Kuwait J. Sci. **44 (4)** pp 49-56, 2017

Trade-off between the number of index-terms and the information retrieval system's performance

Srinvasarengan Lakshmi[1], Balasubramanian Sathiyabhama[2], Krishnan Batri[3],*

[1]*Dept. of Electronics and Communication Engineering, RVS College of Engineering and Technology, Dindigul lakshmichandra18@yahoo.co.in*

[2]*Dept.of Computer Science and Engineering, SONA College of Technology, Salem sathiyabhama@sonatech.ac.in*

[3]*Dept. of Electronics and Communication Engineering PSNA College of Engineering and Technology, Dindigul krishnan.batri@gmail.com*

**Corresponding author: krishnan.batri@gmail.com*

Abstract

Performance of modern day information retrieval (IR) systems depends on the index terms and their occurrence frequency. Hence, a small variation in the frequency of index terms alters the performance of IR systems. This article analyzes the variation in performance of IR systems due to changes in the frequency of index terms. Based on the occurrence frequency, we classified the index terms as `Low' and `High' frequency terms; their performances were also recorded. Low-frequency terms tend to decrease the performance of IR systems. In contrast, the performance of high-frequency terms is better than its counterpart. High-frequency terms do 10% performance improvement in comparison with the low-frequency terms. By deleting the low-frequency index terms, we can save up to 65% of index terms with a maximum of 26% degradation in performance of IR systems.

Keywords: High-frequency; index term; information retrieval; low-frequency; term frequency.

1. Introduction

Information retrieval (IR) deals with the processing, organizing, storing, and retrieving of documents (Baeza-Yates *et al*.,1999). All these above-mentioned steps are supported by the mathematical concepts and expressions. The processing step deals with the identification of index terms, and it extracts the important features of about the terms and the documents (Berger & Lafferty, 1999).

Processing steps extract the important features of index terms and documents. The index terms are not treated equally. The discrimination is based on importance of the terms. The importance is calculated based on the number of occurrences (Baayen, 2001). The term, which is having high occurrence frequency, seems to be more important than its counterpart. Once the index terms are assigned with weights, it can be further used by various IR models.

The index terms and their weights are used to judge the relevance of the documents, while we are posting the query. Do we need all index terms, which are present in a document for representing that particular document? Is it enough to use the part of the index terms? What are they? How can we find them? Or, how can we predict them? These questions lead to a new area called dimensionality reduction in IR (Ciarelli & Oliveira, 2009). The dimensionality reduction develops a method/algorithm for predicting the non-important index terms. We can save a lot of space and computation time by using this prediction. The dimensionality reduction in IR tries to build a trade-off between the IR system's performance, and the number of index terms.

1.1. Motivation of this work

The number of web pages indexed by the search engines is 4.59 billion (as of April 26, 2016). It increases at a rate of 10% per year. Hence, within a decade, it will be 10 billion web pages. The search engines, which are processing these pages, need huge storage space. Apart from this, the computation cost of these search engines also increases. We want to reduce the storage space and computation time. It directly depends on the index terms. Hence, by reducing the number of index terms, we can reduce the storage space and the computation time of the search engines. This becomes the motivation for this work.

1.2. Objective of this work

The main objective of this work contains 3 parts. They are:

1. To test whether the deletion of index terms affect the performance of IR systems.
2. To establish the importance of the index terms based on their occurrence frequency.
3. To measure the correlation between the total number of index terms, and the IR system's performance.

2. Literature survey

Documents can be represented using syntactic and semantic methods (Luhan, 1957). Automation of semantic methods is still striving for success (Manning & Schutze, 1999). Syntactic methods use the statistical translation for automation (Harman, 1986; Quan *et al*., 2011). They discriminate the index terms, based on their importance. The importance of the index terms is treated as weights (Aizawa, 2003; Hiemstra, 2000). The importance is based on term frequency, inverse document frequency etc. These factors are used in their raw format or in their transferred format (Wu & Salton, 1981). Out of these above-mentioned factors, term frequency becomes important (Harman, 1986).

The merits of term frequency made it as an inevitable factor in IR and it becomes the fundamental part of the weighting scheme (Quan *et al*., 2011; Salton & Yang, 1973). The demerits lead to a search for alternatives. Combining the other weighting schemes with the term frequency was one of the alternatives (Harman, 1986). The combination process may be multiplicative or additive (Wu & Salton,1981). The additive to the term frequency weighting may increase or decrease the performance. As a result, some tuning has to be made (Yu *et al*., 1982). Whether the weighting scheme is singular or combined, the term frequency becomes the focus point.

Performance analysis of the term frequency-based weighting has been supported by probability and information theory (Berger & Lafferty, 1999; Hiemstra, 2000; Robertson, 2004). The term and document selection are treated as an equally likely and independent event (Robertson, 2004). The uncertainty associated with the term and document selection is calculated using entropy and it gives a clear picture about the term and term frequency (Aizawa, 2003; Wu *et al*., 2008)).

Some efforts have been carried out to minimize the number of index terms without affecting the performance. LSI is used as a well-known and effective method (Moravec *et al*., 2004). Since LSI is deleting some index terms, it may lead to the system degradation (Moravec *et al*., 2004). The researchers are trying to develop a mechanism, which properly selects the index terms for deletion (Berka & Vajtersic, 2013). The importance of index terms can be calculated either by the supervised or unsupervised way (Karypis & Han, 2000). Both these methods tend to identify the non-contributing index terms (Saleh & Weigang, 2015). Once, it has been identified, those terms are removed. Once the terms are deleted, they may degrade the IR system's performance. The replacement algorithm tries to overcome this performance degradation (Saleh & Weigang, 2015).

In the unsupervised method, the identification of the non-contributing terms becomes a tedious task, as there is no assistance. But the supervised method has some advantages. Hence, they are mostly used in document classification. The unsupervised method does not need any feedback. The computation time is also minimal. Hence, we want to focus on the unsupervised method.

3. Importance of index terms

The vector space model (VSM) is a most prominent one, and it is widely used. The VSM assumes that the documents and the terms are multidimensional vectors. The VSM treats the index terms as the independent entity (Baeza-Yates *et al*., 1999). It considers the document as a bag-of-words. The bag-of-words model uses the multi-set concepts. We used VSM and bag-of-words model in our experiment.

Consider a collection of documents (corpus) `c'. The corpus `c' can be represented as

$$c = \{d_1, d_2, d_3, \dots . d_N\} \tag{1}$$

Where,

c -corpus,

d -document,

N -total number of documents in the corpus.

A document is made up of words. Some words are repeated. The repeated words are replaced with a single entry, and their repetition is calculated as term frequency.

The same corpus `c' may be represented as

$$c = \{t_1, t_2, t_3, \dots . t_m\} \tag{2}$$

where,

t -index term,

m-total number of unique index terms in the corpus.

Now, consider a document ‘d_i’, whose total number of unique index term is ‘x_i’. The document `d_i’ can be represented as

$$d_i = \{t_1, t_2, t_3, \dots \dots . t_{x_i}\} \tag{3}$$

Where,

d_i-i^{th} document in the corpus,

x_i -total number of unique index terms in the i^{th} document.

Let `si’ be the set of unique index terms of a document `di’, then $x_i = |s_i|$. We can represent the index terms present in the corpus `c’ as

$$s_1 \cup s_2 \cup \dots \cup s_N \tag{4}$$

The Equations (3) and (4) can be related as

$$s_1 \cup s_2 \cup \dots \cup s_N = \{t_1, t_2, t_3, \dots, t_m\} \tag{5}$$

`x’ and `m’ can be related as

$$|s_1 \cup s_2 \cup \dots \cup s_N| = m \tag{6}$$

Now, consider an index term. The same index term may be in more than one document. The number of document, which contains the index term `$t_i$’ is given by `Z_i’.

Now, consider a query q. Apply the same VSM concept to the query and the query may be represented as

$$q = \{t_1, t_2, t_3, \dots, t_k\} \tag{7}$$

A term `$t_i$’, which is present in the corpus may be relevant to the query. Probability of a term `t_i’ relevant to the query is giving by

$$p_{ts}(t_i) = \frac{1}{m} \tag{8}$$

This term `t_i’ may present in many documents. Probability of selecting jth document is given by

$$p_{ds}(d_i) = \frac{x_j}{m} \tag{9}$$

Now, consider Equation (9). If the document `$d_j$’ contains all index terms (i.e. $x_j = m$), then the probability of selecting that document `d_j’ is equal to 1. Hence, the probability of selecting a document is more, if it has more index terms.

The probability of selecting a document `dj’ after selecting the index term `ti’ is given as:

$$p_{ds}(d_j)/p_{ts}(t_i) = \frac{x_j}{m} . \frac{1}{m} = \frac{x_j}{m^2} \tag{10}$$

The above assumption is applicable for a single index term. The query may contain more than one index term. In that case, the probability of selecting a document, which is having all query terms (k) is given by

$$p_{ds}(d_i) = k . \frac{x_j}{m^2} \tag{11}$$

4. Testing the importance of index terms

The IR system›s performance at various levels of index terms compared against the IR system›s performance, which contains all index terms. Average precision at full recall level is used as the performance indicator.

4.1. Experimental setup

Experiment is conducted over four test data sets namely i) ADI, ii) CISI, iii) MEDLARIS, and iv) CRANFIELD. Properties of these four data sets are given in the Table 1.

Table 1. Characteristics of the data sets

Properties	ADI	CISI	MED	CRAN
Number of documents	82	1460	1033	1400
Number of terms	374	5743	5831	4486
Average number of document relevant to a query	5	50	23	8
Average number of terms per document	16	45	50	56
Average number of documents per term	4	13	8	16
Average number of terms per query	5	8	10	9

4.1.1. Pre-processing

The four data sets are tokenized. The most common high frequency words are treated as the stop words. These stop words does not convey any information about the document. Hence, these words should be removed before processing the document. The smart stop word list is used for removing these terms (Lewis*et al*., 2004). After removing these stop words, the document may contain some high frequency terms, which are related to that document. We refer those terms as high frequency terms. The other terms are called as low frequency terms. The porter stemmer algorithm is used for stemming purpose (Porter, 1980).

4.1.2. Term-weight

If a term is present in a document, then its weight is `1‘ irrespective of its number of occurrence. It is `0‘ otherwise.

4.1.3. Similarity measure

The similarity between the document and the query is calculated using the conjunctive normal form (CNF) of the extended Boolean model (Salton *et al.*, 1983). `p' value for this conjunctive normal form is set as `1'.

4.2. Results

The average precision of the IR system, when it has all index terms (100%) over the four test data sets is recorded, and it is used as the reference. Now, we want to reduce the index terms' level. We opted for 98%, 95%, 92%, 90%, 80%, 70%, 60%, and 50% level. The terms which are going to be present are selected randomly. As we use the random deletion/selection, we want to repeat the experiment. We repeated for `50' times.

The average precision at full recall level for various index terms' level is plotted and it is given in the Figure 1.

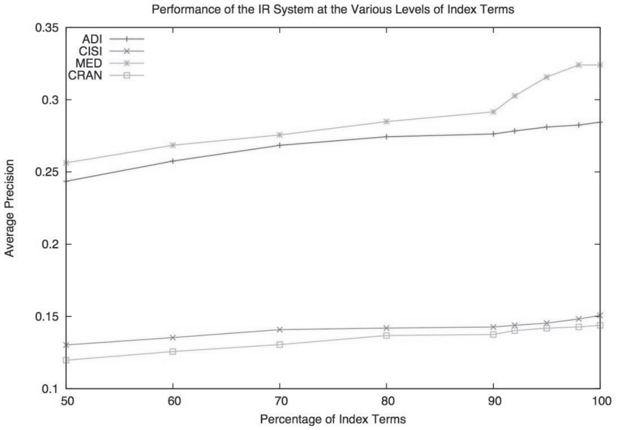

Fig.1. Variation in IR system's performance due to index level variation

The performance degradation at various levels of index terms is compared against the full index term level which is calculated and given in Figure 2.

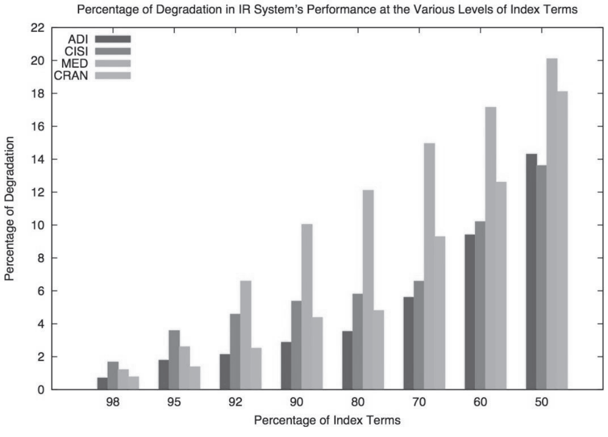

Fig. 2. Percentage of degradation in IR System's performance due to index level variation

From Figure 1, it has been identified that index terms and the IR system's performance are having a positive correlation.

Simple student t-test is used to test the significance of the result. The hypothesis used in the student t-test is given below:

H_0: $\mu \leq$ Average precision computed for the reduced index terms.

H_1: $\mu >$ Average precision computed for the reduced index terms.

The μ value represents the precision value for the full index terms. The computed `t' value is given in the Table 2.

Table 2. Computed 't' value

Index term level	ADI	CISI	MED	CRAN
98%	8.43	5.37	4.38	5.43
95%	4.65	7.48	5.23	3.89
92%	7.58	6.57	7.42	5.14
90%	5.32	8.72	8.56	9.47
80%	6.3	7.46	9.4	7.43
70%	4.85	5.83	8.3	6.48
60%	5.43	4.73	4.3	5.83
50%	6.58	7.8	7.6	7.4

The null hypothesis is successfully rejected at 1% confidence level.

5. Term frequency and its importance

Consider a corpus `c'. Let `m' be the total number of unique index terms present in the corpus. Let the average term frequency of the corpus be `*Avgtf(c)*'. The total number of index terms present in the corpus may be expressed as `m .*Avgtf(c)*'.

The probability of selecting a term `ti' based on its term frequency is given as

$$p_{ts}(t_i) = \frac{tf(t_i)}{m.Avgtf(c)} \qquad (12)$$

Now consider a document `dj'. Number of unique index terms present in the document `d_j' be `$x_j$'. Average term frequency of the document `dj' is given as `*Avgtf(dj)*'. The total number of index terms present in the document `d_j' is expressed as `$x_j$ .*Avgtf*($d_j$)'. Based on the above discussion, the probability of selecting the document `d_j' is given as

$$p_{ds}(d_j) = \frac{x_{j.}}{m} \frac{Avgtf(d_j)}{Avgtf(c)} \tag{13}$$

Now, consider the selection of a document after selecting a term; the probability of selection is given as

$$p_{ds}(d_j)/p_{ts}(t_i) = \frac{tf(t_i)x_{j.}}{m^2} \frac{Avgtf(d_j)}{Avgtf^2(c)} \tag{14}$$

The term frequency of the term `ti' can be represented using `*Avgtf*(d_j)' as

$$tf(t_i) = Avgtf(d_j) + l_{ij} \tag{15}$$

The `lij' is the difference between the average term frequency of the document `dj' and the term frequency of the term `ti' in the particular document `dj'. The value of `lij' is either positive or negative. Based on the modification, Equation (14) can be re-written as

$$p_{ds}(d_j)/p_{ts}(t_i) = \frac{(Avgtf(d_j)+l_{ij})x_{j.}}{m^2} \frac{Avgtf(d_j)}{Avgtf^2(c)}$$

$$= \frac{(Avgtf^2(d_j)x_j + l_{ij})x_{j.}}{m^2} \frac{Avgtf(d_j)}{Avgtf^2(c)} \tag{16}$$

From the above equation, it has been identified that, if `lij' is positive, then the probability of selecting a document is also high. If it is negative, then the chance of selection is low. Based on the above, we came to a conclusion that the terms, which are having the above-average occurrence frequency, will increase the selection probability.

Assume a document `dj', which is having a total of `k' relevant index terms. The probability of selection is given by

$$p_{ds}(d_j)/p_{ts}(t_i) = \frac{x_{j.}}{m^2} \frac{Avgtf^2(d_j)+l_{1j.}x_{j.}.Avgtf(d_j)}{Avgtf^2(c)} + \frac{x_{j.}}{m^2} \frac{Avgtf^2(d_j)+l_{2j.}x_{j.}.Avgtf(d_j)}{Avgtf^2(c)} + \cdots + \frac{x_{j.}}{m^2} \frac{Avgtf^2(d_j)+l_{kj.}x_{j.}.Avgtf(d_j)}{Avgtf^2(c)} \tag{17}$$

Now, we try to separate the above, and below average index terms. Assume that the terms are arranged in descending order based on their occurrence frequency. Let `a' be the total number of above-average index terms, and `b' be the total number of below average index terms. `a', and `b' can be related as a + b = k. Based on above, Equation (17) can be re-written as

$$p_{ds}(d_j)/p_{ts}(t_i) = \frac{k.x_{j.}}{m^2} \frac{Avgtf^2(d_j)}{Avgtf^2(c)} + \frac{x_{j.}}{m^2} \frac{Avgtf(d_j)\sum_{c=1}^{k-b} l_{cj.}x_{j.}.}{Avgtf^2(c)} - \frac{x_{j.}}{m^2} \frac{Avgtf(d_j)\sum_{c=k-b+1}^{k} l_{cj.}}{Avgtf^2(c)} \tag{18}$$

The above Equation (18) gives the probability of selecting the document based on term frequency. *The probability of selecting a document is high if it satisfies the following two conditions: (i) It should possess the index terms present in the query, and (ii) The term frequency of those terms should be higher than the average term frequency of that document.*

6. Testing the importance of term frequency

Experiments are conducted over the same four test data sets used in section 4.

6.1. Term weight

The term frequency is used as the term weight. Other forms of term weighting are not used, because we want to analyze the importance of the term frequency alone.

6.2. Similarity measure

Inner product of the vector space model (VSM) is used as the similarity measure. The inner product is selected over other schemes because it directly calculates the correlation between query terms and the document terms. As we used the term frequency as weighting scheme, the inner product indirectly calculates the correlation between the document term frequency and the query term frequency. It fulfills our requirement. Hence, we selected the inner product over the other similarity measures. Formula used for inner product calculation is given by

$$Inner\ product\ S(q,d) = \sum_{i\epsilon q \cap d} w_{d,t} * w_{q,t} \tag{19}$$

Results obtained for the above experiment for all four data sets are given in following Figure 3.

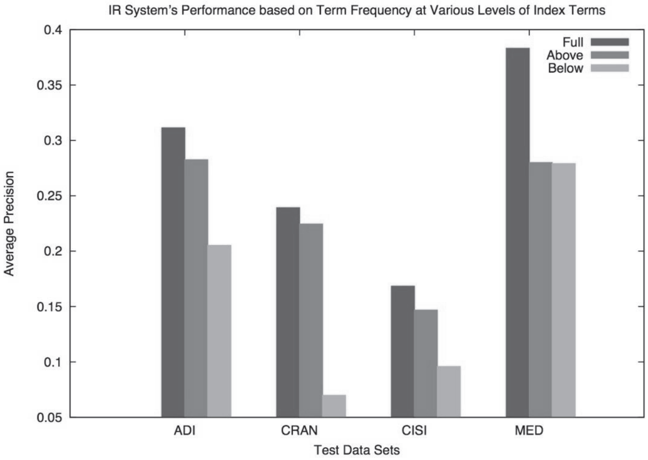

Fig. 3. Variation in IR system's performance due to index term frequency

Figure 3 shows the average precision at 100% recall level. The variation in performance is expressed in terms of percentage of deviation and it is plotted in the Figure 4.

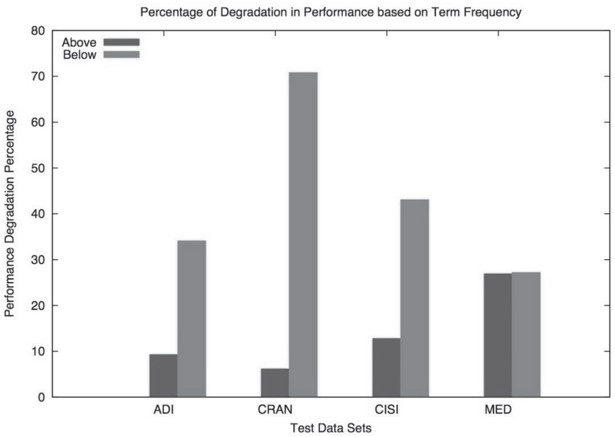

Fig.4. Percentage of degradation in IR system's performance due to index term frequency

s

The variation in performance is much higher for the below average index terms, when compared with the above-average index terms. Figures3 and 4 show the overall consolidated results. We want to analyze the performance for each and every query. For this analysis, we took all queries in the four data sets.

The results are computed for all queries. The experiment is repeated for full, above-average, and below-average index terms. As the experiment is conducted over all queries, we want to confirm the results for each and every query. One-way ANOVA is used for this purpose. The results of the one-way ANOVA are given in Table 3.

Table 3. Results of One way ANOVA

Collection	Sig Level	F-Value	DF
ADI	0.135	2.044	2,103
CISI	0.065	2.809	2,102
MED	0.019	4.150	2,87
CRAN	0.001	74.599	2,672

The `F› value confirms the significant difference among the three sets. Out of these three result sets, we want to compare the individual sets. Bonferroni posthoc test is used for this purpose. The results of Bonferroni posthoc test are given in Table 4.

Table 4. Results of Bonferroni post hoc test

Collection	Above and Below	Above and Full	Below and Full
ADI	0.478	1.000	0.158
CISI	0.323	1.000	0.069
MED	0.045	1.000	0.042
CRAN	0.001	1.000	0.001

The `p' value is used for comparison purpose. There is a significant variation between the below-average and full level, and below-average and above-average index terms' result. But there is no significant variation between the results of above-average and full level index terms.

Table 5 shows the consolidated list of all terms present in every document over the data sets.

Table 5. Number of above, below average and full level index terms

Collection	Full Level	Above Average	Below Average
ADI	2309	1532	777
CISI	71303	15382	55921
CRAN	82790	26540	56250
MED	56610	6278	50332

The above-average index terms constitute the 32 - 11% of all index terms. From the Table 6, we conclude that the above-average index terms will occupy almost 35% of the index terms.

Table 6. Percentage of above, below and full level index terms

Collection	Above average	Below average
ADI	23.04	76.96
CISI	21.57	78.43
CRAN	32.06	67.94
MED	11.09	88.91

In other words, we can save up to 65% of index terms at a cost of 26% performance degradation (maximum).

7. Conclusion

Importance of index terms and the impact of term frequency have been analyzed in this article. The above-average index terms can be used for indexing purpose and by doing so, we can save a minimum of 50% of the index terms (in our experiment we saved 65%). The above-average index terms still degrade the system performance. We got a maximum of 26% performance degradation. But, the degradation is very minimal; when we compare it with the amount of index terms we saved. In near future, we want to improve the system performance by using the above-average index terms alone.

References

Aizawa, A. (2003). An information-theoretic perspective of tf–idf measures, Information Processing & Management, **39**(1):45–65.

Baayen, R. H. (2001). Word frequency distributions, Vol. 18, Springer Science & Business Media.

Baeza-Yates, R., Ribeiro-Neto, B. et al. (1999). Modern information retrieval, Vol. 463, ACMpress New York.

Berger, A. & Lafferty, J. (1999). Information retrieval as statistical translation, in Proceedingsof the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, ACM, pp. 222–229.

Berka, T. & Vajterˇsic, M. (2013). Parallel rare term vector replacement: Fast and effective dimensionality reduction for text, Journal of Parallel and Distributed Computing,**73**(3):341–351.

Ciarelli, P. M. & Oliveira, E. (2009). Agglomeration and elimination of terms for dimensionality reduction, in 'Intelligent Systems Design and Applications, 2009. ISDA'09. Ninth International Conference on', IEEE, pp. 547–552.

Harman, D. W. (1986). An experimental study of factors important in document ranking in Proceedings of the 9th annual international ACM SIGIR conference on Research and development in information retrieval, ACM, pp. 186–193.

Hiemstra, D. (2000). A probabilistic justification for using tfidf term weighting in information retrieval, International Journal on Digital Libraries, **3**(2):131–139.

Karypis, G. & Han, E.-H.S. (2000). Fast supervised dimensionality reduction algorithm with applications to document categorization & retrieval, in Proceedings of the ninth international conference on Information and knowledge management, ACM, pp. 12–19.

Lewis, D. D., Yang, Y., Rose, T. G. & Li, F. (2004). Smart stop word list, Journal of Machine Learning Research.

Luhn, H. P. (1957). A statistical approach to mechanized encoding and searching of literaryinformation, IBM Journal of research and development, **1**(4):309–317.

Manning, C. D. & Schutze, H. (1999). Foundations of statistical natural processing.

Moravec, P., Kolovrat, M. &Snasel, V. (2004). Lsi vs. wordnet ontology in dimension reduction for information retrieval., in 'Dateso', pp. 18–26.

Porter, M. F. (1980). An algorithm for suffix stripping, Program, **14**(3):130–137.

Quan, X., Wenyin, L. & Qiu, B. (2011). Term weighting schemes for question categorization, Pattern Analysis and Machine Intelligence, IEEE Transactions on, **33** (5):1009–1021.

Robertson, S. (2004). Understanding inverse document frequency: on theoretical argumentsfor idf, Journal of documentation, **60**(5):503–520.

Saleh, A. A. & Weigang, L. (2015). A new variables selection and dimensionality reduction technique coupled with simca method for the classification of text documents, in Proceedings of the MakeLearn and TIIM Joint International Conference, Make Learn and TIIM, pp. 583–591.

Salton, G. & Yang, C.-S. (1973). On the specification of term values in automatic indexing, Journal of documentation, **29** (4):351–372.

Salton, G., Fox, E. A. & Wu, H. (1983). Extended boolean information retrieval, Communicationsof the ACM, **26**(11):1022–1036.

Wu, H. & Salton, G. (1981). A comparison of search term weighting: term relevance vs inverse document frequency, in 'ACM SIGIR Forum', Vol. 16, ACM, pp. 30–39.

Wu, H. C., Luk, R. W. P., Wong, K. F. & Kwok, K. L. (2008). Interpreting tf-idf term weights as making relevance decisions, ACM Transactions on Information Systems (TOIS), **26**(3):13.

Yu, C. T., Lam, K. & Salton, G. (1982). Term weighting in information retrieval using the term precision model, Journal of the ACM (JACM), **29**(1):152–170.

Submitted : 03/11/2015
Revised : 05/07/2016
Accepted : 17/07/2016

العلاقة العكسية بين عدد رموز الفهرس وأداء أنظمة استرجاع المعلومات

سرينفاسارينجان لكشمي[١] ، بالاسبرامانيان ساثيابهاما[٢] ، كريشنان باتري[٣] ، *

[1] قسم هندسة الالكترونيات والاتصالات، كلية RVS للهندسة والتكنولوجيا، دينديجول

[2] قسم علوم الحاسوب والهندسة، كلية سونا للتكنولوجيا، سالم

[3]،* قسم هندسة الالكترونيات والاتصالات، كلية PSNA للهندسة والتكنولوجيا، دينديجول

krishnan.batri@gmail.com

*krishnan.batri@gmail.com

خلاصة

يعتمد أداء نظم استرجاع المعلومات حالياً (IR) على رموز الفهرس وتواتر تكرارها. ومن ثم، فإن أي اختلاف طفيف في تواتر رموز الفهرس يغير من أداء أنظمة استرجاع المعلومات. وهذا المقال يحلل الاختلافات في أداء أنظمة استرجاع المعلومات نتيجة للتغيرات في تكرار رموز الفهرس. واستناداً إلى تواتر التكرار، صنفنا رموز الفهرس إلى رموز «منخفضة التكرار» ورموز «عالية التكرار»؛ وتم تسجيل أدائهما. تميل الرموز منخفضة التكرار إلى خفض أداء أنظمة استرجاع المعلومات. وفي المقابل، فإن أداء الرموز عالية التكرار أفضل من نظيرتها. فالرموز عالية التكرار تؤدي بشكل أفضل من الرموز المنخفضة التكرار بنسبة 10 %. وبحذف رموز الفهرس منخفضة التكرار، يمكننا توفير ما يصل إلى 65% من رموز الفهرس وتوفير التراجع في أداء أنظمة استرجاع المعلومات بنسبة 26% كحد أقصى.