A VECTOR SPACE MODEL FOR INFORMATION RETRIEVAL: A MATLAB APPROACH

A. B. Manwar

Department of Computer Science,
S.G.B. Amravati University, Amravati MS, India
avinash.manwar@gmail.com

Hemant S. Mahalle

Department of Computer Science,
P. N. College, Pusad, Dist. Yavatmal MS, India
mahalle_hemant@yahoo.co.in

K. D. Chinchkhede

Department of Applied Electronics,
S.G.B. Amravati University, Amravati MS, India
krish.chinchkhede@gmail.com

Dr. Vinay Chavan

Department of Computer Science,
S. K. Porwal College, Kamptee, Nagpur MS, India
drvinaychavan@yahoo.co.in

Abstract

By and large, three classic framework models have been used in the process of retrieving information: Boolean, Vector Space and Probabilistic. Boolean model is a light weight model which matches the query with precise semantics. Because of its boolean nature, results may be tides, missing partial matching, while on the contrary, vector space model, considering term-frequency, inverse document frequency measures, achieves utmost relevancy in retrieving documents in information retrieval. This paper implements and discusses the issues of information retrieval system with vector space model using MATLAB on Cranfield data collection of aerodynamics domain.

***Keywords*:** tf; idf; vector space model; cosine similarities; term-document; term-query matrices; dot products.

1. Introduction

Enormous amount of text material is increasing at exponential rate, especially with the increasing use and applications of Internet. Day by day it is becoming very difficult to retrieve the relevant information. Various approaches have been used by the researchers to get over the relevancy factor in information retrieval.

An information retrieval model is a quadruple consisting of document collection, set of queries, framework model and a ranking function associated with query-document. A framework model may be boolean, vector space or probabilistic. Boolean model matches query with precise semantics in the document collection by boolean operations with operators AND, OR, NOT. It predicts either relevancy or non-relevancy of each document, leading to the disadvantage of retrieving very few or very large documents. The boolean model is the lightest model having inability of partial matching which leads to poor performance in retrieval of information. Vector space model is introduced by G. Salton in late 1960s in which partial matching is possible. Non-binary weights are used to weight the index terms in queries and in documents. These words are used for calculating degree of similarity between each document and the query. The ranked document set in the decreasing order of degree of similarity thus obtained is precise than the result of boolean model. Index term weights can be calculated in many different ways. The work by Salton and McGill [1] reviews various term-weighting techniques. Although there is contention as to whether probabilistic model outperforms the vector model, Salton

and Buckley [2] showed that the vector model is expected to outperform the probabilistic model with general collections [3].
This paper implements and discusses the issues of information retrieval system with vector space model using MATLAB on Cranfield data collection of aerodynamics domain.
The next section deals with brief review of related work of vector space model in information retrieval.

2. Related work

Maron and Kuhns [4] in early 1960, described probabilistic indexing technique in a mechanized library system yielding probable relevance. Afterword in 1983, Salton and McGill wrote a book [1] which discusses thoroughly the three classic models in information retrieval namely, the boolean, the vector, and the probabilistic models. The book by van Rijsbergen [5] covers the discussion on three classic models and majority of the associated technology of retrieval system. Frakes and Baeza-Yates [6] edited the book on information retrieval which mainly deals with the data structures used in general information retrieval systems. Also, it includes the issue of relevance feedback as well as some query modification techniques [7] and Boolean operations and their implementations [8]. Verhoeff, Goffman, and Belzer [9] described the shortfall of boolean queries for information retrieval. The concept of using boolean formalism in other frameworks had been the great interest area of the researchers. Lee et al proposed a thesaurus-based boolean retrieval system for ranking [10].
Vector space model has been the most popular model in information retrieval among the research vicinity because of the research outcome in indexing, term value specification in automatic indexing carried out by Salton and his associates [11, 12]. Most of this research deals with experiments in automatic document processing and different term weighting approaches for automatic retrieval [2, 13]. In 1972, Karen Sparck Jones introduced the concept of inverse document frequency, a measure of specificity [14, 15] and Salton and Yang uses it for automatic indexing to improve retrieval [12]. Raghavan and Wong [16] analyses vector space model critically with the conclusion that the vector space model is useful and which provides a formal framework for the information retrieval systems.
The next section gives a description of the most influential vector space model in modern information retrieval research.

3. Vector Space Model

The drawback of binary weight assignments in boolean model is remediated in the vector space model which projects a framework in which partial matching is possible [11, 13]. Non-binary weights for index terms in queries and documents are used in the calculation of degree of similarity. Decreasing order of this degree of similarity for the retrieved documents gives the ranked documents with partial match.
For the vector model, the weight $w_{i,j}$ associated with a pair (k_i, d_j) is positive and non-binary. Further, the index terms in the query are also weighted. Let $w_{i,q}$ be the weight associated with the pair$[k_i, q]$, where $w_{i,q} \geq 0$. Then, the query vector $\vec{q}$ is defined as $\vec{q} = (w_{1,q}, w_{2,q}, \dots, w_{t,q})$ where *t* is the total number of index terms in the system. The vector for a document d_j is represented by $\overrightarrow{d_j} = (w_{1,j}, w_{2,j}, \dots, w_{t,j})$.

The vector model proposes to evaluate the degree of similarity of the document d_j with regard to the query q as the correlation between the vectors $\overrightarrow{d_j}$ and $\vec{q}$. This correlation can be measured by the cosine of the angle between these two vectors as,

$$sim(d_j, q) = \frac{\overrightarrow{d_j} \cdot \vec{q}}{|\overrightarrow{d_j}| \times |\vec{q}|}$$

$$= \frac{\sum_{i=1}^{t} w_{i,j} \times w_{i,q}}{\sqrt{\sum_{i=1}^{t} w_{i,j}^2} \times \sqrt{\sum_{j=1}^{t} w_{i,q}^2}}$$

where $|\overrightarrow{d_j}|$ and $|\vec{q}|$ are the norms of the document and query vectors. The factor $|\vec{q}|$ does not affect the ranking (i.e., the ordering of the documents) because it is the same for all documents. The factor $|\overrightarrow{d_j}|$ provides normalization in the space of the documents.
Since $w_{i,j} \geq 0$ and $w_{i,q} \geq 0$, $sim(q, d_j)$ varies from 0 to +1, the vector model ranks the documents according to their *degree of similarity* to the query. A document might be retrieved even if it matches the query only *partially.* [3, page 27-28].

In the vector space model, the frequency of a term k_i inside a document d_j referred to *as* the *tf* factor and provides one measure of how well that term describes the document contents. Furthermore, the inverse of the frequency of a term k_i among the documents in the collection referred to as the *inverse document frequency* or the *idf* factor.

The normalized frequency $f_{i,j}$ of term k_i in document d_j is given by

$$f_{i,j} = \frac{freq_{i,j}}{max_l freq_{l,j}} \quad (1)$$

where the maximum is computed over all terms which are mentioned in the text of the document d_j. The idf_i, inverse document frequency for k_i, be given by

$$idf_i = log \frac{N}{n_i} \quad (2)$$

The best known term-weighting schemes use weights which are given by

$$w_{i,j} = f_{i,j} \times log \frac{N}{n_i} \quad (3)$$

Such term-weighting strategies are called *tf-idf* schemes [3, page 29-30].

The success of vector space model lies in its term-weighting scheme, its partial matching strategy and similarity measure. Mutual independence of index terms has said to be disadvantage of vector space model but practically, consideration of term dependencies is not fruitful. From the research consequence in the field, it seems that the vector model is either superior or almost as good as the known alternatives.

4. Experimental Evaluations

4.1. *Dataset for information retrieval system*

We used a Cranfield collection having 1398 abstracts related to aerodynamics domain which is obtained from [17]. The collection contains a compressed version of document text, relation giving relevance judgements, text of 225 queries, indexed documents and indexed queries. Also, we used stop-list obtained from the collection source.

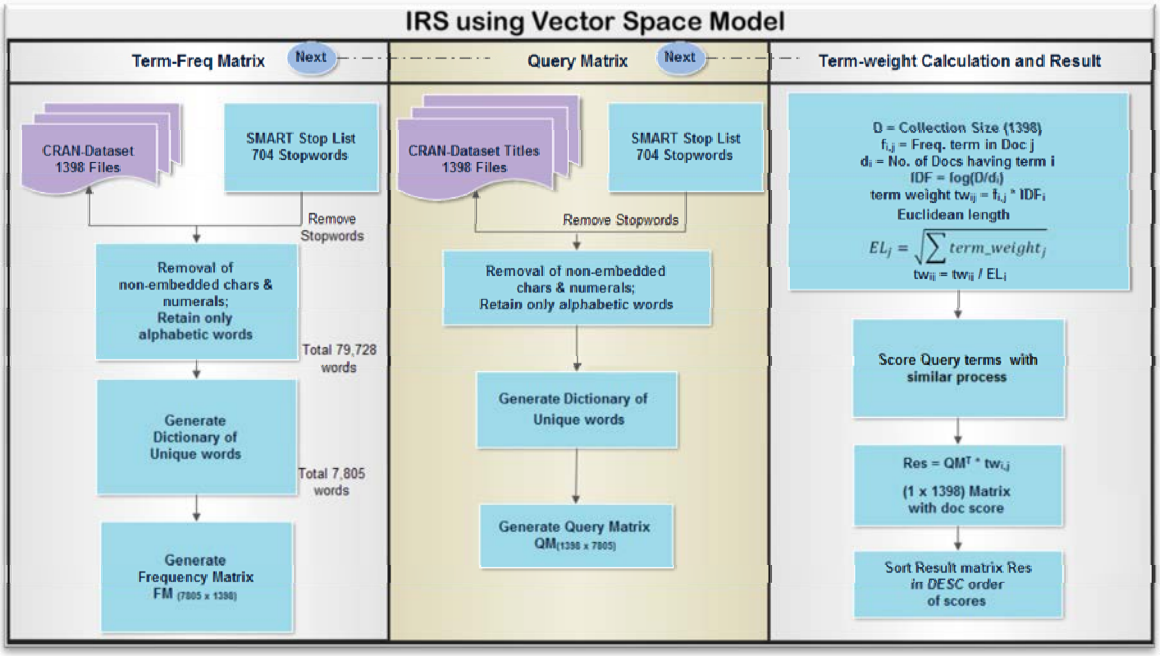
4.2. *Preprocessing*

The compressed version of document text has been preprocessed to obtain a set of 1398 individual abstract files. A PHP script has been written for this purpose. Stemming has not applied for the reasons of i) losing context of search, ii) may reduces precision and iii) can not be applied to proper nouns. To have a clear view of relevancy, instead of using given set of query and relevance judgements, we have created our set of queries from 'titles' of 1398 documents, forming a query set of 1398 entities.

4.3. *Implementation*

A MATLAB is used to implement a vector space model for information retrieval; the complete process is depicted in fig. 1.

Fig. 1. Implementation of vector space model for information retrieval.



As shown in block diagram it consists of three stages:

1. Generation of Term-Frequency matrix

   It is term-frequency matrix of all unique terms in document d_j with $j = 1,2,\dots,N$. The term document matrix (FM) is $M \times N$ matrix with t_i unique terms in dictionary $(i = 1,2,\dots,M)$ and N documents. The elements of FM are represented as $A_{i,j}$ in which each element indicates the frequency of i^{th} term in j^{th} document.

   The Cranfield data collectiont is preprocessed to convert into individual 1398 text files. A SMART stop word text file which is available with the dataset is used for the removal of stop words from the data collection of 1398 files. Also, non-embedding special characters and numerals have been removed from these files. 79,728 words have been collected which are then processed to find the frequency of unique words in each documents. The dictionary of unique words is of 7805 words. Thus the term-frequency matrix is of size 7805x1398.

2. Generation of Query matrix

   The title of each abstract, after removing stop-list words and non-embedding special characters is used as query, which contributes to the set of 1398 unique queries represented as q_j. Here, we have taken queries as titles of the document instead of the dataset queries so as to judge the relevancy more profoundly. The generated matrix for 1398 queries is $QM_{1398\times7805}$.

3. Term-weight calculations and result

   A term-frequency matrix is processed to get the term weights considering *tf-idf* scheme. These term weights are calculated by the formula $tw = (tf \times idf)/EL$, where

   $tw_{i,j}$ is the term-weight i.e. weight of term i in document j,

   $tf_{i,j}$ is the frequency of term i in document j,

   idf is the inverse document frequency representing the terms appearing in many documents and is calculated by the formula $\log(\frac{D}{d_j})$, where D is the total number of documents and d_j is the number of documents containing term j,

   EL is the Euclidean length obtaining by taking square root of sum of squares of individual terms per document.

Query matrix of size $\mathbf{1398 \times 7805}$ is also divided by their corresponding *Euclidean* lengths to obtain the normalized weights. For query $\boldsymbol{q}$, the transpose of query matrix is multiplied by the term-weight matrix. The final result is obtained by ordering the weights in a result matrix in decreasing manner of their weights.

4.4. *Testing phase*

The vector space model (VSM) implemented above is tested thoroughly in different ways, the details of experimentation is explained below:
Fig. 2 shows the distribution of index terms in dictionary for individual documents. The dictionary consists of 7805 unique terms.

Fig. 2. Index terms in a dictionary

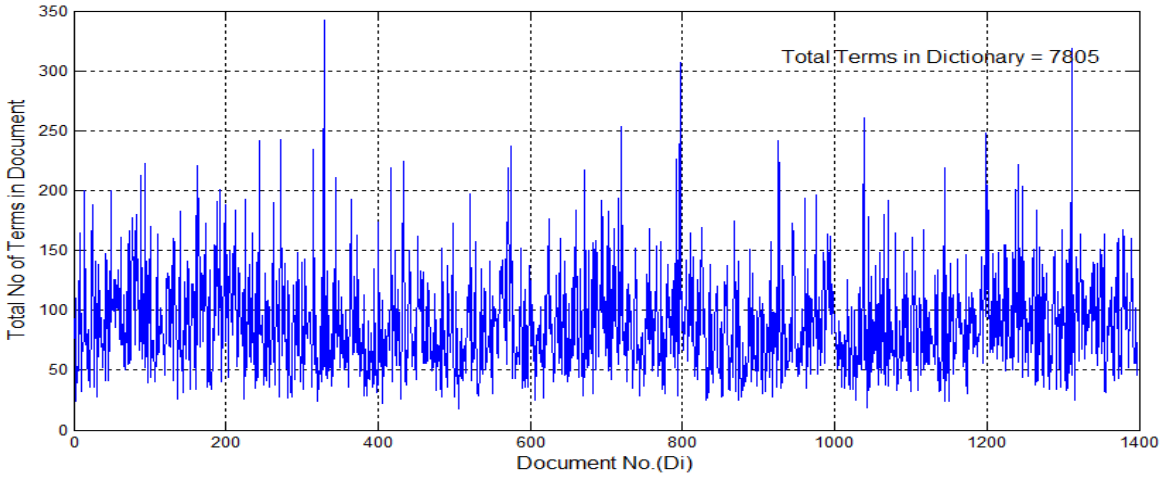
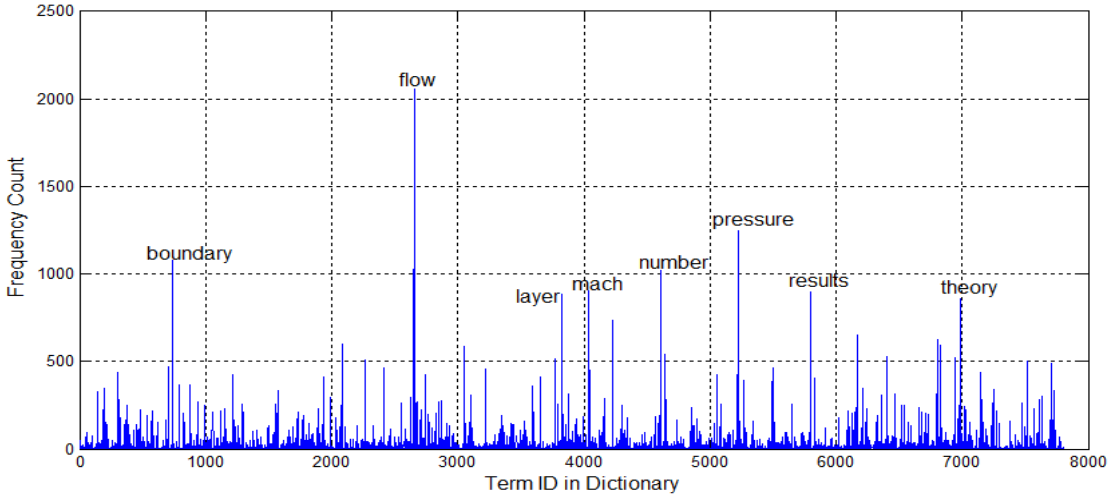


Fig. 3 shows frequency count of each unique term in dictionary distributed in complete dataset. Some of the unique terms such as ('flow', 2059), ('pressure', 1245), ('boundary', 1076), ('results', 897), with high frequency in entire documents is shown.

Fig. 3. Frequency count of each unique term among data collection



Further looking into the dataset, as shown in fig. 4, shows subset of 380 documents out of 1398 abstracts. In document 329 the total terms are 342 but the unique terms are only 162.

Fig. 4. Document-wise total and unique terms

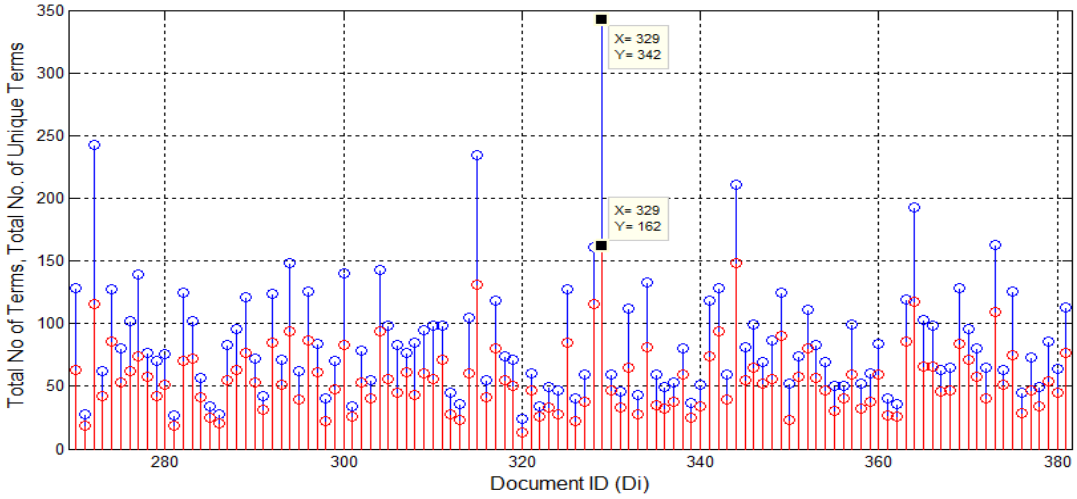


Table I is the subset of 20 queries out of 1398 tested queries against the vector space model for retrieval of relevant documents. In the present study the most relevant document is one whose ID matches with query ID.

Table I is the subset of 20 queries out of 1398 tested queries against the vector space model for retrieval of relevant documents. In the present study the most relevant document is one whose ID matches with query ID.

Table 1. Subset of 20 queries

Q.ID	Details of Queries
1	experimental investigation of the aerodynamics of a wing in a slipstream .
2	simple shear flow past a flat plate in an incompressible fluid of small viscosity .
3	the boundary layer in simple shear flow past a flat plate .
4	approximate solutions of the incompressible laminar boundary layer equations for a plate in shear flow .
5	one-dimensional transient heat conduction into a double-layer slab subjected to a linear heat input for a small time internal .
6	one-dimensional transient heat flow in a multilayer slab .
7	the effect of controlled three-dimensional roughness on boundary layer transition at supersonic speeds .
8	measurements of the effect of two-dimensional and three-dimensional roughness elements on boundary layer transition .
9	transition studies and skin friction measurements on an insulated flat plate at a mach number of 5.8 .
10	the theory of the impact tube at low pressure .
11	similar solutions in compressible laminar free mixing problems .
12	some structural and aerelastic considerations of high speed flight .
13	similarity laws for stressing heated wings .
14	piston theory - a new aerodynamic tool for the aeroelastician .
15	on two-dimensional panel flutter .
16	transformation of the compressible turbulent boundary layer .
17	remarks on the eddy viscosity in compressible mixing flows .
18	the flow field in the diffuser of a radial compressor .
19	an investigation of the pressure distribution on conical bodies in hypersonic flows .
20	generalised-newtonian theory .

The result of experimentation is tabulated in Table 2 shown below:

Table 2. Tabulation of results of first ten queries

	Query-1		Query-2		Query-3		Query-4		Query-5		Query-6		Query-7		Query-8		Query-9		Query-10	
Rank	Doc ID	Wt.	Doc ID	Wt.	Doc ID	Wt.	Doc ID	Wt.	Doc ID	Wt.	Doc ID	Wt.	Doc ID	Wt.	Doc ID	Wt.	Doc ID	Wt.	Doc ID	Wt.
1	1	0.503	3	0.693	3	0.856	4	0.798	5	0.860	6	0.562	80	0.516	8	0.640	9	0.522	10	0.677
2	453	0.332	389	0.670	4	0.539	3	0.520	484	0.384	5	0.468	7	0.502	7	0.474	346	0.452	183	0.399
3	1142	0.240	2	0.590	389	0.506	180	0.443	6	0.369	484	0.336	8	0.439	80	0.440	567	0.434	238	0.295
4	1062	0.233	4	0.424	663	0.471	663	0.389	399	0.210	91	0.261	43	0.389	43	0.389	125	0.352	239	0.293
5	483	0.232	663	0.412	393	0.469	393	0.356	980	0.209	395	0.259	1209	0.369	96	0.369	1353	0.318	1137	0.269
6	695	0.186	393	0.349	2	0.463	569	0.352	91	0.191	398	0.186	182	0.368	1209	0.366	145	0.298	139	0.261
7	1089	0.182	180	0.330	180	0.450	1180	0.337	581	0.186	181	0.176	1210	0.350	182	0.365	260	0.297	1080	0.249
8	1092	0.181	299	0.294	388	0.343	382	0.324	395	0.186	578	0.170	96	0.349	709	0.344	41	0.276	223	0.214
9	372	0.178	375	0.285	308	0.301	307	0.308	706	0.185	159	0.165	709	0.342	1379	0.308	254	0.273	1225	0.195
10	1339	0.177	388	0.280	658	0.291	376	0.307	29	0.174	581	0.164	40	0.340	314	0.300	1379	0.271	1154	0.194
11	793	0.174	309	0.270	1249	0.277	23	0.302	583	0.174	29	0.164	41	0.335	9	0.274	413	0.268	918	0.194
12	1060	0.172	1249	0.266	310	0.273	1074	0.274	845	0.171	90	0.163	314	0.300	1279	0.270	559	0.263	1313	0.192
13	496	0.169	658	0.258	309	0.273	375	0.273	181	0.168	980	0.163	1379	0.296	4	0.266	524	0.256	905	0.181
14	205	0.155	87	0.256	1180	0.269	1300	0.268	1205	0.160	144	0.157	272	0.258	1210	0.264	1262	0.238	232	0.172
15	1237	0.155	662	0.234	306	0.269	474	0.266	508	0.155	868	0.155	1262	0.235	932	0.254	21	0.226	670	0.168
16	919	0.153	308	0.232	192	0.267	628	0.265	168	0.153	303	0.145	337	0.234	1262	0.249	994	0.226	74	0.162
17	688	0.141	23	0.220	662	0.262	610	0.264	586	0.146	1026	0.145	4	0.232	272	0.248	207	0.226	340	0.154
18	969	0.139	73	0.216	628	0.259	255	0.262	541	0.143	871	0.144	932	0.231	40	0.245	1298	0.220	568	0.146

Further query wise details are depicted in subsequent Table 3 to Table 5 respectively.

Table 3. Frequency of terms in retrieved document set for query 1

Query (Q.ID = 1)	**experimental investigation** of the **aerodynamics** of a **wing** in a **slipstream**																	
	Document ID																	
Terms	D-1	D-453	D-1142	D-1062	D-483	D-695	D-1089	D-1092	D-372	D-1339	D-793	D-1060	D-496	D-205	D-1237	D-919	D-688	D-969
'aerodynamics'	2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0
'experimental'	3	1	0	0	3	0	0	0	3	1	0	3	2	0	0	0	2	0
'investigation'	2	0	1	0	0	2	2	3	4	3	0	2	2	1	1	0	2	1
'slipstream'	6	6	9	5	7	0	1	3	0	0	0	0	0	0	0	0	0	0
'wing'	4	4	1	4	0	13	4	5	0	7	9	2	4	9	11	3	0	7
Total Occurrence	**17**	**12**	**11**	**9**	**10**	**15**	**7**	**11**	**7**	**11**	**9**	**7**	**8**	**10**	**12**	**3**	**6**	**8**

Table 4. Frequency of terms in retrieved document set for query 2

Query (Q.ID = 2)	simple shear flow past a **flat plate** in an **incompressible fluid** of **small viscosity**																	
	Document ID																	
Terms	D-3	D-389	D-2	D-4	D-663	D-393	D-180	D-299	D-375	D-388	D-309	D-1249	D-658	D-87	D-662	D-308	D-23	D-73
'flat'	2	3	4	1	3	3	3	0	3	3	3	0	0	2	3	3	2	0
'flow'	3	3	7	4	0	4	3	4	8	5	5	2	3	3	3	4	6	2
'fluid'	0	5	3	1	0	0	0	3	3	0	4	1	0	2	0	0	1	6
'incompressible'	1	1	3	3	1	0	1	1	2	0	1	1	1	1	0	1	3	3
'past'	2	2	5	0	0	0	0	3	2	2	1	3	3	3	1	3	0	0
'plate'	2	3	4	3	3	3	3	4	3	4	4	5	0	3	4	7	2	0
'shear'	2	3	3	3	3	4	3	0	0	6	0	3	6	0	0	0	0	0
'simple'	2	3	3	0	0	0	0	0	1	1	0	0	0	0	1	0	0	0
'small'	0	0	3	0	1	0	0	1	0	0	0	1	0	0	0	0	0	2
'viscosity'	0	1	3	0	1	0	0	0	1	0	0	1	0	0	0	0	0	3
Total Occurrence	**14**	**24**	**38**	**15**	**12**	**14**	**13**	**16**	**23**	**21**	**18**	**17**	**13**	**14**	**12**	**18**	**14**	**16**

Table 5. Frequency of terms in retrieved document set for query 3

Query (Q.ID = 3)	the **boundary layer** in **simple shear flow past a flat plate**																	
	Document ID																	
Terms	D-3	D-4	D-389	D-663	D-393	D-2	D-180	D-388	D-308	D-658	D-1249	D-310	D-309	D-1180	D-306	D-192	D-662	D-628
'boundary'	2	5	0	3	2	2	3	1	3	1	1	2	2	5	3	6	0	5
'flat'	2	1	3	3	3	4	3	3	3	0	0	4	3	1	3	0	3	1
'flow'	3	4	3	0	4	7	3	5	4	3	2	10	5	9	7	3	3	5
'layer'	2	5	0	3	2	2	3	2	3	1	4	4	4	5	3	7	0	3
'past'	2	0	2	0	0	5	0	2	3	3	3	0	1	2	0	3	1	0
'plate'	2	3	3	3	3	4	3	4	7	0	5	3	4	1	3	0	4	2
'shear'	2	3	3	3	4	3	3	6	0	6	3	0	0	0	1	1	0	1
'simple'	2	0	3	0	0	3	0	1	0	0	0	0	0	0	0	0	1	0
Total Occurrence	**17**	**21**	**17**	**15**	**18**	**30**	**18**	**24**	**23**	**14**	**18**	**23**	**19**	**23**	**20**	**20**	**12**	**17**

Table 3 depicts the result obtained for query 1 with query terms and their occurrences in the documents. First 18 documents are listed. Similar results are shown in table 4 and table 5 for queries 2 and 3 respectively.

5. Results and discussion

Even after taking the 'titles' of the abstract documents as a query set, the final result of retrieval is 89.41%. 148 queries have not shown result as first retrieved document; however, within the range of first two retrieved documents, the result obtained is 94.99%. 2.22% of queries have not retrieved the correct result up to the range of first five documents.

Inter-document characterization and document frequency plays vital role in building ranks of the documents in vector space model. As such, term frequency of the documents d_3, d_{389} and d_2 is 24, 42 and 110 respectively; and the frequency of terms *'flow'*, *'plate'* and *'small'* in all documents is 2059, 421 and 306 respectively, which is much more higher than the average frequency- as a result, table IV shows the outcome of query q_2 as document d_3; however, expected relevant document is d_2. Variety of the weight calculation formulas as suggested by Salton and Buckley [2] have been tested on this collection but we found that the standard *tf-idf* weighting scheme gives the best results.

References

[1] G. Salton and M. J. McGill. *Introduction to Modern Information Retrieval.* McGraw-Hill Book Co., New York, 1983.

[2] G. Salton and C. Buckley. *Term-weighting approaches in automatic retrieval.* Information Processing and Management, 24(5):513-523, 1988.

[3] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schutze, *Introduction to Information Retrieval,* Cambridge University Press, New York, USA, 2008.

[4] M. E. Maron and J. L. Kuhns. *On relevance, probabilistic indexing and information retrieval.* Association for Computing Machinery, 7(3):216-244, 1960.

[5] C. J. van Rijsbergen. *Information Retrieval*. Butterworths, London, 1979.

[6] W. B. Frakes and R. Baeza-Yates. *Information Retrieval: Data Structures and Algorithms*. Prentice Hall, Englewood Cliffs, NJ, USA, 1992.

[7] D. Harman. *Relevance feedback and other query modification techniques*. In W. B. Frakes and R. Baeza-Yates, editors, Information Retrieval: Data Structures and Algorithms, pages 241-263. Prentice Hall, Englewood Cliffs, NJ, USA, 1992.

[8] S. Wartick. Boolean operations. In W. B. Frakes and R. Baeza-Yates, editors, *Information Retrieval: Data Structures and Algorithms*, pages 264-292. Prentice Hall, Englewood Cliffs, NJ, USA, 1992.

[9] J. Verhoeff, W. Goffmann, and Jack Belzer. *Inefficiency of the use of Boolean functions for information retrieval systems*. Communications of the ACM, 4(12):557-558, 594, December 1961.

[10] J. H. Lee, W. Y. Kim, and Y. H. Lee. *Ranking documents in thesaurus-based Boolean retrieval systems.* Information Processing and Management, 30(1):79-91, 1993.

[11] G. Salton and M. E. Lesk. *Computer evaluation of indexing and text processing.* Journal of the ACM, 15(1):8-36, January 1968.

[12] Gerad Salton and C. S. Yang. *On the specification of term values in automatic indexing.* Journal of Documentation, 29:351-372, 1973.

[13] G. Salton. The *SMART Retrieval System – Experiments in Automatic Document Processing*. Prentice Hall Inc., Englewood Cliffs, NJ, 1971.

[14] K. Sparck Jones. *A statistical interpretation of term specificity and its application to retrieval*. Journal of Documentation, 28(1):11-20, 1972.

[15] K. Sparck Jones. *A statistical interpretation of term specificity and its application to retrieval*. Information Storage and Retrieval, 9(11):619-633, 1973.

[16] V. V. Raghavan and S. K. M. Wong. A *critical analysis of vector space model for information retrieval*. Journal of the American Society for Information Sciences, 37(5):279-287, 1986.

[17] ftp server of Cornell University ftp://ftp.cs.cornell.edu/pub/smart/cran/ for Cranfield collection.