Analysis of Vector Space Model in Information Retrieval

Jitendra Nath Singh
Dept of Computer Science
BBA University, Lucknow
singhjn2000@gmail.com

Sanjay Kumar Dwivedi
Dept of Computer Science
BBA University, Lucknow
skd200@yahoo.com

ABSTRACT
Information retrieval is great technology behind web search services. In information retrieval, it is common to model index terms and documents as vectors in a suitably defined vector space. The vector space model is one of the classical and widely applied retrieval models to evaluate relevance of web page. The retrieval operation consists of computing the cosine similarity function between a given query vector and the set of documents vector and then ranking documents accordingly. In this paper, we present different approaches of vector space model to compute similarity score of hits from search engine and more importantly, it is felt that this investigation will lead to a clearer understanding of the issues and problems in using the vector space model in information retrieval and our work intends to discuss the main aspects of Vector space models and provide a comprehensive comparison for Term- Count model, Tf-Idf model and Vector space model based on normalization.

Keywords
Vector space model, Information Retrieval, Tf-Idf, Term Frequency, Cosine Similarity.

1. INTRODUCTION
Information retrieval systems are designed to help users to quickly find useful information on the web. The field of information retrieval attained peak popularity during last forty years, number of researchers contributed through their efforts and achieved several remarkable milestones in order to facilitate the internet users with easiest searching in very small slots of time. In the recent years, number of doubts emerged that demand for adequate solutions to satisfy searchers. Performance evaluation of search engines and their differentiation is very popular issue, lots of possible solutions have been proposed but still satisfactory results are not achieved. Researchers followed the statistical way to evaluate the search engines and to select best search engine from various available all. However, for the task of seeking information, these statistical techniques have indeed proven to become the most effective and efficient ones so for. Statistical model, that consisting Vector Space Model [1,2,6,7] and Probabilistic model [2] helped much and became the base line for their framework and algorithms. The Boolean model [2] that is also known as "exact match" model is still being used by most of the online services. In the process of information retrieval two key problems still exist: First, information retrieval process fetch some irrelevant documents together with relevant document. Second, search engines are not capable to perform retrieval of all relevant documents [3].

1.1 RETRIEVAL EFFECTIVENESS
The quality of the results returned by a system in response to a query is measure to use precision and recall. Precision [3] is the number of relevant documents retrieved divided by the total number of documents retrieved. Recall is the number of relevant documents retrieved divided by the total number of relevant documents. The recall and precision should both be equal to one, meaning that the system returns all relevant documents without introducing any irrelevant documents in the result set. This is impossible to achieve in practice.

1.2 EVALUATIVE MODELS
The goal of information retrieval (IR) is to provide users with those documents that will satisfy their information need. Make the information retrieval to be efficient, the documents are typically transformed into a suitable representation. Now such type of information is retrieve efficiently with help of IR models [2]. The models are categorized according the properties of the model. The following major models have been developed to retrieve information: the Set-Theory model, the Statistical model, which includes the vector space and the probabilistic model. Set-theoretic models represent documents as sets of words or phrases. Similarities are usually derived from set-theoretic operations on those sets. Common models are: Standard Boolean model, Smart Boolean and Extended Boolean model. The Boolean model [2] is based on Boolean logic and classical set- theory in that both the documents to be searched and the user's query are conceived as sets of terms. Retrieval is based on whether or not the documents contain the query terms. The Boolean model represents documents by a set of index terms, each of which is viewed as a Boolean variable and valued as True if it is present in a document. Boolean model cannot rank documents in decreasing order of relevance. The vector space and probabilistic models are the two major examples of the statistical retrieval approach. Both models use statistical information in the form of term frequencies to determine the relevance of documents with respect to a query. Although they differ in the way they use the term frequencies, both produce as their output a list of documents ranked by their estimated relevance. The vector space model [2] has been widely used in the traditional IR field. Most search engines also use similarity measures based on this model to rank web documents.

2. VECTOR SPACE MODEL
The vector space model represents documents and queries as vectors in multidimensional space, whose dimensions are the terms used to build an index to represent the documents. It is used in information retrieval, indexing and relevancy rankings and can be successfully used in evaluation of web search engines. The vector space model procedure can be divided in to three stages. The first stage is the document indexing where content bearing terms are extracted from the document text. The second stage is the weighting of the indexed terms to enhance retrieval of document relevant to the user. The last stage ranks the document with respect to the query according to a similarity measure. A common similarity measure known as cosine measure determines angle between the document vector and the query vector. The angle between two vectors is considered as a measure of divergence between the vectors, cosine angle is used to calculate the numeric similarity, determines angle between the document vector and the query vector when they are represented in V-dimensional Euclidian space where V is the

size. The similarity function [4] between a document vector Di and query Q is as,

$$\text{Cosine}\theta = \text{Sim}(Q, D_i) = \frac{\sum_{j=1}^{V} w_{Q,j} \times w_{i,j}}{\sqrt{\sum_{j=1}^{V} w_{Q,j}^2} \times \sqrt{\sum_{j=1}^{V} w_{i,j}^2}} \quad (1)$$

Where $w_{Q,j}$ is the weight of term j in the query, and is defined similar way as $w_{i,j}$ (that is, $tf_{Q,j} \times idfj$).

The term weighting scheme plays an important role for similarity measure.

The weight of term in document vector can be determined using Tf × Idf method. The weight of term is measured how often the term j occurs in the document i (the term frequency $tf_{i,j}$) and in the whole document collection (the document frequency df_j(number of documents containing term j)). The weight of a term j in the document i is:

$$w_{i,j} = tf_{i,j} \times idf = tf_{i,j} \times \log \frac{D}{df_j} \quad (2)$$

Where D is the number of documents in the document collection and Idf stands for inverse document frequency.

Now from Eq2 different term weighting models have been derived (Tf only, Idf only, and combination of these).So based on term weighting different approaches of vector space model have been discussed as:

2.1 Term –Count Model

In Term count model weights have been computed using local information only.

$$\text{Term Weight} = w_i = tf_{i,j} \quad (3)$$

In this model, and as with any weighting scheme, we need database collection to retrieve documents, query and index term. Let us use following example.

We use query "March health awareness" from Trec query dataset on IR system. Here we consider only three documents from number of Document retrieved as:

D1: "the Health Observances for March"
D2: "the Health oriented Calendar"
D3: "the good News for March Awareness"
Retrieval results are summarized in Table 1.0.

Table 1: WEIGHT based on term count & idf value

QUERY**: "**march health awareness"
D1: "the Health Observances for March "
D2: "the Health oriented Calendar"
D3: "the Awareness News for March Awareness " D=3 IDF= $\log(\frac{D}{df_j})$ df_j =number of documents containing term j

		COUNTS $TF_{I,J}$						WEIGHTS, $W_{I=}TF_{I,J} * IDF_J$			
TERMS	Q	D1	D2	D3	df_j	D/df_j	IDF_J	Q	D1	D2	D3
Health	1	1	1	0	2	3/2=1.5	0.1761	0.1761	0.1761	0.1761	0
Observances	0	1	0	0	1	3/1=3	0.4771	0	0.4771	0	0
For	0	1	0	1	2	3/2=1.5	0.1761	0	0.1761	0	0.1761
March	1	1	0	1	2	3/2=1.5	0.1761	0.1761	0.1761	0	0.1761
Awareness	1	0	0	2	1	3/1=3	0.4771	0.4771	0	0	0.9542
Oriented	0	0	1	0	1	3/1=3	0.4771	0	0	0.4771	0
Calendar	0	0	1	0	1	3/1=3	0.4771	0	0	0.4771	0
News	0	0	0	1	1	3/1=3	0.4771	0	0	0	0.4771
The	0	1	1	1	3	3/3	0	0	0	0	0

COMPUTING SIMILARITY score

To calculate the magnitude of each vector since vector has magnitude and direction, we apply Pythagoras's Theorem. For n dimensions, we can write |Di| = (a12 + a22 + a32...+ an2) 1/2.

$$|D_1| = \sqrt{1^2 + 1^2 + 1^2 + 1^2 + 1^2} = \sqrt{5} = 2.236$$

$$|D_2| = \sqrt{1^2 + 1^2 + 1^2 + 1^2} = \sqrt{4} = 2$$

$$|D_3| = \sqrt{1^2 + 1^2 + 2^2 + 1^2 + 1^2} = \sqrt{8} = 2.828$$

$$|Q| = \sqrt{1^2 + 1^2 + 1^2} = \sqrt{3} = 1.732$$

In our example the dot product is just the sum of products between term and query counts.

Dot product is Q. Di.

$\mathbf{Q \cdot D_{1=}}$ 1*1+1*1+ 1*0=2

$\mathbf{Q \cdot D_{2=}}$ 1*1+1*0+1*0=1

$\mathbf{Q \cdot D_{3=}}$ 1*1+1*0+1*2=3

Since a dot product is defined as the product of the magnitudes of vectors times the cosine angle between these,

Dot product = (magnitudes product)*(cosine angle)

Now solving for the cosine angle which gives similarity score between documents and query as. $\text{Cosine}\theta = \text{Sim}(Q, D_i) = \frac{Q \cdot D_i}{|Q| \times |D_i|}$

$$\mathbf{cosine\theta_{d_1}} = \frac{\mathbf{Q \cdot D_1}}{|\mathbf{Q}| * |\mathbf{D_1}|} = \frac{\mathbf{2}}{\mathbf{1.732 * 2.236}} = 0.5165$$

rank1

$$\mathbf{cosine\theta_{D_2}} = \frac{\mathbf{Q \cdot D_2}}{|\mathbf{Q}| * |\mathbf{D_2}|} = \frac{\mathbf{1}}{\mathbf{1.732 * 2}} = 0.2886$$

rank3

$$\mathbf{cosine\theta_{D_3}} = \frac{\mathbf{Q \cdot D_3}}{\mathbf{|Q| * |D_3|}} = \frac{\mathbf{3}}{\mathbf{1.732 * 2.828}} \quad =0.6124$$

rank2

The observation here is that cosine angle is near to 1, the documents are more similar to query terms.

2.2 TF-IDF (CLASSICAL) VECTOR SPACE MODEL

In information retrieval or text mining, the term frequency – inverse document frequency also called Tf-Idf, is a well-known method to evaluate how important is a word in a document. It is often used as weighting factor in information retrieval.Tf-Idf [5] is also a very interesting way to convert the textual representation of information into a Vector Space Model (VSM), or into sparse features. Unlike the Term Count Model, Tf-Idf incorporates local and global information as shown in Eq2; hence weighting schemes use both local and global information.

COMPUTING SIMILARITY SCORE

Using information from Table 1.0 and Eq1 we can calculate cosine value as below.

$$|D_i| = \sqrt{\sum_i w_{i,j}^2}$$

$|D_1| = \sqrt{0.1761^2 + 0.4771^2 + 0.4771^2 + 0.1761^2} = \sqrt{0.3206 =}$ 0.7192

$|D_2| = \sqrt{0.1761^2 + 0.4771^2 + 0.4771^2} = \sqrt{0.4862 =}$ 0.6973

$|D_3| = \sqrt{0.1761^2 + 0.1761^2 + 0.9542^2 + 0.4771^2} =$ $\sqrt{1.2001 =}$ 1.0955

$$|Q| = \sqrt{\sum_i w_{Q,j}^2}$$

$|Q| = \sqrt{0.1761^2 + 0.1761 + 0.4771^2} = \sqrt{0.2896 =}$ 0.5381

$$Q.D_i = \sum_i w_{Q,j} \times w_{i,j}$$

$\mathbf{Q \cdot D_{1=}}$ 0.1761*0.1761+0.1761*0.1761 =0.0620

$\mathbf{Q \cdot D_{2=}}$ 0.1761*0.1761=0.0310

$\mathbf{Q.D_3}$ =0.1761*0.1761+0.4771*0.9542=0.4862

$$\therefore \text{Cosine}\theta = \text{Sim}(Q, D_i) = \frac{Q \cdot D_i}{|Q| \times |D_i|}$$

$$\mathbf{cosine\theta_{d_1}} = \frac{\mathbf{Q \cdot D_1}}{\mathbf{|Q| * |D_1|}} = \frac{\mathbf{0.0620}}{\mathbf{0.5381 * 0.5663}} \quad =0.1602$$

rank2

$$\mathbf{cosine\theta_{d_2}} = \frac{\mathbf{Q \cdot D_2}}{\mathbf{|Q| * |D_2|}} = \frac{\mathbf{0.0310}}{\mathbf{0.5381 * 0.6973}} \quad =0.0826$$

rank3

$$\mathbf{cosine\theta_{d3}} = \frac{\mathbf{Q \cdot D_3}}{\mathbf{|Q| * |D_3|}} = \frac{\mathbf{0.4862}}{\mathbf{0.5381 * 1.0955}} \quad =0.8249$$

rank1

On the basis of cosine value calculated above this model ranks the retrieved documents on similarity score and this model is based on global information; i.e., the Idf (inverse document frequency).

2.3 VECTOR SPACE MODEL BASED ON NORMALISED FREQUENCY

In term count model and Tf-Idf model the weight of term is considered proportional to its term frequency (tfi,j).Since terms with high occurrences are assigned more weight than term repeated few times. Tf-Idf model also consider the Idf weight so to calculate the length of query vector needs to access to every document term hence needs to normalization. In Idf model length of query vector and document vector can be normalized by normalizing the document and query frequencies.

The normalized frequency of a term j in document i is given as:

$$f_{i,j} = \frac{tf_{i,j}}{\max tf_{i,j}} \qquad (4)$$

$f_{i,j}$ = normalized frequency.

$tf_{i,j}$ = frequency of term j in documents i. max $tf_{i,j}$= maximum frequencies of term j in document i. The normalized frequency of a term j in a query Q is given as:

$$f_{Q,j} = 0.5 + 0.5 * \frac{tf_{Q,j}}{\max tf_{Q,j}} \qquad (5)$$

$f_{Q,j}$ = normalized frequency. $tf_{Q,j}$ = frequency of term j in query Q. max $tf_{Q,j}$= maximum frequency of term j in query Q

Based on query "March health awareness" and using normalized document and query frequency.

Retrieval results are summarized in the following table 2.0.

Table 2: Vector Space Model Based on Normalized Frequencies

QUERY**: "**march health awareness"

D1: "the Health Observances for March "

D2: "the Health oriented Calendar"

D3: "the Awareness News for March Awareness"

D=3 ; ; IDF=$\mathbf{log}(\frac{\mathbf{D}}{df_j})$ df_j = number of documents containing term j

		COUNTS $TF_{I,J}$						WEIGHTS, $W_{I=}$ $F_{Q,J} * IDF_J$	WEIGHTS $W_{I=}F_{I,J} * IDF_J$		
TERMS	Q	D1	D2	D3	df_j	D/df_j	IDF_J	Q	D1	D2	D3
Health	1	1	1	0	2	3/2=1.5	0.1761	0.1761	0.1761	0.1761	0
Observances	0	1	0	0	1	3/1=3	0.4771	0	0.4771	0	0
For	0	1	0	1	2	3/2=1.5	0.1761	0	0.1761	0	0.0881
March	1	1	0	1	2	3/2=1.5	0.1761	0.1761	0.1761	0	0.0881
Awareness	1	0	0	2	1	3/1=3	0.4771	0.4771	0	0	0.4771
Oriented	0	0	1	0	1	3/1=3	0.4771	0	0	0.4771	0
Calendar	0	0	1	0	1	3/1=3	0.4771	0	0	0.4771	0
News	0	0	0	1	1	3/1=3	0.4771	0	0	0	0.2285
the	0	1	1	1	3	3/3=1	0	0	0	0	0

Where $F_{i,\ j\ and}\ F_{Q,\ j}$ is normalized document and query frequency calculated as Eq 4 and Eq 5 respectively.

COMPUTING SIMILARITY VALUE

Using Table 2.0 and value calculated using Tf-Idf model, we have calculated similarity values as:

Since there is small change in the result of Table 1.0 as calculated in Table 2.0 using normalized document and query frequency for this particular query and document discussed above? So only change in document weight and dot product in document3 due to repetitive query term in document3 .Now value calculated as:

$|D_3| = \sqrt{0.0881^2 + 0.0881^2 + 0.4771^2 + 0.2386^2} = \sqrt{0.3000} = 0.5478$

$\mathbf{Q}.\mathbf{D_3}$ =0.1761*0.0881+0.4771*0.4771=0.2431

$\mathbf{cosine\theta_{d_1}} = \frac{\mathbf{Q \cdot D_1}}{\mathbf{|Q| * |D_1|}} = \frac{\mathbf{0.0620}}{\mathbf{0.5381 * 0.5663}}$ = 0.2035

rank2

$\mathbf{cosine\theta_{d_2}} = \frac{\mathbf{Q \cdot D_2}}{\mathbf{|Q| * |D_2|}} = \frac{\mathbf{0.0310}}{\mathbf{0.5381 * 0.6973}}$ =0.0826

rank3

$cosine\theta_{d3} = \frac{Q \cdot D_3}{|Q| * |D_3|} = \frac{0.2431}{0.5381 * 0.5478}$ = 0.8246

rank1

3. DISCUSSION AND COMPARISON

Based on the experiments using three approaches of the vector space model, we have made certain observations. The term frequency in term count model is simply the number of times a given term appears in the document and query. Comparison of three approaches of VSM as presented in fig-1 shows that Term-count model provides good results for long documents as compared to small documents, since these contain many words that are very often repeated. Thus, for long documents similarity scores will be higher. In Tf-Idf model, tf×idf weight is numerical statistics which shows the importance of words in the document. It can be successfully used for stop-words (a, an, the etc.) filtering which is so common so that it can give enough weight to meaningful terms. It is shown in table 1.0, the very frequent terms such as "the" shows low weight (a value zero in our case) if it is available in all documents. So in Tf-Idf model, documents weight for non-meaningful words is discarded (low dimensionality) due to this the similarity score is increased. That is why Tf-Idf model gives better result for long documents as compared to term-count model. The VSM model based on normalized frequency gives same result as compared to Tf-idf model for long documents because when the documents frequency is normalized it reduces the documents weight. The low documents weight increased in the similarity score and thus makes a document more near to the query term. Since similarity score (cosine angle) near to 1, the documents will be more close to query terms. These three approaches of vector space model favor only long documents where the documents contain more appearance of query terms.

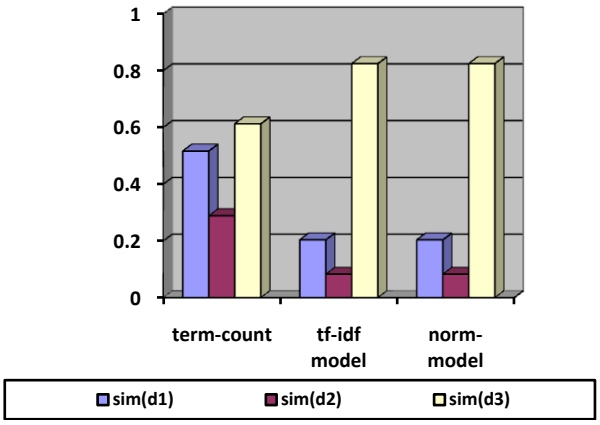

Fig.1: Similarity values for three method of VSM

These three approaches of VSM rank the documents with their similarity score .The documents having zero scores are non-relevant and assigned the lowest possible rank. Therefore, their ranks are required for calculating precision and recall. Precision and recall values are used in measure the relevancy of documents. The retrieved document set is examined from the top.

4. CONCLUSIONS

In this paper we have analyzed three approaches of vector space model for test retrieval queries. The similarity value is calculated by using three approaches of vector space model. After considering the weighting terms in document collection, we can calculate similarity value between queries and documents. Ranking of documents depend upon score of similarity value calculated by different approaches of VSM. A similarity value calculated by term-count model is good for long documents but Tf-Idf and normalization model provide better result for same. Normalization model also uses global weighting scheme and provide same result for long documents compare to Tf-Idf model. So all three approaches of vector space model may favor long documents that contain more appearance of the query terms.

5. REFERENCES

[1] Shalton, G; Wong, A; Yang, C.S.: A vector space Model for automatic indexing Communications of The ACM, Volume 18, Issue 11(November1975).

[2] Sanjay K. Dwivedi,Jitendra Nath Singh, Rajesh Gotam "Information Retrieval Evaluative Model" FTICT 2011: Proceedings of the 2011, International conference on "Future Trend in Information & Communication Technology, Ghaziabad, India, Feb -2011.

[3] Yi Shang Longzhuang Li: Precision Evaluation of Search Engines. World Wide Web (2002).

[4] D.L. Lee, H. Chuang, and K. Seamons. Document ranking and the vector space model. IEEE Transactions on Software, 14(2): 1997.

[5] Chris Buckley. The importance of proper weighting methods. In M. Bates, editor, Human Language Technology. Morgan Kaufman, 1993.

[6] Longzhuang Li, Yi Shang A new statistical method for performance evaluation of search engines. ICTAI 2000.

[7] Longzhuang Li, Yi Shang A new method for automatic performance comparison of search engines. World Wide Web (2000).

[8] Chu, H. & Rosenthal: "Search engines for the World Wide Web: A comparative study and evaluation methodology". In Proceedings of the 59th Annual Meeting of the American Society for Information Science, Baltimore, 1996.

[9] Jinbiao Hou: "Research on Design of an Automatic Evaluation System of Search Engine" .In proceeding of ETP International Conference on Future Computer and Communication .FCC/2009.

[10] Gerald Salton and Chris Buckley. Term weighting approaches in automatic text retrieval. Information Processing and Management, 24(5): Is-sue 5. 1988.

[11] G. Salton and C. Buckley, "Improving Retrieval Performance by Relevance Feedback," J. Amer. Soc.for Information Science, Vol. 41, No. 4, 1990