International Journal of Electrical and Computer Engineering (IJECE)
Vol. 7, No. 6, December 2017, pp. 3705~3711
ISSN: 2088-8708, DOI: 10.11591/ijece.v7i6.pp3705-3711

Arabic Book Retrieval using Class and Book Index Based Term Weighting

M. Ali Fauzi[1], Agus Zainal Arifin[2], Anny Yuniarti[3]
[1]Departement of Computer Science, Universitas Brawijaya
[2,3]Departement of Informatics, Institut Teknologi Sepuluh Nopember

Article Info

Article history:

Received Apr 27, 2017
Revised Sep 8, 2017
Accepted Sep 27, 2017

Keyword:

Information retrieval
Term weighting
IBF
Document ranking
Arabic book

ABSTRACT

One of the most common issue in information retrieval is documents ranking. Documents ranking system collects search terms from the user and orderly retrieves documents based on the relevance. Vector space models based on TF.IDF term weighting is the most common method for this topic. In this study, we are concerned with the study of automatic retrieval of Islamic *Fiqh* (Law) book collection. This collection contains many books, each of which has tens to hundreds of pages. Each page of the book is treated as a document that will be ranked based on the user query. We developed class-based indexing method called inverse class frequency (ICF) and book-based indexing method inverse book frequency (IBF) for this Arabic information retrieval. Those method then been incorporated with the previous method so that it becomes TF.IDF.ICF.IBF. The term weighting method also used for feature selection due to high dimensionality of the feature space. This novel method was tested using a dataset from 13 Arabic Fiqh e-books. The experimental results showed that the proposed method have the highest precision, recall, and F-Measure than the other three methods at variations of feature selection. The best performance of this method was obtained when using best 1000 features by precision value of 76%, recall value of 74%, and F-Measure value of 75%.

Corresponding Author:

M Ali Fauzi,
Departement of Computer Science,
Universitas Brawijaya,
Jl Veteran 62102, Malang, Indonesia.
Email: moch.ali.fauzi@ub.ac.id

1. INTRODUCTION

The number of Arabic electronic document is increasing drastically. However, information retrieval (IR) research conducted on Arabic still much less extensive than IR research on English, despite the fact that Arabic is one of the five official and working languages of the United Nations, spoken by over 242 million people, and, because it is the language of the Qur'an, the second language of many Muslims and Muslim countries around the world [1, 2]. The objective of IR is finding the most relevant information in respect to user's need. One of the most common issues in information retrieval is documents ranking [3]. Documents ranking system collects search query from the user and orderly retrieves documentsbased on the relevance [4, 5, 6]. Many researches about Arabic documents ranking have been conducted before. As the ones usedthe vector space model and cosine similarity to implement their proposed work [7, 8]. Some of the other works used N-Gram matching [9, 10], document crawler module with correct morphological meaning feedback [4], and Boyer–Moore–Horspoolbased string matching to handle orthographic variations and vocalization marks [11]. Most other researchers mainly concentrate on building a good Arabic stemmers [12, 13, 14, 15, 16, 17], only a few of them concentrate on the term weighting.

For general information retrieval task, Salton and Buckley [18] found that normalized TF.IDF is the

best document weighting function. Therefore, this term weighting became the most popular term weighting used in Arabic IR such as in [7, 8, 19, 20, 21]. TF.IDF is the product of term frequency (TF) and inverse document frequency (IDF). TF measures the density of a term in a document and IDF estimate the rarity of a term in the whole document collection. TF.IDF weighting that only based on documents are not enough to enhance document indexing process. Generating more informative terms for document index should also consider the rarity of the term in the whole classes. Terms that occurs in many classes should not be an important term despite their high TF.IDF score. Therefore, Ren and Sohrab [22] proposed a novel term weighting scheme for automatic classification task using the combination of document-based and class-based approaches called TF.IDF.ICF and its variations TF.IDF.ICS_{δ}F. In this scheme, the inverse class frequency (ICF) and the inverse class space density frequency (ICS_{δ}F), is incorporated. The experimental results show that the proposed class-indexing-based term weighting approaches outperformed TF.IDF and the other five different term weighting approaches in automatic classification task [22].

In this study, we have developed class-based indexing method that incorporated with document-based indexing method for Arabic information retrieval. Specifically, we are concerned with the study of automatic retrieval of Islamic *Fiqh* (Law) book collection. This collection contains many books, each of which has tens to hundreds of pages. Each page of the book is treated as a document. The retrieval system will rank the book pages based on their relevance to the user search term. This work was implemented using vector space model (VSM) and cosine similarity based on TF.IDF.ICF term weighting. In this work, we also propose a novel book-based indexing method. This method is the semantic version of ICF called inverse book frequency (IBF). We have classified the book pages using statistical classifier to build the ICF term weighting. Therefore, we can call the ICF is using statistical classes while the IBF is using semantic classes. This semantic class is the book title. Some pages (documents) that share the same book title tend to have similar context. The author hadmanually collected documents that discuss the same topic or issue in one book. Nearly similar to the ICF, IBF consider the rarity of the term in the whole book collection. Terms that occurs in many books should not be an important term despite their high TF.IDF.ICF score. The IBF will be incorporated with previous method to be TF.IDF.ICF.IBF. The term weighting method also used for feature selection due to high dimensionality of the feature space.

2. TERM WEIGHTING

Vector space model is a common method used in Information Retrieval system. In vector space model, each documents is represented in a matrix that contains itsterms or words weight. The weight expressed the contributionof a word or term to the document. The main function of a term weighting system is the improvement of retrieval effectiveness. Proper term weighting can greatly improve the performance of the vector space method [23, 24]. There are some popular term weighting method such as TF, TF.IDF and TF.IDF.ICF.

2.1. Term Frequency (TF)

Term frequency is the simplest method in assigning weights to each term. Each term is assumed to have a contribution that is proportional to the number of its occurrences in the document. The weights of term t in document d using normalized TF can be counted as follows:

$$TF(t,d) = 1 + \log\left(f_{t,d}\right) \tag{1}$$

where $f_{t,d}$ is the number of the term t occurrence in the document d.

2.2. Inverse Document Frequency (IDF)

When the term frequency (TF) is based onthe term occurrences in a document, IDF consider the distribution of the term in the corpus. Unlike TF which is a local weighting method, IDF is a global one. The background of this weighting is a rare term in the corpus is very valuable. The value of each term is assumed to has the opposite proportion to the number of documents in the corpus that contain the term.The weights of term t using normalized IDF can be counted as follows:

$$IDF(t) = 1 + \log\left(\frac{N_d}{df_t}\right) \tag{2}$$

where N_d is the number of documents in corpus and df_t is the number of documents in corpus that contains term t.

2.3. Inverse Class Frequency (ICF)

ICF is a global weighting method like IDF. When the IDF consider the distribution of the term appeareance across the documents incorpus, the ICF pay attention to the distribution of the term appeareance across categories / classes. The rare term, the term that only appears in a certain class, have the higher value that the frequent one. The value of each term is assumed to have the opposite proportion to the number of classes that contain the term. The weights of term t using normalized ICF can be counted as follows:

$$ICF(t) = 1 + \log\left(\frac{N_c}{cf_t}\right) \quad (3)$$

where N_c is the number of classes and cf_t is the number ofclasses that contains term t.

2.4. Inverse Book Frequency (IBF)

IBF is a novel term weighting method that we proposed in this paper. Meanwhile ICF pay attention to the distribution of the term appeareance accross classes, the IBF consider the distribution of the term on a collection of books. Term that only appears in ceratin book and rarely appears in other books is a very valuable term. The value of each term is assumed to have the opposite proportion to the number of books that contain the term. The weights of term t using normalized IBF can be counted as follows:

$$IBF(t) = 1 + \log\left(\frac{N_b}{bf_t}\right) \quad (4)$$

where N_b is the number of books and bf_t is the number ofbooks that contains term t.

2.5. TF.IDF.ICF.IBF

TF.IDF.ICF.IBF is a multiplication of TF, IDF, ICF and IBF. The weight combination of term t in document d can be counted as follows:

$$TF \bullet IDF \bullet ICF \bullet IBF(t,d) = TF(t,d) \bullet IDF(t) \bullet ICF(t) \bullet IBF(t) \quad (5)$$

where $TF(t,d)$ is the TF value of term t in document d, $IDF(t)$ is the IDF value of term t, $ICF(t)$ is the ICF value of term t and $IBF(t)$ is the IBF value of term t.

3. COSINE SIMILARITY

Cosine similarity is a similarity measurement method between two different texts or documents by measuring the cosine of the angle between the document representation vectors [25]. First, we need to build vector representation of each documents using terms weighting valuein each document. This representation in cartesian field is shown in Figure 1. In Figure 1 there are three documents that been represented by vectors d1, d2 and d3 respectively and one query that been represented by vector q.

Cosine similarity calculates the cosine value of the angel θ between query and each of three documents. This value indicates the degree of similarity of each document and the query. Since it is based on the cosine of the angle between two vectors, the value ranges from 0 to 1.The greater the cosine value, the more the similarity between the query and the document. The cosine value 1 states the100% similarity, while the cosine value 0 means 100% notsimilar. The Cosine similarity of query q and document dj can be counted as follows:

$$\cos(q, d_j) = \frac{\sum_{t_k} [Weight(t_k, q)] \bullet [Weight(t_k, d_j)]}{\sqrt{\sum |Weight(q)|^2} \bullet \sqrt{\sum |Weight(d_j)|^2}}, \quad (6)$$

where $\cos(q,d_j)$ is the cosine value between query q and document d_j, $Weight(t_k,q)$, $Weight(t_k,d_j)$ are weighted words t_k tk on query q and document d_j respectively. Mean while $\sqrt{\sum |Weight(q)|^2}$ and $\sqrt{\sum |Weight(d_j)|^2}$ is the length of the query vector q and document vector d_j respectively. For the weight we can use any term weighting methods such as TF, TF.IDF, TF.IDF.ICF, or TF.IDF.ICF.IBF.

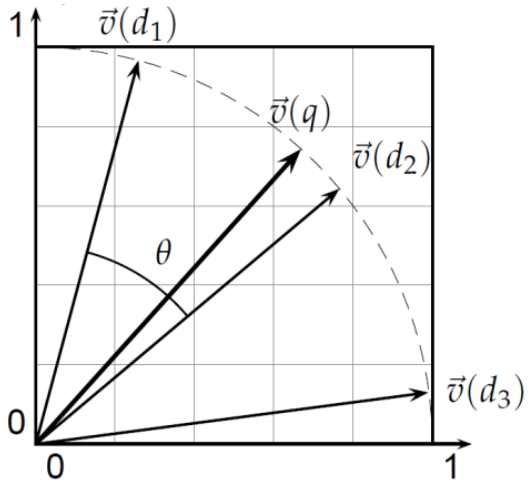

Figure 1. Cosine similarity representation

4. RESEARCH METHOD

Broadly speaking, the information retrieval systemin this study consists of three main stages, preprocessing, features selection and document ranking based on the query from user. In the first stage, preprocessing, there are several steps including tokenization, stopwords removal, stemming and term weighting calculation using TF.IDF.ICF.IBF. The unique terms from this stage would be the original features of each documents. Through the feature selection stage, some of the best featureswere selected from the original feature set.The selection method in this study is based on the TF.IDF.ICF.IBF value of each term. After the best features selected, document ranking stage was conductedby measuring cosine similarity between document vector and query vector based on TF.IDF.ICF.IBF term weighting value. After that, the documents will be sorted descendingly according to their cosine similarity value. This ranking shows the document ranking results according to the level of similarity to the user query.

5. RESULTS AND ANALYSIS

Dataset that have been used in this experiment is an Arabic corpus which is taken from 13 e-books in *Maktabah Syamilah* application. Since every pages of the books was treated as a document, we have 6996 documents distributed in 5 categories. From the whole documents, there are 47.447 distinct terms.

The experiment was conducted using 7 queries. Each of the queries has more than one relevant document. The experiment was also conducted using feature selection that varies from 250 to 1000 best features. The Ground Truth data that been used in this experiment were obtained from an expert. The data contain some queries and the corresponding relevant documents, or technically the pages of particular books, for each of them.

In this experiment, precision, recall and F-Measure of TF.IDF.ICF.IBF method was measured. The experiment result of the proposed method then be compared to some previous term weighting methods including TF.IDF and TF.IDF.ICF. The experiment will also be conducted using another variation of book based indexed term weighting called TF.IDF.IBF. The term weighting methods were not only used during cosine similarity computation, but also used for feature selection.

Table 1. The Performance of the System Using 1000 Features Compared with Previous Methods

	Previous Methods				Proposed Methods			
Term	TF. IDF		TF.IDF. ICF		TF.IDF. IBF		TF.IDF. ICF.IBF	
Weighting	P	R	P	R	P	R	P	R
Q1	1.00	1.00	0.50	0.50	1.00	1.00	1.00	1.00
Q2	0.50	0.25	0.50	0.25	0.50	0.25	0.75	0.75
Q3	0.75	0.75	0.75	0.75	0.75	0.75	0.75	0.75
Q4	0.10	0.33	0.17	0.33	0.17	0.33	0.29	0.67
Q5	1.00	0.50	1.00	0.50	1.00	0.50	1.00	0.50
Q6	0.33	0.50	0.33	0.50	0.33	0.50	0.50	0.50
Q7	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Mean	67%	62%	61%	55%	68%	62%	**76%**	**74%**
F1-Measure	64%		58%		65%		**75%**	

Table 2. The Performance of the System Using 500 Features Compared with Previous Methods

	Previous Methods				Proposed Methods			
Term	TF. IDF		TF.IDF. ICF		TF.IDF. IBF		TF.IDF. ICF.IBF	
Weighting	P	R	P	R	P	R	P	R
Mean	56%	58%	59%	58%	60%	58%	66%	65%
F1-Measure	57%		58%		59%		**66%**	

Table 3. The Performance of the System Using 250 Features Compared with Previous Methods

	Previous Methods				Proposed Methods			
Term	TF. IDF		TF.IDF. ICF		TF.IDF. IBF		TF.IDF. ICF.IBF	
Weighting	P	R	P	R	P	R	P	R
Mean	56%	58%	59%	58%	60%	58%	66%	65%
F1-Measure	57%		58%		59%		**66%**	

The comparison results of the methods using 1000 best features, 500 features and 250 features can be seen at Table 1, Table 2 and Table 3 respectively. The results showed that the proposed method, TF.IDF.ICF.IBF term weighting method have the highest precision, recall, and F-Measure than the other three methods at variations of feature selection. The best performance of this method was obtained when using best 1000 features by precision value of 76%, recall value of 74%, and F-Measure value of 75%. This method is able to search for relevant documents by considering not only the documents index, but also the books and classes index. Therefore, this method can obtain the relevant documents from the appropriate book and category based on the the characteristics of the query entered so that the result became more accurate.

Meanwhile, the TF.IDF.IBF method took second place with the best performance was obtained when using the 1000's best features by precision value of 68%, recall value of 62%, and F-Measure value of 65%. The common term weighting method, TF.IDF, encounter a significant loss in performance when using fewer features. This results showed that the TF.IDF method has lost a lot of important features when only a small number of features used. The results also depicted that the TF.IDF.IBF (without ICF) method higher precision and recall value compared with TFIDF and TF.IDF.ICF. This shows that the addition of IBF has a better impact than ICF. The best results of this method was obtained when using 1000's best feature by precision value of 68%, recall value of 62%, and F-Measure value of 65%.

In addition, from Table 1, 2, and 3 can also be seen that the features reduction also affect the performance of each methods. The fewer features used, the lower performance obtained. TF.IDF have a very significant decrease in performance as the number of features reduced. This is because a lot of important features were lost during the reduction. Some important features had lost because they have small TF.IDF value than some of the other features that should be eliminated. Unlike TF.IDF, TF.IDF.ICF.IBF still has a pretty good performance even though use a little number of features because this method can keep the features that have important roles.

6. CONCLUSION

TF.IDF.ICF.IBF term weighting method can be applied to the retrieval of Arabic documents that have a hierarchy of books with many pages. The experiment results showed that this method has the highest precision, recall and F-Measure value compared with other term weighting methods including TF.IDF, TF.IDF.ICF, and TF.IDF.IBF. The average value of F-Measure using this method is 75%, while the average

value of precision is 76% and the average value of recall reaches 74%. Using feature selection, TF.IDF.ICF.IBF method still has a pretty good performanceeven though a little number of featuresused because this method can keep the features that have important roles. This method obtained the best value when using 1000's best feature by precision value of 76%, recall value of 74%, and the F-Measure value of 75%. As this term weighting method had successfully used in feature selection and document ranking system for documents that have a hierarchy of books with many pages, in future studies, this method can be applied to the classification of documents with the same hierarchy.

REFERENCES

[1] Brunner, B. "The Time Almanac 2000 (Boston, MA: Information Please LLC, 1999)". *In Chief, Time Almanac*, (2005).

[2] Lewis, M. Paul, Gary F. Simons, and Charles D. Fennig. *Ethnologue: Languages of the world*. Vol. 16. Dallas, TX: SIL international, 2009.

[3] Lwin PH. Query Dependent Ranking for Information Retrieval Based on Query Clustering. *International Journal of Informatics and Communication Technology (IJ-ICT).* 2012 Nov 17; 2(1):25-30.

[4] Elraouf, Esraa Abd, Nagwa Lotfy Badr, and Mohamed Fahmy Tolba. "An Efficient Ranking Module for an Arabic Search Engine." *IJCSNS* 10.2 (2010): 218.

[5] Manning, Christopher D., Prabhakar Raghavan, and Hinrich Schütze. *Introduction to information retrieval*. Vol. 1. No. 1. Cambridge: Cambridge university press, 2008.

[6] Enikuomehin T, Sadiku JS. Text Wrapping Approach to natural Language Information retrieval using significant Indicator. IAES International Journal of Artificial Intelligence. 2013 Sep 1;2(3):136.

[7] El Emary, I., and Jaafa Atwan. "Designing and Building an Automatic Information Retrieval System for Handling the Arabic Data." *American Journal of Applied Sciences* 2.11 (2005): 1520-1525.

[8] Harrag, Fouzi, Aboubekeur Hamdi-Cherif, and Eyas El-Qawasmeh. "Vector space model for Arabic information retrieval—application to "Hadith" indexing." *Applications of Digital Information and Web Technologies, 2008. ICADIWT 2008. First International Conference on the*. IEEE, 2008.

[9] Mustafa, Suleiman H. "Character contiguity in N-gram-based word matching: the case for Arabic text searching." *Information processing & management* 41.4 (2005): 819-827.

[10] Mayfield, James, et al. "JHU/APL at TREC 2001: Experiments in Filtering and in Arabic, Video, and Web Retrieval." *TREC*. 2001.

[11] Mustafa, Suleiman Hussein. "Arabic string searching in the context of character code standards and orthographic variations." *Computer standards & interfaces* 20.1 (1998): 31-51.

[12] Larkey, Leah S., Lisa Ballesteros, and Margaret E. Connell. "Light stemming for Arabic information retrieval." *Arabic computational morphology*. Springer Netherlands, 2007. 221-243.

[13] Chen, Aitao, and Fredric C. Gey. "Building an Arabic Stemmer for Information Retrieval." *TREC*. Vol. 2002. 2002.

[14] Taghva, Kazem, Rania Elkhoury, and Jeffrey Coombs. "Arabic stemming without a root dictionary." *Information Technology: Coding and Computing, 2005. ITCC 2005. International Conference on*. Vol. 1. IEEE, 2005.

[15] Larkey, Leah S., Lisa Ballesteros, and Margaret E. Connell. "Improving stemming for Arabic information retrieval: light stemming and co-occurrence analysis." *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2002.

[16] Abu-Salem, Hani, Mahmoud Al-Omari, and Martha W. Evens. "Stemming methodologies over individual query words for an Arabic information retrieval system." *Journal of the Association for Information Science and Technology* 50.6 (1999): 524.

[17] Kadri, Youssef, and Jian-Yun Nie. "Effective stemming for Arabic information retrieval." *proceedings of the Challenge of Arabic for NLP/MT Conference, Londres, Royaume-Uni*. 2006.

[18] Salton G, Buckley C. Term-weighting approaches in automatic text retrieval. Information processing & management. 1988 Jan 1;24(5):513-23.

[19] Al-Taani, Ahmad T., Ahmed S. Ghorab, and Hazem M. Al-Najjar. "An Arabic-English Indexing System using Inverted Index Algorithm."

[20] Harrag, Fouzi, et al. "Experiments in improvement of Arabic information retrieval." *3rd International Conference on Arabic Language Processing (CITALA), Rabat, Morocco*. 2009.

[21] Erritali M. Information Retrieval: Textual Indexing Using an Oriented Object Database. Indonesian Journal of Electrical Engineering and Computer Science. 2016 Apr 1;2(1):205-14.

[22] Ren, Fuji, and Mohammad Golam Sohrab. "Class-indexing-based term weighting for automatic text classification." *Information Sciences* 236 (2013): 109-125.

[23] Salton, Gerard, Anita Wong, and Chung-Shu Yang. "A vector space model for automatic indexing." *Communications of the ACM* 18.11 (1975): 613-620.

[24] Chisholm, Erica, and Tamara G. Kolda. "New term weighting formulas for the vector space method in information retrieval." *Computer Science and Mathematics Division, Oak Ridge National Laboratory* (1999).

[25] Pramukantoro, Eko Sakti, and M. Ali Fauzi. "Comparative analysis of string similarity and corpus-based similarity for automatic essay scoring system on e-learning gamification." In *Advanced Computer Science and Information Systems (ICACSIS), 2016 International Conference on*, pp. 149-155. IEEE, 2016.