A Study on Semantic Based Indexing and Fuzzy Relevance Model

Bo-Yeong Kang[0] Dae-Won Kim(KAIST) Sang-Ok Gu Sang-Jo Lee
Dept. of Computer Engineering in Kyungpook National University
comeng99@hotmail.com dwkim@if.kaist.ac.kr

의미 기반 인덱스 추출과 퍼지 검색 모델에 관한 연구

강보영[0] 김대원(KAIST) 구상옥 이상조
경북대 컴퓨터 공학과

ABSTRACT

If there is an Information Retrieval system which comprehends the semantic content of documents and knows the preference of users, the system can search the information better on the Internet, or improve the IR performance. Therefore we propose the IR model which combines semantic based indexing and fuzzy relevance model. In addition to the statistical approach, we chose the semantic approach in indexing, *lexical chains*, because we assume it would improve the performance of the index term extraction. Furthermore, we combined the semantic based indexing with the fuzzy model, which finds out the exact relevance of the user preference and index terms. The proposed system works as follows: First, the presented system indexes documents by the efficient index term extraction method using lexical chains. And then, if a user tends to retrieve the information from the indexed document collection, the extended IR model calculates and ranks the relevance of user query, user preference and index terms by some metrics. When we experimented each module, semantic based indexing and extended fuzzy model, it gave noticeable results. The combination of these modules is expected to improve the information retrieval performance.

1. Introduction

If there is an Information Retrieval (IR) system which comprehends the semantic content of documents and knows the preference of users, it can be much help to search the information on the Internet, or improve the performance of the existing systems. Therefore, in this paper, we focus on the system design in the aspects of indexing and retrieval model.

We choose the semantic approach in indexing because we assume it would improve the index term extraction performance. And there is a further research that knowledge about discourse structures and their signaling linguistic phenomena can help in selecting keywords from a text that are reflective of its content[1][2][3][4][5]. The proposed system indexes the documents by the semantic approach using lexical chains[3]. And then we combine the semantic based indexing with fuzzy relevance model. The fuzzy system ranks documents according to the exact relevance of the user preference and a user query by some metrics[6]. Moens said that the disadvantage of extraction indexing method is that the words in a document is potentially ambiguous[7]. So selected terms also may be ambiguous. The proposed method, *semantic window*, performs sense disambiguation roughly in the process of finding keywords, so we can select sense disambiguated keywords from a document. This enables us to deal with lexical ambiguities better. Moreover, the existing indexing has used only statistic methods. But it is insufficient for understanding documents and extracting representative words. Suggested approach helps us to reduce the disadvantages of existing extraction indexing method.

Most IR system must rank documents according to the degree of relevance to the user query. Thus, the notion of relevance is at the center of information retrieval. So far a variety of similarity models including boolean model, vector model and fuzzy model have been proposed to reflect the relevance degree[8][9]. The main disadvantage of Boolean model is that there's no notion of a partial match to the query conditions. So the exact matching may lead to retrieval of too few or too many documents. The vector model recognizes that the use of boolean models is too limiting, and proposed a framework in which partial matching is possible. However, in practice, consideration of term frequencies is not suitable to the indiscriminate applications and might hurt the overall performance. The fuzzy models, MMM and PAICE, could handle the disadvantages of 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. However, the previous fuzzy models didn't consider the concept of user preference. In a real situation, the user wants to reflect his or her preference in the search process.

To reflect the user preference, we proposed a new similarity metric which consists of domain preference and membership value preference.

This paper is organized as follows: in section 2, we will examine the overall system design which combines the semantic approach of indexing and the extended fuzzy system. We test and analyze each module of the suggested system in section 3. And then we will conclude this paper in section 4.

2. Semantic Based Indexing and Fuzzy Relevance Model

The configuration of the proposed system is like following figure 1. First, we index documents by the semantic approach to extract indexing words which represent the documents better. The presented system gets user preference directly from users to get user intention, and retrieves information by the extended fuzzy model which computes the relation of the user's preference and index terms.

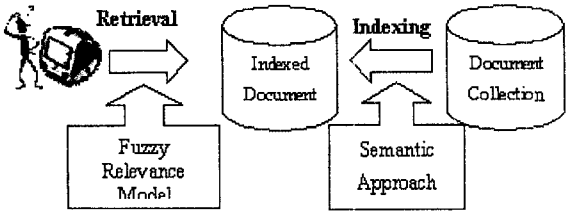

Figure 1. Configuration of the proposed system

2.1 Semantic Approach of Indexing

Barzilay and Elhadad used lexical chains for extracting representative words for text summarization[10]. We extended their approach by improvement of sense disambiguation. Semantic indexing module configuration is like the figure 2.

The noun scoring representation in a chain is as follows:

$$\delta(N_i) = \sum_k c(r^k_{N_i}) \times \rho(r^k_{N_i})$$

k = {identity, synonym, hypernym, hyponym, antonym}

$\delta(N_i)$ stands for the score of a noun, r_{Ni} stands for k relation among the relation kinds which noun i has, $c(r_{Ni})$ stands for the count of k relation, and $\rho(r_{Ni})$ represents the score of k relation. The chain which satisfies the following condition is called *strong chain*[10]: we select keywords from a few strong chains.

$$\text{Score}(\text{Chain}_x) = \text{Sum}(\text{Score}(\text{Noun}_i)) + \log_2\text{Num}(\text{Noun}_i),\ i \in x$$
$$\text{Score}(\text{Chain}) > \text{Average}(\text{Scores}) + C^* \text{StandardDeviation}(\text{Scores})$$

We should add the statistic metric(i.e., tf/idf etc) to an weighting module, but we didn't develop the combination method of semantic metric and statistic metric yet. We will continue to develop the weighting measure.

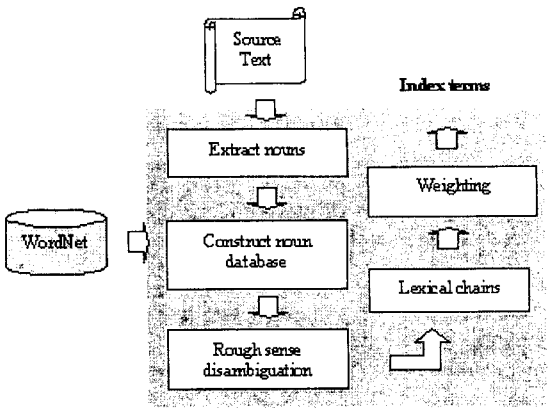

Figure 2. Configuration of indexing

2.2 Fuzzy Relevance Model

The fact that each index term has vagueness and the user preference is a very important factor in IR system. Besides the weight of each index term, the reference on the distribution of index term weight is also essential in designing similarity model. Hence, we developed a new similarity measure between fuzzy sets, which can generate a better relevance degree and reflect the user preference and weight. To accomplish this, we proposed two new concepts, domain preference on the domain axis and membership degree preference on the membership axis. By introducing these concepts, a user can give a weight to the specific part which they think more important. The final similarity value is a composition of domain preference and membership preference using integration function.

This preference is a weight on the domain axis of comparing fuzzy sets. The definition of this value is shown in equation 1. Membership preference function is defined as equation 2.

$$dDomain = f(x)dx \qquad (1)$$

$$dMV = f_{MV}(y)dy \qquad (2)$$

The algorithm requires two steps. First, preliminary similarity value is calculated using domain preference function. And after that, we compute the whole similarity value by applying the membership preference function.

For a point x on the domain, the similarity value $\psi_{A,B}(x, y)$ which corresponds to specific membership value y is calculated by equation 3. According to the above equation, if fuzzy set A and B is similar at specific level y, it means both fuzzy sets are greater than membership value y. By equation 1, domain preference can be applied the integral form given in equation 4.

$$\psi_{A,B}(x, y) = \begin{cases} 1 & \text{if } y \le MIN(\mu_A(x), \mu_B(x)) \\ 0 & \text{otherwise} \end{cases} \qquad (3)$$

$$\xi(y) = \begin{cases} \int_{Domain} \psi_{A,B}(x,y)dDomain \\ \int_{Domain} \psi_{A,B}(x,y) f_{Domain}(x)dx \end{cases} \quad (4)$$

Given a domain preference function $f_{Domain}(x)$ and a domain area r, similarity calculation process is carried out at specific membership value y. For all x∈ r, $\psi_{A,B}(x,y)$ is set to 1 by equation 3. The integration value of domain function in equation 4 becomes the similarity value which reflects the preference at specific level y. The similarity value obtained by domain preference is based on the specific membership level y. So it's necessary to add the membership preference concept in the second step of algorithm. This computation is represented in equation 5. $\delta_{A,B}$ is a final similarity value, which is obtained by integrating $\xi(y)$ on membership area(MV, membership-value). This can be rewritten in equation 6 using membership function $f_{MV}(y)$.

$$\delta_{A,B} = \int_{MV} \xi(y)dMV$$

$$\delta_{A,B} = \int_{y=0}^{1} \xi(y) f_{MV}(y)dy$$

3. Experimental Results

We had an experiment of each module, indexing and fuzzy model, and verified the performance of those modules. But we didn't have an experiment of the performance of the proposed model yet.

There are few reports about the performance of a keyword extraction system itself. Hence, using results of a keyphrase extraction system, KEA[11][12], we evaluated the presented system. KEA must be trained by domain specific data collection. 11 abstracts of 11 papers were randomly selected from ACM on-line paper archive as test collection. KEA gave better results than our system, when it extracted less than three keywords. In case of extracting more than three keywords, our system gave better results than KEA without any learning and any domain specific dictionary. The experiment for fuzzy relevance model was carried out for the Korean language set (KT SET) which consists of 4,414 documents and 50 queries. In the table 1, the result of ranking is compared for various models. The most left column shows the ranking order, and the value within parenthesis means the relevance degree for the given query. The coefficient values of MMM model were set like this. C and1 is in [0.5,0.8] and C or1 > 0.2. The coefficient r of PAICE model was set to 1.0 for *and* query, and to 0.7 to *or* query. As we expected, the most relevant document is D 4 and least relevant one is D 3 in most cases. And the value of similarity (relevance) between D 1 and D 2 is not quite different. But, in our proposed method, the difference is very clear and D 1 is more relevant than D 2 because we applied a high degree of membership value preference.

Rank	Vector	MMM	PAICE	Proposed
1	D 4 (0.94)	D 4 (0.38)	D 4 (0.80)	D 4 (1.00)
2	D 1 (0.87)	D 3 (0.24)	D 1 (0.80)	D 1 (0.84)
3	D 2 (0.71)	D 1 (0.24)	D 2 (0.80)	D 2 (0.21)
4	D 3 (0.60)	D 2 (0.23)	D 3 (0.30)	D 3 (0.10)

Table 1. Experimental result: the rank and relevance degrees

4. Conclusion and Future Works

But, there are some limitations in the proposed IR system. We extract the keywords only from nouns which exist in a document. Therefore the presented system doesn't extract such keywords that can represent a document better but don't exist in a document. The suggested system was developed to work in general domains, because we use WordNet as a dictionary developed for general use, but test documents have many abbreviated words and terminology of Computer Science. These factors interrupted constructing exact lexical chains. In addition, the presented system didn't support compound nouns yet. Generallly keywords are often more informative when they compose a compound noun. As a future work of fuzzy relevance model, we'll analyze the proposed model in various contexts and compare with other methods by using the recall and precision metrics which are often used to measure the performance of IR system. The improvements of these problems will enable us to implement a robust system, by which we are able to guess the content of a document better and retrieve the exact documents.

REFERENCES

[1] Hahn, U. Making unerstanders out of parsers: semantically driven parsing as a key concept for realistic text understanding applications. *International Journal of Intelligent Systems*, 4: 345—393, 1989.

[2] Lewis, D.D. and Sparck Jones, K. Natural language processing for information retrieval. *Communications of the ACM*, 39(1): 92—101, 1996.

[3] Bookstein, A., Klein, S.T. and Raita, T. Clumping properties of content-bearing words. *JASIS*, 49(2): 102-114, 1998

[4] Morris, J. Lexical cohesion, the thesaurus, and the structure of text. Master's thesis. Department of Computer Science. University of Toronto(Tech. Rep. No. CSRI-219), 1988.

[5] Morris, J. and Hirst, G. Lexical cohesion computed by thesaural relations as an indicator of the structure of text. *Computational Linguistics*, 17(1): 21-43, 1991.

[6] Kim, D.W. and Kwang H. Lee. A New Fuzzy Information Retrieval System based on User Preference Model. *Proceedings of IEEE 10th International Conference on Fuzzy Systems*, 2001

[7] Moens, M.F. Automatic Indexing and Abstracting of Document Texts. Kluwer Academic Publishers, 2000.

[8] Baeza-Ytes, R. and Ribeiro-Neto, B. Modern Information Retrieval,.Addison-Wesley, 1999.

[9] Lee. J. H. On the evaluation of Boolean operators in the extended Boolean retrieval framework. *Proc. of the 17th annual international ACM-SIGIR conference on Research and development in information retrieval*, pp. 182--190, 1994.

[10] Barzilay, R. and Elhadad, M., lexical chains for text summarization. *Proceedings of the ACL'97 Workshop on Intelligent Scalable Text Summarization*, 1997.

[11] Frank, E., Paynter, G., Witten, I., Gutwin, C. and Nevill-Manning, C. Domain-specific Keyphrase Extraction. *In Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence*, pp. 668--673, 1999.

[12] Witten, I.H., Paynter, G.W., Frank, E., Gutwin, C. and Nevill-Manning, C.G. KEA: Practical Automatic Keyphrase Extraction. *In Proceedings of Digital Libraries '99: The fourth ACM Conference on Digital Libraries*, pp. 254--255, 1999.