퍼지 및 지능시스템학회 논문지 2006, Vol. 16, No. 3, pp. 326-331

A Study on Fuzzy Ranking Model based on User Preference

Dae-Won Kim[1]

School of Computer Science and Engineering, Chung-Ang University, Seoul 156-756, Korea

Abstract

A great deal of research has been made to model the vagueness and uncertainty in information retrieval. One such research is fuzzy ranking models, which have been showing their superior performance in handling the uncertainty involved in the retrieval process. In this study we develop a new fuzzy ranking model based on the user preference. Through the experiments on the TREC-2 collection of Wall Street Journal documents, we show that the proposed method outperforms the conventional fuzzy ranking models.

Key words : Fuzzy similarity measure, relevance ranking, information retrieval

1. Introduction

In recent years a great deal of research in information retrieval(IR) has aimed at modelling the vagueness and uncertainty which invariably characterize the management of information. A set of approaches belonging to this class goes under the name of fuzzy IR. The main levels of application of fuzzy set theory to IR have concerned the representation of documents and the query [1, 2, 3], the associative mechanism such as fuzzy thesauri [4], and the fuzzy ranking models [5, 6, 7, 8].

Many fuzzy ranking models have been showing their superior performance in handling the uncertainty involved in the retrieval process [5, 6, 7, 8]. The ranking is achieved by calculating a similarity between two fuzzy sets, a document D and a query Q. The best-known ranking models are the MMM, PAICE, and P-NORM. However, in spite that the user has an ability to re¤ect their preference for the information need in searching, these conventional fuzzy ranking models are limited to incorporate the user preference when calculating the rank of documents. Taking the problems of existing methods into account, in this study we develop a new fuzzy ranking model based on the user preference.

2. Previous Work

The boolean relevance model is a simple retrieval model based on set theory and boolean algebra. Given its inherent simplicity and neat formalism, boolean models have received great attention in past years and have been adopted by many of the early commercial systems. But the main disadvantage is that there's no notion of a partial match to the query conditions. Thus the exact matching may lead to retrieval of too few or too many documents.

To overcome this disadvantage, a fuzzy model, also called the extended boolean model, was proposed. The fuzzy model could handle the disadvantages of the classical boolean model by introducing the notion of document weight. The document weight is a measure of the degree to which the document is characterized by each index term. The document weights for all index terms lie in the range [0, 1]. However, previous fuzzy model didn't consider the concept of user preference. In this section, we consider three fuzzy models [5], i.e., MMM, PAICE, and P-NORM models.

In the Mixed Min and Max (MMM) model, each index term has a fuzzy set associated with it. The document weight of a document with respect to an index term A is considered to be the degree of membership of the document in the fuzzy set associated with A. Thus given a document D with index-term weight $(d_{A1}, d_{A2}, ..., d_{An})$ for terms $A_1, A_2, ..., A_n$, and the queries

$$Q_{and} = (A_1 \; and \; A_2 \; and \; ... \; and \; A_n) \quad (1)$$

$$Q_{or} = (A_1 \; or \; A_2 \; or \; ... \; or \; A_n) \quad (2)$$

The query-document similarity in the MMM model is computed in the following manner.

$$SIM(Q_{and}, D) = C_{and1} \times \gamma(A) + C_{and2} \times \delta(A)(3)$$

$$SIM(Q_{or}, D) = C_{or1} \times \delta(A) + C_{or2} \times \gamma(A) \quad (4)$$

접수일자 : 2006년 4월 13일
완료일자 : 2006년 5월 30일

$$\gamma(A) = min(d_{A1}, d_{A2}, ..., d_{An}) \quad (5)$$
$$\delta(A) = max(d_{A1}, d_{A2}, ..., d_{An}) \quad (6)$$

where C_{or1}, C_{or2} are softness coeffcients for the *or* operator, and C_{and1}, C_{and2} are softness coeffcients for the *and* operator.

This model is similar to the MMM model in that it assumes that there is a fuzzy set associated with each index term and document weight of a document. However, while the MMM model considers only the maximum and minimum document weights for the index terms while calculating the similarity, the PAICE model takes into account all of the document weights.

$$SIM(Q, D) = \sum_{i=1}^{n} r^{i-1} d_i / \sum_{i=1}^{n} r^{i-1} \quad (7)$$

where Q, r^i mean the query and the constant coeffcient, respectively. d_i means index term weights that is considered in ascending order for *and* operation and descending order for *or* operation.

The previous two fuzzy relevance models, MMM and PAICE models, do not provide a way of evaluating query weights. They only consider the document weights. The P-NORM model explicitly reects the query weight in its model. Given a document D with index-term weights $(d_{A1}, d_{A2}, ..., d_{An})$ for terms $A_1, A_2, ..., A_n$, and the queries Q with weights $(q_{A1}, q_{A2}, ..., q_{An})$, the query-document relevance is calculated as

$$SIM(Q_{and}, D) = 1 - \left(\frac{\sum_{i=1}^{n} (1 - d_{Ai})^p (q_{Ai})^p}{\sum_{i=1}^{n} (q_{A_i})^p} \right)^{1/p} \quad (8)$$

$$SIM(Q_{or}, D) = \left(\frac{\sum_{i=1}^{n} (d_{Ai})^p (q_{Ai})^p}{\sum_{i=1}^{n} (q_{A_i})^p} \right)^{1/p} \quad (9)$$

where p is a control coeffcient ranged from 1 to ∞. In general, the P-NORM model has shown its superior effectiveness to other fuzzy relevance models.

3. Fuzzy ranking model using user preference

3.1 Motivation

As we show below, the conventional ranking models have the following shortcoming in their approach to uncertainty. Let us suppose that we are given a vector of query Q with a fuzzy set of the term and its membership degree:

Q = {fuzzy(0.8),IR(0.7),korea(0.3),author(0.2)}

A document collection consists of four documents (D_1, D_2, D_3, D_4) in which each document is represented as a fuzzy set of the index term and its weight.

D_1 = {fuzzy(0.8),IR(0.7)}
D_2 = {fuzzy(0.2),IR(0.2),korea(0.3),author(0.2)}
D_3 = {korea(0.7),IR(0.8)}
D_4 = {fuzzy(0.8),IR(0.7),korea(0.3),author(0.2)}

Given a query Q and the document collection, we are wondering what is the best result of ranking? Intuitively, we know that D_4 is the most relevant document and D_3 is the least relevant. However, it is arguable to say which one of the two documents D_1 and D_2 has a higher rank. One can regard that the rank of D_1 is higher than that of D_2 because D_1 contains the index terms ('fuzzy' and 'system') showing high-matching similarities. We can also consider that D_2 is more relevant than D_1 because the number of matched terms in D_2 is larger than those in D_1. Such discrepancies arise because conventional ranking models are limited to resolve the uncertainty in a retrieval system.

To solve the addressed problems, we propose a novel ranking model based on the user preference. The key notion of the proposed ranking model is to develop a similarity measure between fuzzy sets in which users can assign their own preference to the decision of ranking.

3.2 Preference-based relevance ranking

Having established the index terms from given documents, a ranking model to calculate the similarity between a document and a query is required. We introduce a notion of user preference, which can provide a more clear ranking result. In this study, each document is represented and regarded as a fuzzy set:

$$D = \{(t_i, \mu_D(t_i))\} \quad (10)$$

where t_i is a term in the index set I and $\mu_D(t_i)$ represents a measure of degree to which the document D is characterized by each index term t_i. Eq. 10 can be expressed in L. Zadeh's convenient notation as $D = \sum_{i=1}^{n} \mu_D(t_i)/t_i = \mu_D(t_1)/t_1 + \mu_D(t_2)/t_2 + \ldots + \mu_D(t_n)/t_n$, where n is a number of terms.

A variety of ranking measures between fuzzy sets have been proposed [7, 8]. However, most of these measures have no mechanism to reect the user preference. Thus we propose a novel similarity measure incorporating the user preference or intention. Firstly, the similarity measure computes the degree of overlap between a document and a query. For each document (D) and query (Q) represented in fuzzy set, we obtain the overlap value between two fuzzy sets at each membership degree (μ) before computing the total overlap. The overlap function $f(\mu)$ at a membership degree μ between D and Q is defned as:

$$f(\mu : D, Q) = \sum_{i=1}^{n} \delta(t_i, \mu : D, Q) \quad (11)$$

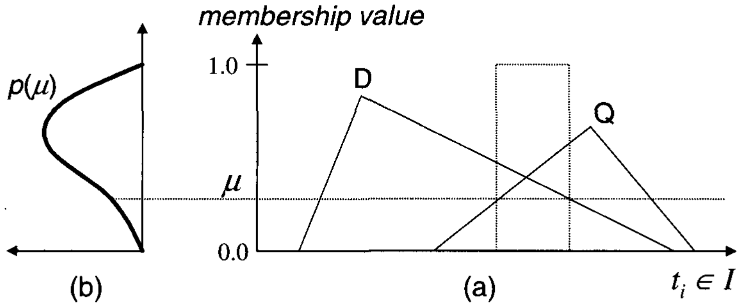

Figure 1: Preference-based similarity computation: (a) overlap degree at μ_p between a document D and a query Q; (b) a membership preference function

where

$$\delta(t_i, \mu : D, Q) = \begin{cases} 1.0 & \text{if } \mu_D(t_i), \mu_Q(t_i) \geq \mu \\ 0.0 & \text{otherwise} \end{cases} \tag{12}$$

$\delta(t_i, \mu : D, Q)$ determines whether two sets are overlapped at the membership degree μ for index term t_i. It returns an overlap value of 1.0 when the membership degrees of the two sets are both greater than μ; otherwise, it returns 0.0. Figure 1(a) depicts an overlap value $f(\mu)$ at membership degree μ between two fuzzy sets. The index terms $t_i \in I$, satisfying both $\mu_D(t_i) \geq \mu$ and $\mu_Q(t_i) \geq \mu$, are given a value 1.0 by Eq. 12. Based on this calculation, we derive the following definition of the similarity measure between a document and a query.

Definition 3.1. Let D and Q be two fuzzy sets representing a document and a query, respectively. Let $f(\mu : D, Q)$ be an overlap function at a given membership degree μ between D and Q, and $p(\mu)$ be a membership preference function. Then, a similarity $S(D, Q)$ between D and Q is defined as

$$S(D, Q) = \sum_{\mu} f(\mu : D, Q) p(\mu) \tag{13}$$

$S(D, Q)$ is obtained by summing $f(\mu : D, Q)$ over the whole range of membership degrees. A larger value of $S(D, Q)$ means that two sets D and Q are more similar to each other, indicating that D is more relevant to Q.

Here, $p(\mu)$ is a preference function of membership, which is determined by the user. When two ranking results that have different fuzzy sets yield the same degree of similarity, the preference function is able to discern the two ranking results by focusing on the higher range of membership degrees. When users search the Web for information, they tend to focus on the document with the terms of highest matching. Thus the relevance of the highest-matched document plays an important role in user satisfaction. In such cases, $p(\mu)$ is given a value in the range [0.7,1.0] when $\mu_D(t_i)$ is considered significant, i.e., $\mu_D(t_i) \geq 0.7$. Under this case, index terms with higher weights place greater emphasis on the calculation of the similarity between D and Q. Conversely, $p(\mu)$ is given a value in the range [0.0,0.3] when $\mu_D(t_i)$ is considered insignificant, i.e., $\mu_D(t_i) \leq 0.3$. Given the preference function $p(\mu)$ in Fig. 1(b), the similarity $S(D, Q)$ is obtained by summing the product of $f(\mu : D, Q)$ and $p(\mu)$ for all membership degrees.

Example 3.2. Consider the ranking example in Section 3.1. For simplicity, let us suppose that $f(\mu, D, Q)$ is calculated at six μ values ($\mu = 0.0, 0.2, 0.4, 0.6, 0.8, 1.0$). Given that $p(\mu)$ is assigned a value of 1.0 if $\mu_D(t_i) \geq 0.6$ and 0.5 otherwise, the similarity $S(D_1, Q)$ is found to 6.0 by the following calculation:

$$\begin{aligned} S(D_1, Q) &= \sum_{\mu} f(\mu : D_1, Q) p(\mu) \\ &= f(0.0)p(0.0) + f(0.2)p(0.2) + \\ &\quad f(0.4)p(0.4) + f(0.6)p(0.6) \\ &\quad + f(0.8)p(0.8) + f(1.0)p(1.0) \\ &= 2 \times 0.5 + 2 \times 0.5 + 2 \times 0.5 + \\ &\quad 2 \times 1.0 + 1 \times 1.0 + 0 \times 1.0 \\ &= 6.0 \end{aligned}$$

Similarly, we find that $S(D_2, Q) = 4.0$, $S(D_3, Q) = 2.0$, and $S(D_4, Q) = 8.0$. Thus the ranking sequence is $D_4 \rightarrow D_1 \rightarrow D_2 \rightarrow D_3$. It is clear that D_4 is the most relevant to Q, and D_3 is the least relevant. Notably, by assigning greater preference on the terms of higher membership degrees, D_1 has a higher rank than D_2 even though the number of matched terms of D_1 is smaller than those of D_2. We see from this example that the proposed similarity measure can clarify the uncertainty in retrieval process by explicitly reflecting the user preference.

Table 1: Average precision of the proposed model for 40 queries varying μ_p, 0.1, 0.3, 0.5, 0.7 and 0.9

Top	0.1	0.3	0.5	0.7	0.9
Top 1	22.50%	25.00%	**35.00%**	27.50%	27.50%
Top 2	25.00%	21.25%	27.50%	**30.00%**	**30.00%**
Top 3	21.67%	17.50%	25.83%	**27.50%**	**27.50%**
Top 4	20.63%	18.75%	25.00%	**25.63%**	**25.63%**
Top 5	19.50%	18.50%	25.00%	**25.50%**	**25.50%**
Top 6	17.92%	17.50%	23.33%	**23.75%**	**23.75%**
Top 7	16.79%	17.14%	**22.50%**	**22.50%**	**22.50%**
Top 8	15.63%	15.94%	**22.19%**	21.25%	21.25%
Top 9	15.28%	15.28%	**21.39%**	20.56%	20.83%
Top 10	14.50%	14.75%	**20.75%**	19.75%	19.75%
Top 11	13.64%	13.86%	**20.45%**	18.41%	18.41%
Top 12	12.50%	13.54%	**20.63%**	19.17%	19.17%
Top 13	12.12%	12.88%	**19.42%**	18.85%	18.85%
Top 14	11.79%	12.32%	**18.75%**	17.86%	17.86%
Top 15	11.67%	12.33%	**18.67%**	17.50%	17.50%
Top 16	11.25%	11.72%	**18.44%**	16.88%	16.88%
Top 17	11.32%	12.06%	**18.38%**	15.88%	15.88%
Top 18	11.39%	12.08%	**17.78%**	15.56%	15.56%
Top 19	11.13%	11.50%	**17.24%**	15.00%	15.00%
Top 20	15.38%	15.28%	**16.88%**	14.50%	14.50%
Avg.	15.38%	15.28%	**21.76%**	20.68%	20.69%

4. Experimental results

To demonstrate the effectiveness of the proposed method, we conducted retrieval experiments in which the proposed ranking method was compared with the PAICE, P-NORM, and vector model in the normalized TF×IDF index. The retrieval results were assessed using the precision and recall measures.

The data set employed was the TREC-2 collection of 1990 Wall Street Journal (WSJ) documents, which comprises 21,705 documents; the built-in 40 queries were used for relevance judgement. The parameters used in the ranking methods were set as follows:the preference function $p(\mu)$ is given a value of 1.0 if $\mu_D(t_i) \geq \mu_p$ and 0.1 otherwise; μ_p is a preference threshold for determining the degree of signi£cance, in this work varying μ_p as 0.1, 0.3, 0.5, 0.7 and 0.9.

As shown in Example 3.2, the ranks of relevant documents are changed by the preference assignment on the terms according to the membership degree μ. Therefore, in this section, we analyze the dependence of the search performance of the proposed method on the choice of $p(\mu)$, speci£cally the preference threshold μ_p.

Table 1 lists the search results of the proposed ranking model, average precision ranging from Top 1 to Top 20 documents for 40 queries varying the preference threshold μ_p. The best result obtained in each Top N is marked in bold face. From the table, we see that the best precision is obtained for μ_p = 0.5 although the precision result obtained at μ_p = 0.7 is better than that obtained at μ_p = 0.5 under Top 10 documents. Overall, the precision values obtained at $\mu_p \geq 0.5$ are better than those obtained at $\mu_p <$ 0.5.

As a second experiment, the search result obtained by the proposed method using the preference function with μ_p = 0.5 was compared with the search result obtained using the PAICE, P-NORM and vector model.

The search performance of the PAICE model was much similar with that of the P-NORM. The average precision of the PAICE and P-NORM models for 40 queries ranging from Top 1 to Top 20 are 1.48% and 1.62%, and the average recall of the PAICE and P-NORM are 0.47% and 0.59%, respectively.

Figure 2 shows the search results of each ranking model, P-NORM, vector and the proposed. The P-NORM and vector models give average precisions of 1.62% and 19.48% respectively. In contrast, the proposed model gives the higher average precision of 21.76%. Moreover, we see that the average precision of the proposed ranking model for the Top-ranked document (35.00%) is remarkably higher than those of the other two models. The average recall of the P-NORM and vector models are 1.62% and 14.53%, respectively. In this case, the proposed model gives an average recall of 16.82%. From this experiment,

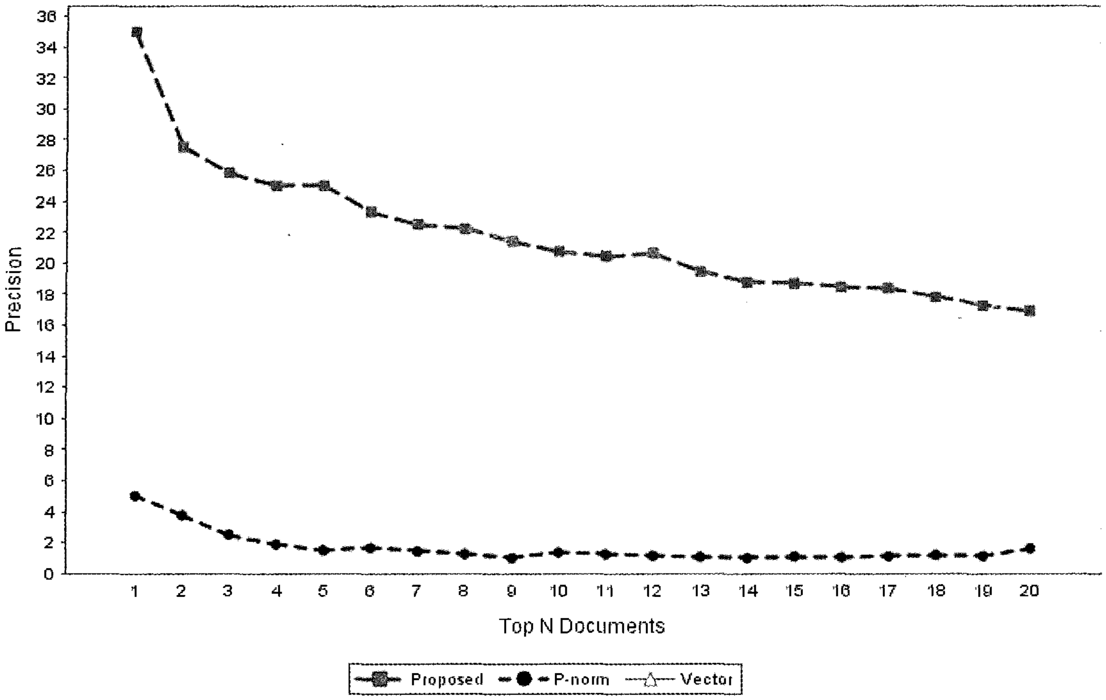

Figure 2: Average precision of each model for 40 queries

we see that the proposed ranking model can clarify the uncertainty in retrieval process by explicitly re¤ecting the user preference and therefore, remarkably outperforms the PAICE, P-Norm and vector models.

5. Conclusion

In this paper, the limitations of conventional ranking methods are examined. Based on these considerations, a new fuzzy retrieval system incorporating the notion of user preference has been proposed for document retrieval. The proposed preference-based ranking provides more clear similarity calculation between a document and a query by allowing users to assign their preference or intention to the weights of terms.

Acknowledgement. This Research was supported by the Chung-Ang University Research Grants in 2006.

References

[1] M.J. Martín-Baustista, D.H.Kraft, M.A.Vila, J.Chen, J.Cruz, User pro£ls and fuzzy logic for web retrieval issues, Soft Computing, Vol. 6, 2002, 365–372.

[2] R.R.Yager and F.E.Petry, A framework for linguistic relevance feedback in content-based image retrieval using fuzzy logic, Information Sciences, In Press, April 2005.

[3] A.F. Smeaton, Relevance feedback and a fuzzy set of search terms in an information retrieval system, Information Technology Research Development Applications archive, Vol. 3, Issue 1, 1984.

[4] L.J.Kohout, Keravanou, E., and Bandler, W. Information retrieval system using fuzzy relational products for thesaurus construction. Proceedings IFAC Fuzzy Information, Marseille, France, 7-13, 1983.

[5] J.H. Lee, On the evaluation of Boolean operators in the extended boolean retrieval framework, Proceedings of the 17th SIGIR conference, 1994, 182–190.

[6] R. Baeza-Yates, et al., Modern information retrieval, Addison-Wesley, 1999.

[7] W.J. Wang, New similarity measures on fuzzy sets and on elements, Fuzzy Sets and Systems, Vol.85, 1997, 305–309.

[8] J. Fan, W. Xie, Some notes on similarity measure and proximity measure, Fuzzy Sets and Systems, Vol.101, 1999, 403–412.

[9] N. Stokes and J. Carthy, Combining semantic and syntactic document classi£ers to improve £rst story detection, In proceedings of the 24th ACM SIGIR conference, New Orleans, 2001, pp. 424.

[10] W. Gale, K. Church, and D. Yarowsky, Estimating upper and lower bounds on the performance of word-sense disambiguation programs, ACL, 1992.

[11] S. Deerwester, S.T. Dumais, G.W. Furnas, T.K. Landauer, and R. Harshman, Indexing by latent semantic analysis. Journal of the American Society of Information Science, vol. 41(6), 1990. pp.391-407.

[12] T.G. Kolda and D.P. O'Leary, A semidiscrete matrix decomposition for latent semantic indexing in information retrieval. In Proceedings of ACM Transaction of Information Systems, Vol. 16, 1998, pp. 322-346.

[13] T. A. Letsche and M. W. Berry, Large-scale information retrieval with latent semantic indexing, Information Sciences, Vol. 100, Issues 1-4, 1997, pp. 105-137

[14] C. Fellbaum et al., WordNet:An eletroic lexical database, The MIT press, 1998, pp.338-339.

저자소개

Dae-Won Kim

한국 퍼지 및 지능시스템학회 이사
현재 중앙대학교 컴퓨터공학부 교수
제 13권 6호(2003년 12월호) 참조
E-mail : dwkim@cau.ac.kr