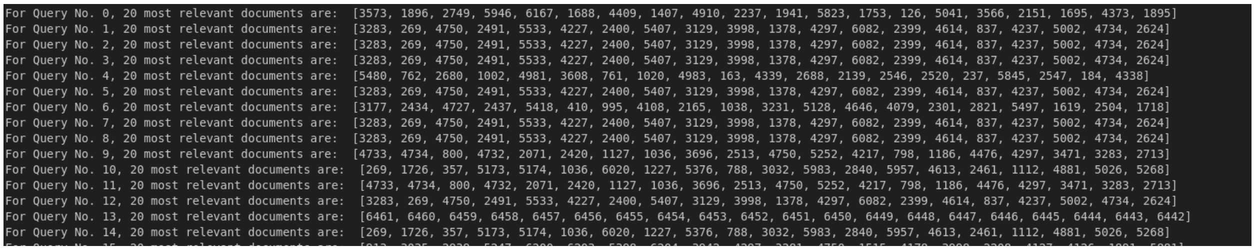
```
[0, 3, 4, 8, 5, 13, 6, 14, 7, 12, 10, 9, 11, 2, 1]
[0, 2, 5, 7, 12, 13, 8, 1, 4, 3, 9, 14, 10, 6, 11]
[0, 5, 7, 8, 2, 4, 12, 10, 1, 13, 14, 6, 11, 3, 9]
[3, 14, 12, 8, 0, 1, 10, 13, 5, 7, 2, 4, 9, 11, 6]
[3, 5, 0, 8, 1, 10, 7, 13, 14, 11, 9, 12, 4, 2, 6]
[3, 0, 5, 14, 1, 12, 2, 4, 8, 13, 10, 11, 9, 6, 7]
[0, 5, 3, 1, 9, 11, 13, 6, 4, 12, 2, 14, 7, 10, 8]
[3, 0, 1, 5, 8, 9, 6, 4, 2, 14, 10, 13, 11, 12, 7]
[0, 3, 8, 9, 6, 1, 10, 2, 4, 7, 12, 5, 14, 11, 13]
[10, 4, 1, 8, 3, 7, 14, 13, 12, 6, 11, 9, 5, 2, 0]
[1, 5, 10, 14, 11, 12, 13, 9, 8, 3, 7, 6, 4, 2, 0]
[1, 4, 14, 5, 10, 2, 0, 12, 9, 3, 7, 8, 6, 13, 11]
[1, 5, 2, 0, 14, 8, 7, 4, 13, 11, 9, 10, 12, 3, 6]
[5, 0, 3, 1, 4, 2, 11, 9, 10, 7, 14, 6, 12, 13, 8]
[0, 13, 12, 10, 2, 1, 3, 8, 7, 6, 4, 9, 5, 11, 14]
```

Documents retrieved using LSI Model for p=15 and V=15



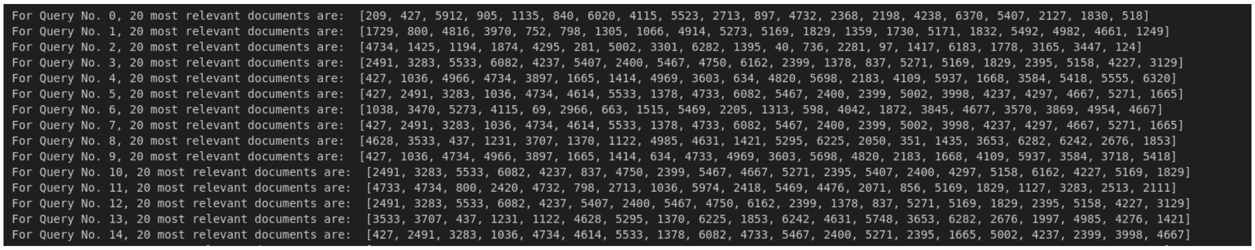
```
[0, 2, 1, 5, 7, 3, 4, 9, 6, 13, 8, 11, 12, 10, 14]
[2, 9, 1, 10, 14, 0, 3, 13, 4, 5, 7, 8, 6, 12, 11]
[0, 3, 7, 1, 4, 2, 5, 13, 9, 6, 10, 8, 11, 14, 12]
[3, 6, 10, 9, 0, 8, 7, 2, 1, 14, 4, 11, 13, 5, 12]
[4, 7, 3, 1, 8, 2, 10, 0, 5, 11, 6, 12, 14, 13, 9]
[3, 0, 4, 13, 2, 1, 8, 9, 5, 7, 10, 6, 11, 14, 12]
[0, 9, 2, 3, 14, 7, 5, 1, 4, 10, 11, 6, 8, 12, 13]
[0, 3, 13, 2, 9, 1, 10, 5, 8, 6, 7, 11, 14, 4, 12]
[0, 2, 1, 9, 7, 6, 13, 3, 4, 10, 8, 5, 14, 12, 11]
[2, 10, 3, 1, 11, 7, 13, 6, 0, 9, 4, 14, 12, 8, 5]
[0, 11, 1, 10, 4, 2, 8, 14, 13, 7, 3, 9, 6, 5, 12]
[2, 9, 3, 11, 13, 1, 7, 4, 5, 12, 0, 10, 8, 14, 6]
[2, 11, 3, 0, 1, 14, 9, 10, 4, 5, 7, 12, 13, 8, 6]
[1, 2, 6, 7, 0, 10, 8, 4, 5, 9, 12, 11, 3, 14, 13]
[2, 0, 4, 1, 14, 7, 5, 3, 9, 10, 6, 13, 11, 12, 8]
```

Documents retrieved using LSI Model for p=15 and V=15 without Stopwords removal



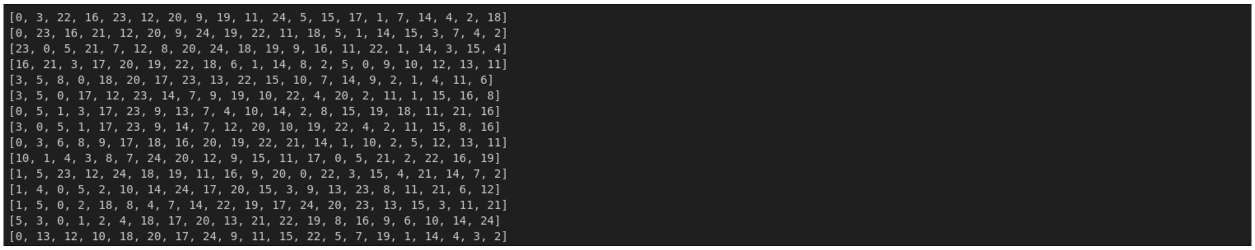
```
For Query No. 0, 20 most relevant documents are:  [3573, 1896, 2749, 5946, 6167, 1688, 4409, 1407, 4910, 2237, 1941, 5823, 1753, 126, 5041, 3566, 2151, 1695, 4373, 1895]
For Query No. 1, 20 most relevant documents are:  [3283, 269, 4750, 2491, 5533, 4227, 2400, 5407, 3129, 3998, 1378, 4297, 6082, 2399, 4614, 837, 4237, 5002, 4734, 2624]
For Query No. 2, 20 most relevant documents are:  [3283, 269, 4750, 2491, 5533, 4227, 2400, 5407, 3129, 3998, 1378, 4297, 6082, 2399, 4614, 837, 4237, 5002, 4734, 2624]
For Query No. 3, 20 most relevant documents are:  [3283, 269, 4750, 2491, 5533, 4227, 2400, 5407, 3129, 3998, 1378, 4297, 6082, 2399, 4614, 837, 4237, 5002, 4734, 2624]
For Query No. 4, 20 most relevant documents are:  [5480, 762, 2680, 1002, 4981, 3608, 761, 1020, 4983, 163, 4339, 2688, 2139, 2546, 2520, 237, 5845, 2547, 184, 4338]
For Query No. 5, 20 most relevant documents are:  [3283, 269, 4750, 2491, 5533, 4227, 2400, 5407, 3129, 3998, 1378, 4297, 6082, 2399, 4614, 837, 4237, 5002, 4734, 2624]
For Query No. 6, 20 most relevant documents are:  [3177, 2434, 4727, 2437, 5418, 410, 995, 4108, 2165, 1038, 3231, 5128, 4646, 4079, 2301, 2821, 5497, 1619, 2504, 1718]
For Query No. 7, 20 most relevant documents are:  [3283, 269, 4750, 2491, 5533, 4227, 2400, 5407, 3129, 3998, 1378, 4297, 6082, 2399, 4614, 837, 4237, 5002, 4734, 2624]
For Query No. 8, 20 most relevant documents are:  [3283, 269, 4750, 2491, 5533, 4227, 2400, 5407, 3129, 3998, 1378, 4297, 6082, 2399, 4614, 837, 4237, 5002, 4734, 2624]
For Query No. 9, 20 most relevant documents are:  [4733, 4734, 800, 4732, 2071, 2420, 1127, 1036, 3696, 2513, 4750, 5252, 4217, 798, 1186, 4476, 4297, 3471, 3283, 2713]
For Query No. 10, 20 most relevant documents are:  [269, 1726, 357, 5173, 5174, 1036, 6020, 1227, 5376, 788, 3032, 5983, 2840, 5957, 4613, 2461, 1112, 4881, 5026, 5268]
For Query No. 11, 20 most relevant documents are:  [4733, 4734, 800, 4732, 2071, 2420, 1127, 1036, 3696, 2513, 4750, 5252, 4217, 798, 1186, 4476, 4297, 3471, 3283, 2713]
For Query No. 12, 20 most relevant documents are:  [3283, 269, 4750, 2491, 5533, 4227, 2400, 5407, 3129, 3998, 1378, 4297, 6082, 2399, 4614, 837, 4237, 5002, 4734, 2624]
For Query No. 13, 20 most relevant documents are:  [6461, 6460, 6459, 6458, 6457, 6456, 6455, 6454, 6453, 6452, 6451, 6450, 6449, 6448, 6447, 6446, 6445, 6444, 6443, 6442]
For Query No. 14, 20 most relevant documents are:  [269, 1726, 357, 5173, 5174, 1036, 6020, 1227, 5376, 788, 3032, 5983, 2840, 5957, 4613, 2461, 1112, 4881, 5026, 5268]
```

Documents retrieved using Probabilistic Model for p=20 and V=25



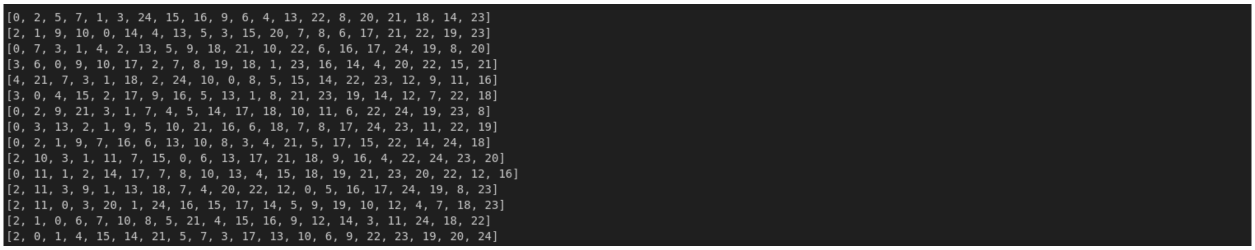
```
 For Query No. 0, 20 most relevant documents are:  [209, 427, 5912, 905, 1135, 840, 6020, 4115, 5523, 2713, 897, 4732, 2368, 2198, 4238, 6370, 5407, 2127, 1830, 518]
 For Query No. 1, 20 most relevant documents are:  [1729, 800, 4816, 3970, 752, 798, 1305, 1066, 4914, 5273, 5169, 1829, 1359, 1730, 5171, 1832, 5492, 4982, 4661, 1249]
 For Query No. 2, 20 most relevant documents are:  [4734, 1425, 1194, 1874, 4295, 281, 5002, 3301, 6282, 1395, 40, 736, 2281, 97, 1417, 6183, 1778, 3165, 3447, 124]
 For Query No. 3, 20 most relevant documents are:  [2491, 3283, 5533, 6082, 4237, 5407, 2400, 5467, 4750, 6162, 2399, 1378, 837, 5271, 5169, 1829, 2395, 5158, 4227, 3129]
 For Query No. 4, 20 most relevant documents are:  [427, 1036, 4966, 4734, 3897, 1665, 1414, 4969, 3603, 634, 4820, 5698, 2183, 4109, 5937, 1668, 3584, 5418, 5555, 6320]
 For Query No. 5, 20 most relevant documents are:  [427, 2491, 3283, 1036, 4734, 4614, 5533, 1378, 4733, 6082, 5467, 2400, 2399, 5002, 3998, 4237, 4297, 4667, 5271, 1665]
 For Query No. 6, 20 most relevant documents are:  [1038, 3470, 5273, 4115, 69, 2966, 663, 1515, 5469, 2205, 1313, 598, 4042, 1872, 3845, 4677, 3570, 3869, 4954, 4667]
 For Query No. 7, 20 most relevant documents are:  [427, 2491, 3283, 1036, 4734, 4614, 5533, 1378, 4733, 6082, 5467, 2400, 2399, 5002, 3998, 4237, 4297, 4667, 5271, 1665]
 For Query No. 8, 20 most relevant documents are:  [4628, 3533, 437, 1231, 3707, 1370, 1122, 4985, 4631, 1421, 5295, 6225, 2050, 351, 1435, 3653, 6282, 6242, 2676, 1853]
 For Query No. 9, 20 most relevant documents are:  [427, 1036, 4734, 4966, 3897, 1665, 1414, 634, 4733, 4969, 3603, 5698, 4820, 2183, 1668, 4109, 5937, 3584, 3718, 5418]
 For Query No. 10, 20 most relevant documents are:  [2491, 3283, 5533, 6082, 4237, 837, 4750, 2399, 5467, 4667, 5271, 2395, 5407, 2400, 4297, 5158, 6162, 4227, 5169, 1829]
 For Query No. 11, 20 most relevant documents are:  [4733, 4734, 800, 2420, 4732, 798, 2713, 1036, 5974, 2418, 5469, 4476, 2071, 856, 5169, 1829, 1127, 3283, 2513, 2111]
 For Query No. 12, 20 most relevant documents are:  [2491, 3283, 5533, 6082, 4237, 5407, 2400, 5467, 4750, 6162, 2399, 1378, 837, 5271, 5169, 1829, 2395, 5158, 4227, 3129]
 For Query No. 13, 20 most relevant documents are:  [3533, 3707, 437, 1231, 1122, 4628, 5295, 1370, 6225, 1853, 6242, 4631, 5748, 3653, 6282, 2676, 1997, 4985, 4276, 1421]
 For Query No. 14, 20 most relevant documents are:  [427, 2491, 3283, 1036, 4734, 4614, 5533, 1378, 6082, 4733, 5467, 2400, 5271, 2395, 1665, 5002, 4237, 2399, 3998, 4667]
```

Documents retrieved using Probabilistic Model for p=20 and V=25 without Stopwords removal



```
[0, 3, 22, 16, 23, 12, 20, 9, 19, 11, 24, 5, 15, 17, 1, 7, 14, 4, 2, 18]
[0, 23, 16, 21, 12, 20, 9, 24, 19, 22, 11, 18, 5, 1, 14, 15, 3, 7, 4, 2]
[23, 0, 5, 21, 7, 12, 8, 20, 24, 18, 19, 9, 16, 11, 22, 1, 14, 3, 15, 4]
[16, 21, 3, 17, 20, 19, 22, 18, 6, 1, 14, 8, 2, 5, 0, 9, 10, 12, 13, 11]
[3, 5, 8, 0, 18, 20, 17, 23, 13, 22, 15, 10, 7, 14, 9, 2, 1, 4, 11, 6]
[3, 5, 0, 17, 12, 23, 14, 7, 9, 19, 10, 22, 4, 20, 2, 11, 1, 15, 16, 8]
[0, 5, 1, 3, 17, 23, 9, 13, 7, 4, 10, 14, 2, 8, 15, 19, 18, 11, 21, 16]
[3, 0, 5, 1, 17, 23, 9, 14, 7, 12, 20, 10, 19, 22, 4, 2, 11, 15, 8, 16]
[0, 3, 6, 8, 9, 17, 18, 16, 20, 19, 22, 21, 14, 1, 10, 2, 5, 12, 13, 11]
[10, 1, 4, 3, 8, 7, 24, 20, 12, 9, 15, 11, 17, 0, 5, 21, 2, 22, 16, 19]
[1, 5, 23, 12, 24, 18, 19, 11, 16, 9, 20, 0, 22, 3, 15, 4, 21, 14, 7, 2]
[1, 4, 0, 5, 2, 10, 14, 24, 17, 20, 15, 3, 9, 13, 23, 8, 11, 21, 6, 12]
[1, 5, 0, 2, 18, 8, 4, 7, 14, 22, 19, 17, 24, 20, 23, 13, 15, 3, 11, 21]
[5, 3, 0, 1, 2, 4, 18, 17, 20, 13, 21, 22, 19, 8, 16, 9, 6, 10, 14, 24]
[0, 13, 12, 10, 18, 20, 17, 24, 9, 11, 15, 22, 5, 7, 19, 1, 14, 4, 3, 2]
```

Documents retrieved using LSI Model for p=20 and V=25



```
[0, 2, 5, 7, 1, 3, 24, 15, 16, 9, 6, 4, 13, 22, 8, 20, 21, 18, 14, 23]
[2, 1, 9, 10, 0, 14, 4, 13, 5, 3, 15, 20, 7, 8, 6, 17, 21, 22, 19, 23]
[0, 7, 3, 1, 4, 2, 13, 5, 9, 18, 21, 10, 22, 6, 16, 17, 24, 19, 8, 20]
[3, 6, 0, 9, 10, 17, 2, 7, 8, 19, 18, 1, 23, 16, 14, 4, 20, 22, 15, 21]
[4, 21, 7, 3, 1, 18, 2, 24, 10, 0, 8, 5, 15, 14, 22, 23, 12, 9, 11, 16]
[3, 0, 4, 15, 2, 17, 9, 16, 5, 13, 1, 8, 21, 23, 19, 14, 12, 7, 22, 18]
[0, 2, 9, 21, 3, 1, 7, 4, 5, 14, 17, 18, 10, 11, 6, 22, 24, 19, 23, 8]
[0, 3, 13, 2, 1, 9, 5, 10, 21, 16, 6, 18, 7, 8, 17, 24, 23, 11, 22, 19]
[0, 2, 1, 9, 7, 16, 6, 13, 10, 8, 3, 4, 21, 5, 17, 15, 22, 14, 24, 18]
[2, 10, 3, 1, 11, 7, 15, 0, 6, 13, 17, 21, 18, 9, 16, 4, 22, 24, 23, 20]
[0, 11, 1, 2, 14, 17, 7, 8, 10, 13, 4, 15, 18, 19, 21, 23, 20, 22, 12, 16]
[2, 11, 3, 9, 1, 13, 18, 7, 4, 20, 22, 12, 0, 5, 16, 17, 24, 19, 8, 23]
[2, 11, 0, 3, 20, 1, 24, 16, 15, 17, 14, 5, 9, 19, 10, 12, 4, 7, 18, 23]
[2, 1, 0, 6, 7, 10, 8, 5, 21, 4, 15, 16, 9, 12, 14, 3, 11, 24, 18, 22]
[2, 0, 1, 4, 15, 14, 21, 5, 7, 3, 17, 13, 10, 6, 9, 22, 23, 19, 20, 24]
```

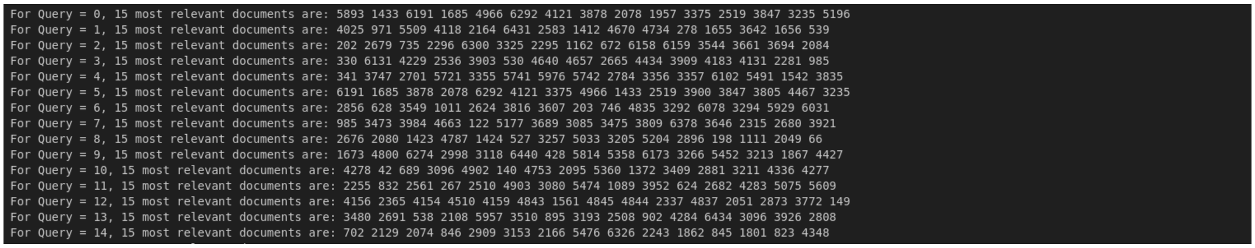
Documents retrieved using LSI Model for p=20 and V=25 without Stopwords removal

These findings lead us to the conclusion that in case of Probabilistic model, model is easily getting influenced by the consideration of greater/lower number of p-stems in the corpus. Because, just by varying the value of p(25 to 15), keeping V constant (15), entirely different set of documents are retrieved in case of probabilistic models, however in case of LSI Models, this difference is not that pronounced and more or less same documents are retrieved.

Also, the results suggest that varying V keeping p constant does not change the running of the algorithm in any way and same set of documents are retrieved, which makes sense, since V is the Hyperparameter which controls the number of documents to be retrieved, which is independent of the working of the algorithm. Increasing V simply leads to display of more documents and decreasing V has the opposite effect.

Task 6: Transformer-Based Model

In this part, the tf-idf matrix created in Task 2 are considered since they portray a better picture of information of the document. Data Preprocessing (without performing Stemming and without removing Stopwords) was done by calling the standard class for data preprocessing. Then, the Sentence Transformer all-MiniLM-L6-v2 was imported from the sentence_transformers library of python which allows us to use open-source models on HuggingFace for free. Following this, text was encoded in the format expected by the transformer by encode_text function of the library. Similarity was evaluated by computing the cosine similarity between each document and each query. Following this, documents were ranked in decreasing order of their similarities. We can see that the documents retrieved by the Transformer Model are very different from those retrieved by using the other 2 models. It is due to the fact that Transformers using the concept of Self-Attention and other Normalisation layers, is able to capture the context more effectively, which is not possible in the case of Probabilistic model and LSI models, since they rely on the assumption of capturing of features based upon only the top-p stems in the corpus, which is highly illogical, since rarer words express more information than most commonly occurring words, and should be kept, which is not the case with these models. Transformers, on the other hand, by applying Attention effectively captures the features involved in the text. So, we conclude that the application and effective usage of the above models is limited and Transformers are the go-to tool for NLP Tasks since they are independent of the limited assumptions of top p-occurring stems or stopwords in the corpus.



```
For Query = 0, 15 most relevant documents are: 5893 1433 6191 1685 4966 6292 4121 3878 2078 1957 3375 2519 3847 3235 5196
For Query = 1, 15 most relevant documents are: 4025 971 5509 4118 2164 6431 2583 1412 4670 4734 278 1655 3642 1656 539
For Query = 2, 15 most relevant documents are: 202 2679 735 2296 6300 3325 2295 1162 672 6158 6159 3544 3661 3694 2084
For Query = 3, 15 most relevant documents are: 330 6131 4229 2536 3903 530 4640 4657 2665 4434 3909 4183 4131 2281 985
For Query = 4, 15 most relevant documents are: 341 3747 2701 5721 3355 5741 5976 5742 2784 3356 3357 6102 5491 1542 3835
For Query = 5, 15 most relevant documents are: 6191 1685 3878 2078 6292 4121 3375 4966 1433 2519 3900 3847 3805 4467 3235
For Query = 6, 15 most relevant documents are: 2856 628 3549 1011 2624 3816 3607 203 746 4835 3292 6078 3294 5929 6031
For Query = 7, 15 most relevant documents are: 985 3473 3984 4663 122 5177 3689 3085 3475 3809 6378 3646 2315 2680 3921
For Query = 8, 15 most relevant documents are: 2676 2080 1423 4787 1424 527 3257 5033 3205 5204 2896 198 1111 2049 66
For Query = 9, 15 most relevant documents are: 1673 4800 6274 2998 3118 6440 428 5814 5358 6173 3266 5452 3213 1867 4427
For Query = 10, 15 most relevant documents are: 4278 42 689 3096 4902 140 4753 2095 5360 1372 3409 2881 3211 4336 4277
For Query = 11, 15 most relevant documents are: 2255 832 2561 267 2510 4903 3080 5474 1089 3952 624 2682 4283 5075 5609
For Query = 12, 15 most relevant documents are: 4156 2365 4154 4510 4159 4843 1561 4845 4844 2337 4837 2051 2873 3772 149
For Query = 13, 15 most relevant documents are: 3480 2691 538 2108 5957 3510 895 3193 2508 902 4284 6434 3096 3926 2808
For Query = 14, 15 most relevant documents are: 702 2129 2074 846 2909 3153 2166 5476 6326 2243 1862 845 1801 823 4348
```

Top 15 Documents retrieved for the First 15 queries in the corpus