A report on

RANKED INFORMATION RETRIEVAL SYSTEM

Submitted by

Ayush Singh	**2018A7PS0274P**
Himanshu Pandey	**2018A7PS0196P**
Mohul Maheshwari	**2018A7PS0229P**
Naman Goenka	**2018A7PS0398P**
Prakhar Mishra	**2018A7PS0257P**

In the partial fulfilment of the course

Information Retrieval

(CS F469)

As

Project Assignment

April-May 2021

Date: 4th May 2021

Acknowledgement

This project could not have been possible without the guidance and support of all the people who guided and supported us. Thus, we would like to express our sincere gratitude to all of them.

A special thanks to Prof. Sudeept Mohan, Head of Department, Computer Science and Information Systems, BITS Pilani for providing the opportunity to conduct a study on this subject. We would like to acknowledge Dr. Vinti Agarwal, Assistant Professor, Computer Science and Information Systems who served as a Instructor-in-charge in the project and guided us in formulating the topic and each and every step on the journey thereafter.

Finally, we express our gratitude to our friends and family for the moral support that they consistently provided over the course of this project.

Abstract

This document is a report for the assignment “Ranked Information Retrieval System”, where we were tasked with creating a vector-space based information retrieval system, and attempt to improve the model with some ideas of our own. This document explains our attempt at the same.

Working on this project helped us gain a better understanding of how IR systems work, and also enabled us to tackle some practical problems faced during building such a system.

We have prepared a vector-space based IR system. The system indexed a collection of unstructured documents obtained from English Wikipedia. The system was developed in three parts.

In the first part, we developed a ranked retrieval based information retrieval based on the vector-space model. The queries to the system are free-text queries. After this system was built, we evaluated its performance on multi-term queries.

In the second part, we have improved the retrieval and ranking for the documents by adding support for zoned indexes and champion lists.

In the third part, we have filtered the query vector on the basis of a threshold idf value. We calculated the document count i.e. how many terms from the query vector are present in each document, then we have populated the selected document list on basis of document score until we have the required number of documents and finally sort the selected documents on basis of document score calculated with filtered query vector.

We present the results of 10 multi term queries on basic vector space model covering various aspects, which also describes improvements and limitations of the two improved vector space models.

Introduction

Information retrieval (**IR**) is the process of obtaining information system resources that are relevant to an information need from a collection of those resources. Searches can be based on full-text or other content-based indexing. Automated information retrieval systems are used to reduce what has been called information overload. An IR system is a software system that provides access to books, journals and other documents; stores and manages those documents. Web search engines are the most profound IR applications.

In the case of large document collections, the resulting number of matching documents can far exceed the number a human user could possibly sift through. Accordingly, it is essential for a search engine to rank-order the documents matching a query. To do this, the search engine computes, for each matching document, a score with respect to the query at hand. This type of scoring is also referred to as **ranked retrieval**.

Boolean Model or BIR is a simple baseline query model where each query follows the underlying principles of relational algebra with algebraic expressions and where documents are not fetched unless they completely match with each other. Since the query is either fetch the document (1) or doesn't fetch the document (0), there is no methodology to rank them.

The **Vector Space Model** has vectors of index items each assigned with weights. The weights range from positive (if matched completely or to some extent) to negative (if unmatched or completely oppositely matched) if documents are present. Term Frequency - Inverse Document Frequency (tf-idf) is one of the most popular techniques where weights are terms (e.g. words, keywords, phrases etc.) and dimensions is number of words inside the corpus.

The **similarity score between query and document** can be found by calculating cosine value between query weight vector and document weight vector using cosine similarity. Desired documents can be fetched by ranking them according to similarity score and fetched top k documents which have the highest scores or are most relevant to query vectors.

The size of document vectors in vector space models are very large due to the huge size of the corpus. Therefore to retrieve the results efficiently, we resort to building **champion lists**, which is to select in advance N documents with the highest score for each term in the dictionary, where N is decided in advance.

Since ranking function is only a proxy, getting a list of documents close to top K documents given by ranking function measure should suffice especially when we can improve efficiency of the system. **Eliminating the index** can be achieved by considering high idf terms in query vector and prioritizing documents having many query terms.

Methodology

Data/Corpus:

English Wikipedia documents were made available to us on a Google Drive Link.

For this assignment, we used files 'wiki_37' and 'wiki_64' to build our index and run queries for retrieval of matching documents.

Each document present in the Corpus has the following structure:

> start tag : < doc id = "Document id" url = "Wikipedia URL" title = "Document title" >
> end tag :< /doc >

The text between start and end text is the content corresponding to each document.

Libraries/Frameworks Used:

We used Python for designing our retrieval system. Python is a high level interpreted language. Its object oriented features help write scalable and readable code with ease. The various Python libraries we used are:

- re (regular expression library)
- operator
- time
- BeautifulSoup
- os
- codecs
- pickle
- numpy
- math
- string

Vocabulary:

Total 8784 documents were parsed. The final vocabulary of the system consists of 103407 terms.

Pre-processing:

For preprocessing we followed the following steps:

- All hyperlinks marked by string starting from 'http' were removed.

- All forms of whitespace such as '\n', '\t', were converted to a single space. Afterwards, any occurrence of more than one whitespace was removed.
- The 'punctuation' set of the String library in python was used as a delimiter for tokenization.
- Converted all alphabetic characters to their lowercase characters
- Each resulting token was preprocessed such that only alphanumeric sequence within that token was retained.

At the end of these steps, we get a list of all the terms present in the documents of our corpus, whose set is the vocabulary of the corpus.. **We have not performed any other techniques such as stop word removal, stemming and lemmatization**. **Same preprocessing function was used all over the project for preprocessing free text**. Final preprocessed query tokens of an input query can be seen while running test_queries.py file.

Implementation of the basic vector space model:

For the first part of our assignment, we implemented a primitive vector space model for ranked retrieval. This implementation was divided into the following steps:

- Steps completed before query processing:
  - Parsing of the document and creation of the inverted index. For each term in the vocabulary, the inverted index stores the document frequency and the posting list.
  - The individual posting lists are populated.
- Steps completed during query processing:
  - Parsing of the query and creation of query term-frequency index.
  - Calculation of the inverse document frequency (idf) scores for all the query terms. For this assignment, we were asked to use the lnc.ltc scoring scheme based on the SMART notation.
  - Calculation of the term frequency (tf) scores for all the query terms for each document in the inverted index.
  - Calculation of the final tf-idf scores for each relevant document, i.e., documents present in posting lists of at least one of the query terms.
  - Normalization of scores.

Computation Details:

N = Number of documents
$tf_{t,d}$ = Number of times the term t appears in document d
$tf_{t,q}$ = Number of times the term t appears in query q
df_t = Number of documents the term t appears in

For the term frequency weight calculation in the correct format for the documents, we used

- $w_{t,d} = (1 + \log tf_{t,d})$ if $tf_{t,d} > 0$ and 0 otherwise

For the term frequency weight calculation in the correct format for the queries, we used

- $w_{t,q} = (1 + \log tf_{t,q})$ if $tf_{t,q} > 0$ and 0 otherwise

For the inverse document frequency calculation in the correct format for the documents, we used

- $\text{idf}_t = 1$

For the inverse document frequency calculation in the correct format for the queries, we used

- $\text{idf}_t = \log_{10}(\text{N}/\text{df}_t)$

For the tf.idf score calculation in the correct format for the documents, we used

- $\text{score}_{t,d} = \text{tf}_{t,d}.\text{idf}_t = \text{w}_{t,d}.1 = \text{w}_{t,d}$
- $\boldsymbol{d} = \text{score}(d) = \sum_{t \in q \cap d} \text{score}_{t,d}$

For the tf.idf score calculation in the correct format for the queries, we used

- $\text{score}_{t,q} = \text{tf}_{t,q}.\text{idf}_t = \text{w}_{t,q}.\log_{10}(\text{N}/\text{df}_t)$
- $\boldsymbol{q} = \text{score}(q) = \sum_{t \in q} \text{score}_{t,q}$

The dimensions of the vector space are the query terms ($t \in q$).
The values of $\text{score}_{t,d}$ and $\text{score}_{t,q}$ are vectors in a specific t dimension.
For a vector space with n distinct query terms, the summation in final score calculation yields an n-dimensional vector.

After the respective vectors are created for the documents and the queries, the cosine similarity scores are calculated to rank the documents based on the query using

$$\cos(\vec{q},\vec{d}) = \frac{\vec{q} \bullet \vec{d}}{|\vec{q}||\vec{d}|} = \frac{\vec{q}}{|\vec{q}|} \bullet \frac{\vec{d}}{|\vec{d}|} = \frac{\sum_{i=1}^{|V|} q_i d_i}{\sqrt{\sum_{i=1}^{|V|} q_i^2} \sqrt{\sum_{i=1}^{|V|} d_i^2}}$$

Implementation of Improved Model 1:

As part of the first improvement, we also took into account the terms found in the titles of each document while creating the index so as to give weightage to the title as well, which can help us fetch more relevant documents as compared to the basic vector space model. Incorporation of champion list and zoned indexes are instrumental in improving efficiency as well as relevance.

- Steps completed before query processing:
  - Parsing of the documents and their titles and creation of a combined inverted index , where the inverted indexes store the document frequency as well as a weighted average of the term frequency of the term appearing in the title as well as the content of a document. The weighted average was decided to be computed as tf_weighted = 0.6*title tf + 0.4* content tf. The weighted tf is stored as float to save space, and is hence not normalised which requires it to be stored as double.
  - Sorting of the posting lists of each term in the index and selection of top R=100 documents for each term (champion list) is performed.

- We are also storing a normalization constant corresponding to each document in a new file with docid as key and the normalization constant as the value, so that we do not have to calculate the denominator for each document using cosine normalization during query retrieval to improve the speed.

- Steps completed during query processing:
  - Calculation of the term frequency (tf) scores for all the query terms for each document in the champion lists of the query terms.
  - Calculation of the final tf-idf scores for each relevant document, i.e., documents present in champion lists of at least one of the query terms.
  - Query normalization of scores.

Computation Details:

N = Number of documents

$tf_{t,d}$ = Number of times the term t appears in document d as a weighted average of appearance in the title and the document

$tf_{t,q}$ = Number of times the term t appears in query q

df_t = Number of documents the term t appears in

For the term frequency weight calculation in the correct format for the documents, we used

- $w_{t,d} = (1 + \log tf_{t,d})$ if $tf_{t,d} > 0$ and 0 otherwise

For the term frequency weight calculation in the correct format for the queries, we used

- $w_{t,q} = (1 + \log tf_{t,q})$ if $tf_{t,q} > 0$ and 0 otherwise

For the inverse document frequency calculation in the correct format for the documents, we used

- $idf_t = 1$

For the inverse document frequency calculation in the correct format for the queries, we used

- $idf_t = \log_{10} (N/df_t)$

For the tf.idf score calculation in the correct format for the documents, we used

- $score_{t,d} = tf_{t,d}.idf_t = w_{t,d}.1 = w_{t,d}$
- $\boldsymbol{d} = score(d) = \sum_{t \in q \cap d} score_{t,d}$

For the tf.idf score calculation in the correct format for the queries, we used

- $score_{t,q} = tf_{t,q}.idf_t = w_{t,q}.\log_{10} (N/df_t)$
- $\boldsymbol{q} = score(q) = \sum_{t \in q} score_{t,q}$

The dimensions of the vector space are the query terms ($t \in q$).

The values of $score_{t,d}$ and $score_{t,q}$ are vectors in a specific t dimension.

For a vector space with n distinct query terms, the summation in final score calculation yields an n-dimensional vector.

After the respective vectors are created for the documents and the queries, the cosine similarity scores are calculated to rank the documents based on the query using

$$\cos(\vec{q},\vec{d}) = \frac{\vec{q} \bullet \vec{d}}{\left|\vec{q}\right|\left|\vec{d}\right|} = \frac{\vec{q}}{\left|\vec{q}\right|} \bullet \frac{\vec{d}}{\left|\vec{d}\right|} = \frac{\sum_{i=1}^{|V|} q_i d_i}{\sqrt{\sum_{i=1}^{|V|} q_i^2} \sqrt{\sum_{i=1}^{|V|} d_i^2}}$$

**$|d|^2$ has been already computed and stored in our normalization index therefore can be directly read instead of being calculated for every query.

Implementation of Improved Model 2:

As part of the second improvement of the model, we took into account the fact that some query words in a multi-term query might not be relevant to the retrieval, and might inflate the scores of non relevant documents, leading to low precision.

- Steps before query processing:
  - Parsing of the document and creation of the inverted index. For each term in the vocabulary, the inverted index stores the document frequency and the posting list.
  - The individual posting lists are sorted in ascending order of the document IDs.
- Steps during query processing
  - Building the query vector, with a modification such that query terms with idf less than a fixed value are not included.
  - Creating a new list of documents depending on the count of terms matched in the document.
  - Calculating the lnc.ltc scores of each document in the new list.
  - Sorting the list based on the document scores.
  - Normalization of scores.

Computation details:

The scores and ranks in the improved vector space model are computed in a similar way as the basic vector space model with a few additional steps that modify the query.

N = Number of documents
df_t=Number of documents the term t appears
MINIDF = 1.3 = Threshold IDF
Query_idf = vector containing idf values of all the terms in a query
Query_tokens = modified query vector
Scores_count = List having docid as index and count of terms matching as value
Scores_count_doc = Contains scores_count in decreasing order
Docs = contains list of documents in decreasing order of scores_count

NO_DOCS_REQD = Required number of documents to be fetched

Only those terms in a query which have an idf value > 1.3 (threshold value) are considered. Threshold values have been found after working on different values and coming to a conclusion that 1.3 gives the best result.

For selecting only those terms which have an idf value greater than certain threshold, we used

- val = [True if idf>MINIDF else False for idf in query_idf.values()]

For removing the terms which do not have threshold idf we used,

- If query_idf[token] <= MINIDF: query_tokens.remove(token)

While calculating the document score, we are also storing the count of query terms matched in each document.Suppose our modified query vector contains n terms. Now, we will fetch those documents which contain all the n terms from the query, if this does not fetch enough documents to meet our requirements, we will fetch documents which match n-1 terms, n-2 terms and so on until we have enough documents. Now, we sort these documents in decreasing order of score and return them.

For every document in a term's posting list, in addition to we increment the document count by 1 and repeat this for all query terms

- scores_count[docid] = scores_count[docid] + 1

For adding the documents in order of decreasing score count, we used

- docs = [x[0] for x in sorted_count_doc]
  return docs[NO_DOCS_REQD]

We have the documents in decreasing order of score. Also these documents match a good number of terms from the query vector. Now we can display these documents to the user.

Results

Basic Vector Space Model

1. **Query: “list of mathematicians“**

   Time Taken: 0.5750460624694824 seconds
   Precision : No of relevant documents retrieved / no of documents retrieved
   : 10 / 10 = 1
   Recall: : No of relevant documents retrieved / no of relevant documents
   : 10 / 26 = 0.38

Top 10 Documents [Titles]	Score	Is the Document Relevant to the Query?
List of mathematicians (Z)	0.6858636628380806	Yes
List of mathematicians (Y)	0.6858636628380806	Yes
List of mathematicians (X)	0.6858636628380806	Yes
List of mathematicians (W)	0.6858636628380806	Yes
List of mathematicians (V)	0.6858636628380806	Yes
List of mathematicians (U)	0.6858636628380806	Yes
List of mathematicians (T)	0.6858636628380806	Yes
List of mathematicians (S)	0.6858636628380806	Yes
List of mathematicians (R)	0.6858636628380806	Yes
List of mathematicians (Q)	0.6858636628380806	Yes

2. **Query: 'Members of the United States House of Representatives'**

   Time Taken: 1.1786561012268066 seconds
   Precision : No of relevant documents retrieved / no of documents retrieved
   : 8 / 10 = 0.8
   Recall: : No of relevant documents retrieved / no of relevant documents
   : 8 / 14 = 0.5714

Top 10 Documents [Titles]	Score	Is the Document Relevant to the Query?
William Holmes Brown	0.35371704367563134	Yes
List of Members of the United States House of Representatives in the 32nd Congress by seniority	0.3413117148771327	Yes
List of Members of the United States House of Representatives in the 33rd Congress by seniority	0.3413117148771327	Yes
List of Members of the United States House of Representatives in the 34th Congress by seniority	0.3413117148771327	Yes
List of Members of the United States House of Representatives in the 35th Congress by seniority	0.3413117148771327	Yes
List of Members of the United States House of Representatives in the 36th Congress by seniority	0.3413117148771327	Yes
List of Members of the United States House of Representatives in the 37th Congress by seniority	0.3413117148771327	Yes
List of Members of the Canadian House of Commons with military service (N)	0.26796810723035686	No

John Carroll (Hawaii politician)	0.24604832883459082	Yes
Northview	0.22220864486466896	No

3. **Query: "Black River"**
   Time Taken: 1.0761568546295166 seconds
   Precision : No of relevant documents retrieved / no of documents retrieved
   : 1 / 10 = 0.1
   Recall: : No of relevant documents retrieved / no of relevant documents
   : 1 / 1 = 1

Top 10 Documents [Titles]	Score	Is the Document Relevant to the Query?
Alfred Black	0.36056835506973617	No
List of Black political parties	0.2376154296026761	No
Kadifekale Samsun	0.2293779159089281	No
Fresh River (Massachusetts)	0.22771051611523077	No
Black River (1993 film)	0.22384502278686993	Yes
Uoro River	0.2607763493938971	No
List of settlements on the River Tees	0.19017444002443118	No
Alfred Black (cricketer)	0.18194221055418616	No
Terengganu River	0.17576361911738983	No
Liuyang River	0.17366253750219635	No

4. **Query: 'Jazz festival'**
   Time Taken:1.09295654296875 seconds
   Precision : No of relevant documents retrieved / no of documents retrieved
   : 3 / 10 = 0.3
   Recall: : No of relevant documents retrieved / no of relevant documents
   : 3 / 12 = 0.25

Top 10 Documents [Titles]	Score	Is the Document Relevant to the Query?
Trondheim Jazz Festival	0.22734334393643957	Yes
Polarjazz	0.21413875302733187	Yes
North Coast Brewing Company	0.20159040647606294	No
Newport Rebels	0.18361271393688064	No
Mister Peabody Goes to Baltimore	0.16761778761594892	No
ReMembrance (Joe McPhee album)	0.16340280884898492	No
In Black and White (Trio X al bum)	0.14414300778747235	No
Mildura Country Music Festival	0.12052659640987519	Yes
Woman Talk	0.1176193500663347	No

Pstereo Festival	0.11507851081838368	Yes

5. **Query: 'dimension 5'**
   Time Taken: 1.0692546367645264 seconds
   Precision : No of relevant documents retrieved / no of documents retrieved
   : 1 / 10 = 0.1
   Recall: : No of relevant documents retrieved / no of relevant documents
   : 1 / 2 = 0.5

Top 10 Documents [Titles]	Score	Is the Document Relevant to the Query?
Vladimir Vlassov	0.09229067248738389	No
Harald Duschek	0.08902509976531355	No
Peter Leitner	0.08902509976531355	No
Dimension 5 (film)	0.0748402577251654	Yes
Đorđe Vukobrat	0.07354577434988445	No
Poblacion V, Calamba	0.07139197682833476	No

Ivar Jakobsen	0.0688116968908772	No
Investment (macroeconomics)	0.06680305102923353	No
Paul Standfield	0.06612038259810422	No
Joe Lukeman	0.06573418933147177	No

6. **Query: 'election in north wales'**
   Time Taken: 0.6148538589477539 seconds
   Precision : No of relevant documents retrieved / no of documents retrieved
   : 0 / 10 = 0
   Recall: : No of relevant documents retrieved / no of relevant documents
   : 0 / 6 = 0

Top 10 Documents [Titles]	Score	Is the Document Relevant to the Query?
1951 in Wales	0.36560743003953766	No
Mid and West Wales (National Assembly for Wales electoral region)	0.30314566650181457	No
North Wales (National Assembly for Wales electoral region)	0.2593207763766138	No

New South Wales Derby	0.22870030307062209	No
Wrexham (Assembly constituency)	0.2198824385749321	No
Clwyd South (Assembly constituency)	0.20707069549697707	No
Camponotus pallidiceps	0.1466832905104807	No
Tendring District Council election, 2015	0.2040409296651519	No
South Wales West (National Assembly for Wales electoral region)	0.2040409296651519	No
South Wales Central (National Assembly for Wales electoral region)	0.19942359049828315	No

7. **Query: "Basketball Player"**
   Time Taken: 0.7999465465545654 seconds
   Precision : No of relevant documents retrieved / no of documents retrieved
   : 7 / 10 = 0.7
   Recall: : No of relevant documents retrieved / no of relevant documents
   : 7 / 16 = 0.4375

Top 10 Documents [Titles]	Score	Is the Document Relevant to the Query?
1994 in basketball	0.46208922687089243	No
Maurice McHartley	0.3293589744768359	Yes

Petar Despotović	0.29142904535982506	Yes
2015–16 Turkish Women's Basketball League	0.2886648515544993	No
Lassi Tuovi	0.26315741066135884	Yes
Tarek Ammoury	0.23527383192935383	Yes
Logan Thunder (QBL)	0.22080410892808 21	No
Gary Plummer (basketball)	0.21450086880570562	Yes
Donna Burns	0.21179924135813044	Yes
George de Paula	0.21103970331462704	Yes

8. **Query: “Art Collector”**
   Time Taken: 0.8207855224609375 seconds
   Precision : No of relevant documents retrieved / no of documents retrieved
   : 3 / 10 = 0.3
   Recall: : No of relevant documents retrieved / no of relevant documents
   : 3 / 9 = 0.33

Top 10 Documents [Titles]	Score	Is the Document Relevant to the Query?
Peninsula Museum of Art	0.18310225179237863	No
Jacques Colombier (art director)	0.17944176202841094	No

Art dealer	0.17196533036454312	No
Harriet Wadeson	0.1515918163804187	No
Medrar for contemporary art	0.14652082094660585	No
Patrick Meagher (artist)	0.14459190555623494	Yes
Priveekollektie Contemporary Art / Design	0.13925634207801843	No
József Ács (painter)	0.1264171389944757	Yes
Walter P. Chrysler Jr.	0.11715637579092662	Yes
Children's Art Museum of Nepal	0.11654006902589462	No

9. **Query: “British Businessman”**
   Time Taken: 1.3531482219696045 seconds
   Precision : No of relevant documents retrieved / no of documents retrieved
   : 2 / 10 = 0.2
   Recall: : No of relevant documents retrieved / no of relevant documents
   : 2 / 8 = 0.25

Top 10 Documents [Titles]	Score	Is the Document Relevant to the Query?
British School of Bucharest	0.18147603332233264	No
Simon Arora	0.1716154483507595	Yes

British National Bibliography	0.15254167442540453	No
British International School of Zagreb	0.15254167442540453	No
Tony Langley	0.1489144711179368	Yes
George Rodgers (British politician)	0.133448539088067	No
Michael Ryan (actor)	0.133448539088067	No
Peter Jones (British Army officer)	0.12921099236594827	No
All-British League	0.12583513286116377	No
Ewen Broadbent	0.12535307518016045	No

10. Query: 'incidents which occurred in 2015'
Time Taken: 1.3883562088012695 seconds
Precision : No of relevant documents retrieved / no of documents retrieved
: 6 / 10 = 0.6
Recall: : No of relevant documents retrieved / no of relevant documents
: 6 / 12 = 0.5

Top 10 Documents [Titles]	Score	Is the Document Relevant to the Query?
2015 Bahia landslide	0.20463046823666356	Yes
2015 Turkey blackout	0.1869117481295692	Yes

List of Cuba–United States aircraft hijackings	0.16406403452557822	No
2015 in Hungary	0.1248491172033054	Yes
2015 in Serbia	0.12452777704771376	Yes
2015 Butler Bulldogs softball team	0.12411215765976594	No
2015–16 Brighton & Hove Albion F.C. season	0.11770272709796543	Yes
Karma (2015 TV series)	0.11705657633496394	No
2015 in Swedish television	0.11505882297045582	No
2015 ARFU Women's Sevens Championships	0.11178011875452921	Yes

Improved-1 Vector Space Model Improvements

1. **Query: "Black River"**
   Time Taken: 0.0013298988342285156 seconds
   Precision : No of relevant documents retrieved / no of documents retrieved
   : 1 / 10 = 0.1
   Recall: : No of relevant documents retrieved / no of relevant documents
   : 1 / 1 = 1
   (For reference of corresponding results from basic vector space model results refer table no. 3)

Top 10 Documents [Titles]	Score	Is the Document Relevant to the Query?
Liuyang River	0.48497738695653125	No
Alfred Black	0.4674063687025042	No
Laodao River	0.39994692165071294	No
Fresh River (Massachusetts)	0.38462551029210523	No
Uoro River	0.3663863164396	No
List of Black political parties	0.35400207406527634	No
Black River (1993 film)	0.3499840041890272	Yes
Alfred Black (cricketer)	0.32537039879415236	No
Liverpool River	0.28686704806226226	No
Terengganu River	0.28674003143081406	No

2. **Query: 'Jazz festival'**
   Time Taken: 0.0011153221130371094 seconds
   Precision : No of relevant documents retrieved / no of documents retrieved
   : 7 / 10 = 0.7
   Recall: : No of relevant documents retrieved / no of relevant documents
   : 7 / 12 = 0.583
   (For reference of corresponding results from basic vector space model results refer table no. 4)

Top 10 Documents [Titles]	Score	Is the Document Relevant to the Query?
Trondheim Jazz Festival	0.4046730219390639	Yes
Polarjazz	0.2812544686021669	Yes
Arvika Festival	0.2357940723840135	Yes
Pstereo Festival	0.20778057911582867	Yes
AV Festival	0.190671314691344	Yes
Festival Western de Saint-Tite	0.1900116729133078	No
Mildura Country Music Festival	0.18295372393402262	Yes
Richard Pite	0.18000309113527635	No
Kazakhstan in the ABU TV Song Festival	0.17624093829230664	Yes
Ondrej Krajnak	0.1717588752858209	No

3. **Query: 'dimension 5'**
   Time Taken: 0.0006539821624755859 seconds
   Precision : No of relevant documents retrieved / no of documents retrieved
   : 1 / 10 = 0.1
   Recall: : No of relevant documents retrieved / no of relevant documents
   : 1 / 2 = 0.5
   (For reference of corresponding results from basic vector space model results refer table no. 5)

Top 10 Documents [Titles]	Score	Is the Document Relevant to the

		Query?
Five Sixteenth-Century Poems	0.14145858216972804	No
Witmer Lake	0.07694384569133565	No
Dimension 5 (film)	0.04887154791354635	Yes
Robert Herring House	0.0473725200469583	No
The Mystery of Holly Lane	0.045832639852634534	No
Fakahoko	0.04546341593145644	No
Once Again (John Legend album)	0.03737645960579285	No
Niuean mythology	0.033536471474251664	No
De Bruijn torus	0.03293453311553621	No
Fao (god)	0.03232156740884942	No

CORNER CASE FOR IMPROVEMENT 1

Query: 'election in north wales'
Time Taken: 0.0016129016876220703 seconds
Precision : No of relevant documents retrieved / no of documents retrieved
: 2 / 10 = 0.2
Recall: : No of relevant documents retrieved / no of relevant documents
: 2 / 6 = 0.33
(For reference of corresponding results from basic vector space model results refer table no. 6)

Top 10 Documents [Titles]	Score	Is the Document Relevant to the Query?
1951 in Wales	0.36560743003953766	No
New South Wales Derby	0.30314566650181457	No
Brooklyn by-election, 1951	0.26241923766569886	Yes
Connecticut Senate election, 2006	0.2593207763766138	No
1946 New South Wales Grand Prix	0.22870030307062209	No
Fatima Island (New South Wales)	0.2198824385749321	No
North Wales (National Assembly for Wales electoral region)	0.20707069549697707	Yes
Tendring District Council election, 2015	0.2047006646871565	No
South Wales West (National Assembly for Wales electoral region)	0.2040409296651519	No
South Wales Central (National Assembly for Wales electoral region)	0.19942359049828315	No

Improved-2 Vector Space Model Improvements

1. **Query: "Basketball Player"**
   Time Taken: 0.0035500526428222656 seconds
   Precision : No of relevant documents retrieved / no of documents retrieved
   : 9 / 10 = 0.9
   Recall: : No of relevant documents retrieved / no of relevant documents
   : 9 / 16 = 0.5625

(For reference of corresponding results from basic vector space model results refer table no. 7)

Top 10 Documents [Titles]	Score	Is the Document Relevant to the Query?
Maurice McHartley	0.3293589744768359	Yes
Petar Despotović	0.29142904535982506	Yes
Tarek Ammoury	0.23527383192935383	Yes
Gary Plummer (basketball)	0.21450086880570562	Yes
Donna Burns	0.21179924135813044	Yes
George de Paula	0.21103970331462704	Yes
Mo Ke	0.20480367493040977	Yes
2014–15 Ivy League men's basketball season	0.19848549592288403	No
Bader Makki	0.19838905366067103	Yes
Brian Davis (basketball)	0.1967385400169552	Yes

2. **Query: "Art Collector"**
   Time Taken: 0.00162363052368164 seconds
   Precision : No of relevant documents retrieved / no of documents retrieved
   : 5 / 10 = 0.5
   Recall: : No of relevant documents retrieved / no of relevant documents

: 5 / 9 = 0.56

(For reference of corresponding results from basic vector space model results refer table no. 8)

Top 10 Documents [Titles]	Score	Is the Document Relevant to the Query?
Peninsula Museum of Art	0.18310225179237863	No
Walter P. Chrysler Jr.	0.11715637579092662	Yes
Donald Saff	0.10784539530894063	Yes
Gagea olgae	0.08665996513636545	No
Art McNally Award	0.0847467418327695	No
List of Irish botanical illustrators	0.08366515294530369	No
Yegizaw Michael	0.07664336244053255	Yes
Charles Benenson	0.07632931532133133	Yes
The Art of Ecstasy	0.07366719799646725	No
Merlin Little Thunder	0.06979396911555844	Yes

3. **Query: "British Businessman"**
   Time Taken: 0.0049588680267333984 seconds
   Precision : No of relevant documents retrieved / no of documents retrieved
   : 6 / 10 = 0.6

Recall: : No of relevant documents retrieved / no of relevant documents
: 6 / 8 = 0.75
(For reference of corresponding results from basic vector space model results refer table no. 9)

Top 10 Documents [Titles]	Score	Is the Document Relevant to the Query?
Simon Arora	0.1716154483507595	Yes
Tony Langley	0.1489144711179368	Yes
Tony Murray (businessman)	0.1109627165771897	Yes
Jezdimir Vasiljević	0.10019654050496528	No
Milan Platovsky	0.09755932668767284	No
Fred Done	0.09298611850648031	Yes
Peter Done	0.08918097030765365	Yes
John Kidston Swire	0.08530932078712197	Yes
Norman Myer	0.08017920231870682	No
John S. MacDonald	0.07232200550350551	No

CORNER CASE FOR IMPROVEMENT 2

Query: ‘incidents which occurred in 2015’
Time Taken: 0.0016129016876220703 seconds
Precision : No of relevant documents retrieved / no of documents retrieved
: 1 / 10 = 0.1
Recall: : No of relevant documents retrieved / no of relevant documents
: 1 / 12 = 0.0833
(For reference of corresponding results from basic vector space model results refer table no. 10)

Top 10 Documents [Titles]	Score	Is the Document Relevant to the Query?
2015 Turkey blackout	0.05852570051876621	Yes
Ernst Lörtscher	0.05766708624621551	No
BaFa' BaFa'	0.0415151821834754	No
Connecticut Senate election, 2006	0.02654779687220703	No
Brian Davis (basketball)	0.025023782549643252	No
Helix HeadRoom	0.0195520642555501	No
INSAT-4C	0.018958334005099997	No
2016 Indianapolis 500	0.01891151383427319	No
Connington South rail crash	0.018584270412150557	No
Slovenian Navy	0.018553894391926586	No

Discussion and Conclusions

Issues with the basic vector space model in part 1:

- Title of the documents is not considered while scoring and ranking the documents. The titles which have more common words with the query are ought to be more relevant to the search requirement.
- In case of a very large corpus, queries can be slow because of iterating through all the

documents in the posting list.

Improvement 1 proposed via vector space model in part 2:

- Implementation of weighted term frequency to take into account the query terms appearing in the title as well.
- Implementation of **champion lists** by cutting down the size of posting lists to top 100 or lesser documents for each term **along with use of pre-built normalisation index** can help improve the efficiency of the system.

Examples of queries with Improved Results: [Details in Results section]

- Black River
- Jazz Festival
- Dimension 5

Corner cases when the improvements might not work:

- Inappropriate or irrelevant titles.
- Cases where important documents are not present in the champion lists.

Issues with Vector Space Models in parts 1 and improvement 1 in part 2:

- Do not directly take into account the number of words in the document matching with the query.
- Do not take into account the df saturation i.e. words which occur very frequently in the corpus therefore they cannot be used to rank the documents properly.

Improvement 2 proposed via model in part 2:

- This model removes the words which cannot be used to uniquely identify and score documents therefore creating a better scoring system.
- This model also overcomes shortcomings of the model in part-1 and improvement 1 in part 2 with added support for taking into account **number of matching query words and also uses pre-built normalisation index,** thus improving both relevance and efficiency.

Examples of queries with Improved Results: [Details in Results Section]

- Basketball player
- Art Collector
- British Businessman

Corner cases when the improvement 2 might not work:

- When terms with high document frequency are important to the meaning of the query and removing them may cause loss in that meaning..eg. **Query: 'incidents which occurred in 2015'** in this query 'in' is a connecting term which tells us that we want incidents only in year 2015.Removal of in leads to loss in that meaning

Future Scope of Project:

Including Machine learning models, Natural Language Processing Techniques, skip pointers and positional indexes could help create a scalable system. Preprocessing techniques such as stemming, lemmatization could also help improve the accuracy and recall of the retrieval system.

References

1. C. D. Manning, P. Raghavan and H. Schutze. Introduction to Information Retrieval, Cambridge University Press, 2008.
2. Information Retrieval - https://en.wikipedia.org/wiki/Information_retrieval
3. Python - https://en.wikipedia.org/wiki/Python_(programming_language)
4. Beautiful Soup Documentation - https://www.crummy.com/software/BeautifulSoup/bs4/doc/