A New Method for Fuzzy Information Retrieval Based on Fuzzy Hierarchical Clustering and Fuzzy Inference Techniques

Yih-Jen Horng, Shyi-Ming Chen, *Senior Member, IEEE*, Yu-Chuan Chang, and Chia-Hoang Lee

***Abstract*—In this paper, we extend the work of Kraft *et al.* to present a new method for fuzzy information retrieval based on fuzzy hierarchical clustering and fuzzy inference techniques. First, we present a fuzzy agglomerative hierarchical clustering algorithm for clustering documents and to get the document cluster centers of document clusters. Then, we present a method to construct fuzzy logic rules based on the document clusters and their document cluster centers. Finally, we apply the constructed fuzzy logic rules to modify the user's query for query expansion and to guide the information retrieval system to retrieve documents relevant to the user's request. The fuzzy logic rules can represent three kinds of fuzzy relationships (i.e., fuzzy positive association relationship, fuzzy specialization relationship and fuzzy generalization relationship) between index terms. The proposed fuzzy information retrieval method is more flexible and more intelligent than the existing methods due to the fact that it can expand users' queries for fuzzy information retrieval in a more effective manner.**

***Index Terms*—Fuzzy agglomerative hierarchical clustering, fuzzy information retrieval systems, fuzzy logic rules, fuzzy relationships, query expansion.**

I. Introduction

THE goal of an information retrieval system is to evaluate the degrees of relevance of the collected documents with respect to a user's queries and retrieve the documents with a high degree of satisfaction to the user. In order to get good retrieval performance, the user's query must be able to appropriately describe the user's requests. Currently, most of the commercial information retrieval systems are based on the Boolean logic model. They assume that a user's queries can precisely be characterized by the index terms. However, this assumption is inappropriate due to the fact that the user's queries may contain fuzziness [22]. The reason for the fuzziness contained in the user's queries is that the user may not know much about the subject he/she is searching or may not be familiar with the information retrieval system. Therefore, the query specified by the user may not describe the information request properly. Since fuzzy set theory [29] can be used to describe imprecise or fuzzy information, many researchers have applied the fuzzy set theory to information retrieval systems [2], [3], [8], [9], [13], [18]–[22].

In [2], Bordogna *et al.* presented a relevance feedback model based on associative neural networks to provide an association mechanism in information retrieval systems. The purpose of the association mechanism in information retrieval systems is to build the association relationships between index terms and to modify the user's queries by adding or replacing index terms associated with the queries. Generally speaking, the modified user's queries should find more relevant documents than that of the original user's queries and thus improve the retrieval performance. Therefore, the study of the association mechanism is very important in the field of information retrieval. In [3], Chen *et al.* presented a fuzzy-based concept information system that integrates human categorization and numerical clustering. In [8], Chen *et al.* presented a method for document retrieval using knowledge-based fuzzy information retrieval techniques. In [9], Chen *et al.* presented fuzzy information retrieval techniques based on multi-relationship fuzzy concept networks. In [13], Horng *et al.* presented a fuzzy information retrieval method based on document terms reweighting techniques,

In [20], Kraft *et al.* explored several ways of using fuzzy clustering techniques in information retrieval systems, where the most important one is to capture the relationships among index terms. They use fuzzy logic rules to represent the association relationships between index terms and to form the basis of the association mechanism. After a user submits his/her queries, the fuzzy logic rules are then applied under a fuzzy logic system to modify the user's original queries. Experimental results show that the modified user's queries can get a better retrieval performance than the original queries. In [20], Kraft *et al.* utilized the complete link clustering method and the fuzzy c-means clustering method to partition documents for information retrieval.

In this paper, we extend the work of Kraft *et al.* [20] to present a new method to modify a user's queries for fuzzy information retrieval. First, we present a fuzzy agglomerative hierarchical clustering algorithm for clustering documents and to get the document cluster center of each document cluster. Then, we present a method to construct fuzzy logic rules based on the document clusters and their document cluster centers. Finally, we apply the constructed fuzzy logic rules to modify the user's query for query expansion and to guide the information retrieval system to retrieve documents relevant to a user's request. The

Manuscript received October 7, 2002; revised November 6, 2003, March 2, 2004, and May 25, 2004. This work was supported in part by the National Science Council, Republic of China, under Grant NSC 91-2213-E-011-052.

Y.-J. Horng and C.-H. Lee are with the Department of Computer and Information Science, National Chiao Tung University, Hsinchu 300, Taiwan, R.O.C.

S.-M. Chen and Y.-C. Chang are with the Department of Computer Science and Information Engineering, National Taiwan University of Science and Technology, Taipei 106, Taiwan, R.O.C.

Digital Object Identifier 10.1109/TFUZZ.2004.840134

fuzzy logic rules can represent three kinds of fuzzy relationships (i.e., fuzzy positive association relationship, fuzzy specialization relationship, and fuzzy generalization relationship) between index terms. The proposed document retrieval method is more flexible and more intelligent than the existing methods due to the fact that it can expand users' queries for fuzzy information retrieval in a more effective manner.

The rest of this paper is organized as follows. In Section II, we briefly review some clustering methods. In Section III, we propose a new fuzzy agglomerative hierarchical clustering method. In Section IV, we compare the clustering performance of the proposed fuzzy agglomerative hierarchical clustering method with that of the complete link clustering method. In Section V, we propose a new method for fuzzy logic rules discovery. In Section VI, we propose a new method for query modification for fuzzy information retrieval based on the constructed fuzzy logic rules. The conclusions are discussed in Section VII.

II. A Review of Clustering Methods

The single link clustering (SLC) method is one of the hierarchical agglomerative clustering methods (HACM). In [24], Rohlf reviewed some algorithms of the single link clustering method. The time complexity of these algorithms ranges from $O(N \log N)$ to $O(N^5)$, where N is the number of items. Many of these algorithms are not suitable for information retrieval applications, when the data sets have large N and high dimensionality [23]. The SLINK algorithm [28] is one of the single link clustering methods. In [28], Sibson pointed out that the SLINK algorithm is an optimally efficient algorithm for the single link clustering method. In [23], Rasmussen pointed out that the SLINK algorithm generates the hierarchy in a form known as the *pointer representation*. The SLINK method is one of the SLC methods [24] and it also takes the major characteristic of SLC, where the degree of similarity between two document clusters is defined by the maximal degree between any pairs of documents, each of which is in one of the two clusters.

In [20], Kraft *et al.* utilized the agglomerative hierarchical clustering (AHC) method [23] and the fuzzy c-means clustering method [1] to partition documents. In the following, we briefly review these two clustering methods.

The agglomerative hierarchical clustering method for partitioning the given documents is reviewed from [20] as follows.

Step 1) Let each document be a document cluster.

Step 2) Merge two document clusters which have the largest degree of similarity into one document cluster.

Step 3) **If** the degree of similarity between any two document clusters is smaller than a heuristic threshold value α, where $\alpha \in [0, 1]$, **then Stop**,
else, go to **Step 2)**.

In [20], Kraft *et al.* pointed out that the measurement of the degree of similarity between two document clusters can be done by the CLC method [23], which takes the minimum degree of similarity between any pairs of documents, each of which is in one of the two clusters (i.e., one document from the first document cluster and the other from the second document cluster).

It is obvious that the agglomerative hierarchical clustering method is a crisp clustering method. That is, in the resulting document clusters, each document can only belong to exactly one document cluster.

The other clustering method used in [20] is the fuzzy c-means algorithm [1]. Assume that there is a set of n documents, $D = \{d_1, d_2, \ldots, d_n\}$, where each document d_i can be regarded as a data point represented by a vector of dimension s shown as follows:

$$d_i = \langle w_{i1}, w_{i2}, \ldots, w_{is} \rangle$$

where $1 \leq i \leq n$. Here, w_{ij} denotes the degree of significance of term t_j in d_i, where $0 \leq w_{ij} \leq 1$ and $1 \leq j \leq s$. Furthermore, assume that we want to partition the documents into c document clusters, where $c \geq 2$. Then, the fuzzy c-means algorithm will get c document cluster centers and the membership degree of each document belonging to each document cluster by minimizing the following objective function:

$$J_m = \sum_{k=1}^{c} \sum_{i=1}^{n} (\mu_{ki})^m \|d_i - v_k\|^2. \tag{1}$$

Here, μ_{ki} denotes the membership degree of document d_i with respect to document cluster A_k, v_k denotes the document cluster center of document cluster A_k, and $\|d_i - v_k\|$ denotes the distance from document d_i to document cluster center v_k. Moreover, d_i and v_k are represented as vectors of dimension s, and m is a control parameter, where $m > 1$. Note that m stands for the degree of fuzziness of the clustering algorithm. The larger the value of m, the higher the degree of fuzziness, i.e., the likelihood of a document belonging to multiple clusters at the same time is higher. It should be noted that the membership degree of each document d_i belonging to each document cluster A_k must be nonnegative, i.e., $\mu_{ki} \geq 0$. Moreover, the summation of the membership degrees of each document d_i belonging to each cluster must be equal to one, i.e., $\sum_{k=1}^{c} \mu_{ki} = 1$, where $1 \leq i \leq n$. The membership degree of document d_i belonging to document cluster A_k is calculated as follows:

$$\mu_{ki} = \frac{\left[\|d_i - v_k\|^2\right]^{-1/(m-1)}}{\sum\limits_{j=1}^{c} \left[\|d_i - v_j\|^2\right]^{-1/(m-1)}} \tag{2}$$

and the document cluster center v_k of document cluster A_k is calculated as follows:

$$v_k = \frac{\sum\limits_{i=1}^{n} (\mu_{ki})^m d_i}{\sum\limits_{i=1}^{n} (\mu_{ki})^m}. \tag{3}$$

Initially, the fuzzy c-means algorithm randomly assigns a value for each μ_{ki}, where $\mu_{ki} \geq 0$ and $\sum_{k=1}^{c} \mu_{ki} = 1$ for each i. Then, the following two steps are performed iteratively until the difference of the values of μ_{ki} (and v_k) in the current iteration and those in the previous iteration are smaller than some convergence threshold δ, where $\delta > 0$. First, it uses the existing μ_{ki} in formula (3) to obtain the document cluster center v_k. Then, it uses the newly derived document cluster center v_k shown in formula (2) to update the value of v_k. In [1], Bezdek *et al.* have proven that the fuzzy c-means algorithm will converge.

In [20], Kraft *et al.* used several experts to evaluate the document clusters produced by the CLC method [23] and the ones produced by the fuzzy c-means clustering method [1]. Since the fuzzy c-means algorithm will produce "fuzzy" document clusters in the sense that each document may belong to multiple document clusters, it is in contrast to the "crisp" document clusters produced by the complete link clustering method in which each document only can belong to exactly one document cluster. In order to compare the performance of these two clustering methods, "hardening" is performed to the fuzzy document clusters obtained by the fuzzy c-means algorithm. That is, for each document d_i, they found a document cluster where d_i has a maximum membership degree among all the document clusters. Then, they set the membership degree of document d_i in this document cluster to one and set the membership degrees for the other document clusters to zero. From the experts' opinions [20], the complete link clustering method seems to perform better than the fuzzy c-means method. However, since Kraft *et al.* performed the "hardening" operation to the fuzzy document clusters, the experts have only compared two sets of "crisp" document clusters derived from the two clustering methods, respectively. Therefore, the advantage of the fuzzy clustering method is not revealed in their experiments. In fact, the fuzzy clustering method should perform better than the crisp clustering method when the boundaries between document clusters are not crisp or when the document clusters are overlapping.

Because the traditional agglomerative hierarchical clustering method is a crisp clustering method, where each document can only belong to exactly one document, it lacks flexibility. Therefore, in this paper, we extend the work of [20] to present a fuzzy agglomerative hierarchical clustering method to overcome the drawback of the traditional agglomerative hierarchical clustering method, where a document can not belong to multiple document clusters at the same time. Both the proposed fuzzy agglomerative hierarchical clustering method and the fuzzy c-means clustering method [1] have the advantage of flexibility to allow a document to belong to multiple document clusters at the same time. The difference between the fuzzy c-means clustering method and the proposed fuzzy agglomerative hierarchical clustering method is that the fuzzy c-means clustering method needs to predefine the number of clusters by the user and the proposed method uses the "similarity threshold value" α and the "difference threshold value" λ to deal with the process of clustering automatically, where $\alpha \in [0,1]$ and $\lambda \in [0,1]$.

III. Fuzzy Agglomerative Hierarchical Clustering Method

In this section, we present a fuzzy agglomerative hierarchical clustering (FAHC) for clustering documents. Let the membership degree of document $\mathrm{d_y}$ belonging to cluster A_i be denoted by $\mathrm{M}_{Ai}(\mathrm{d_y})$, where $\mathrm{M}_{Ai}(\mathrm{d_y}) \in [0,1]$. If d_i is an element of cluster A_j, then $\mathrm{d}_i \in A_j$. The proposed fuzzy agglomerative hierarchical clustering method is now presented as follows.

Fuzzy Agglomerative Hierarchical Clustering Algorithm:

```
Input: The similarity threshold value
$\alpha$ and the difference threshold value $\lambda$,
where $\alpha \in [0,1]$ and $\lambda \in [0,1]$.
Output: Document clusters.
Step 1: Let each document be a document
cluster and set the membership degree of
each document belonging to its document
cluster to 1.
Step 2: Find a pair of document clusters
$A_i$ and $A_j$ among the set of document clus-
ters that has the largest degree of simi-
larity $\psi$ and $\psi$ is not less than $\alpha$, where
$\psi \in [0,1]$.
Step 3: Find a set S of pairs of docu-
ment clusters $A_k$ and $A_l$ among the rest of
the document clusters that have degrees
of similarity larger than or equal to
the similarity threshold value $\alpha$, where
$\alpha \in [0,1]$.
Step 4: For each pair of document cluster
$A_k$ and $A_l$ in the set S do
 begin
  if $\psi - \omega < \lambda$ and the pair of document
clusters $A_i$ and $A_j$ obtained in Step 2 and
the pair of document clusters $A_k$ and $A_l$
obtained in Step 3 share the same document
cluster $A_s$ (i.e., $A_s = A_i = A_k$), then
  begin
   make a copy $A_i^*$ of document cluster $A_i$;
   merge document cluster $A_i$ with document
   cluster $A_j$ into a new cluster $A_{ij}$;
   for each element $d_\mathrm{y}$ in document
   cluster $A_i$, the membership degree
   $\mathrm{M}_{Aij}(\mathrm{d_y})$ of document $d_\mathrm{y}$ belonging
   to the new document cluster $A_{ij}$
   is equal to $\mathrm{M}_{Ai}(\mathrm{d_y}) \times \psi/(\psi+\omega)$;
   for each element $d_\mathrm{z}$ in document cluster
   $A_j$, the membership degree $\mathrm{M}_{Aij}(\mathrm{d_z})$ of
   document $d_\mathrm{z}$ belonging to the new
   document cluster $A_{ij}$ is equal to
   $\mathrm{M}_{Ai}(\mathrm{d_z}) \times \psi/(\psi+\omega)$;
   merge document cluster $A_i^*$ with document
   cluster $A_l$ into a new cluster $A_{il}^*$;
   for each element $d_\mathrm{y}$ in document cluster
   $A_i^*$, the membership degree $\mathrm{M}_{Ail^*}(\mathrm{d_y})$ of
   document $d_\mathrm{y}$ belonging to the new
   document cluster $A_{il}^*$ is equal to
   $\mathrm{M}_{Ai^*}(\mathrm{d_y}) \times \psi/(\psi+\omega)$;
   for each element $d_\mathrm{u}$ in document cluster
   $A_l$, the membership degree $\mathrm{M}_{Ail^*}(\mathrm{d_u})$ of
   document $d_\mathrm{u}$ belonging to the
   new document cluster $A_{il}^*$ is equal to
   $\mathrm{M}_{Al}(\mathrm{d_u})$
  end
 else
  begin
```

```
    merge document cluster $A_i$ with document
    cluster $A_j$ into a new cluster $A_{ij}$;
    for each element $d_y$ in document cluster
    $A_i$, the membership degree $M_{Aij}(d_y)$ of
    document $d_y$ belonging to the new
    document cluster $A_{ij}$ is equal to
    $M_{Ai}(d_y)$;
    for each element $d_z$ in document cluster
    $A_j$, the membership degree $M_{Aij}(d_z)$ of
    document $d_z$ belonging to the new
    document cluster $A_{ij}$ is equal to $M_{Ai}(d_z)$;
   end
  end.
Step 5: Recalculate the degree of similarity between each pair of document clusters by taking the minimum of the degrees of similarity between any pair of documents, where the documents in each pair are taken from different document clusters.
Step 6: If the degree of similarity between any two document clusters is smaller than the threshold value $\alpha$, where $\alpha \in [0,1]$,
then Stop
  else go to Step 2.
```

After partitioning documents into several document clusters, the document cluster center v_k of document cluster A_k can be obtained by the following formula:

$$v_k = \frac{\sum_{i=1}^{n} \mu_{ki} d_i}{\sum_{i=1}^{n} \mu_{ki}} \tag{4}$$

where $d_i = \langle w_{i1}, w_{i2}, \ldots, w_{is} \rangle$, w_{ij} denotes the weight of term t_j in document d_i, μ_{ki} denotes the membership degree of document d_i belonging to document cluster A_k, $0 \le w_{ij} \le 1$, $0 \le \mu_{ki} \le 1$, and $1 \le j \le s$. This holds for all $\mathrm{k} = 1, 2, \ldots$.

IV. Comparison of the Clustering Results of the CLC Method and the Proposed FAHC Method

We have implemented the SLINK method [23], the CLC method [23] and the proposed FAHC method on a Pentium IV PC using Delphi version 5.0. We have chosen 247 research reports in the field of computer science [30] as the set of documents for clustering, which are a subset of a collection of research reports of the National Science Council (NSC), Taiwan, R.O.C. Each report consists of several parts, including a report identifier, a title, the researchers' names, a Chinese abstract, an English abstract,..., etc. Since the proposed method intends to deal with English documents, we take the English abstracts of the reports to represent the contents of the documents. However, since each selected document contains a large amount of words, these documents should be preprocessed to reduce the set of words into a manageable size before the clustering algorithms are applied on the selected documents. The selected documents are preprocessed in two steps. First, words appearing with high frequencies in all documents are eliminated. Then, the word extractor stems each remaining word to its "root form" [10]. The collection of these root-formatted words forms a set of index terms T for the document set. The weight "$w\_term\_document(t, d_i)$" of term t in document d_i is calculated as follows:

$$w\_term\_document(t, d_i) = \frac{\left(0.5 + 0.5 \frac{tf_{it}}{\underset{k=1,2,\ldots,L}{\mathrm{Max}} \ tf_{ik}}\right) \log \frac{N}{df_t}}{\underset{j=1,2,\ldots,L}{\mathrm{Max}} \left\{ \left(0.5 + 0.5 \frac{tf_{ij}}{\underset{k=1,2,\ldots,L}{\mathrm{Max}} \ tf_{ik}}\right) \log \frac{N}{df_j} \right\}} \tag{5}$$

where (5) is derived from [26], tf_{it} denotes the frequency of term t appearing in document d_i, df_t denotes the number of documents containing term t, L denotes the number of index terms contained in document d_i, and N denotes the number of documents in the corpus. The larger the value of $w\_\mathrm{term\_document}(t, d_i)$, the more important is the term t in document d_i. Note that the value of $w\_\mathrm{term\_document}(t, d_i)$ is normalized and is between zero and one.

After the weight of each index term in each document has been calculated, we can represent each document d_i as a vector shown as follows:

$$d_i = \langle w_{i1}, w_{i2}, \ldots, w_{is} \rangle \tag{6}$$

where $w_{ij} = w\_\mathrm{term\_document}(t_j, d_i)$ denotes the weight of term t_j in document d_i, $0 \le w_{ij} \le 1$, and s denotes the number of terms in the set of index terms.

The degree of similarity $\mathrm{sim}(d_i, d_j)$ between any two documents d_i and d_j is calculated by the following formula:

$$\mathrm{sim}(d_i, d_j) = \sum_{k=1,2,\ldots,s} \frac{\min(w_{ik}, w_{jk})}{\max(w_{ik}, w_{jk})} \tag{7}$$

where $\mathrm{sim}(d_i, d_j) \in [0, 1]$.

We use the 247 research reports [30] as the set of documents for clustering. Table I shows the SLINK method partitioning the 247 documents into document clusters for different "similarity threshold values" α, where the "similarity threshold value" α is between 0.04 and 0.08. When the "similarity threshold value" $\alpha = 0.04$, there is only one cluster produced. When the "similarity threshold value" α increases, there are more document clusters which contain more than one document being produced and there are more and more document clusters being produced containing only one document. Table II shows the complete link method [23] partitioning the 247 NSC documents into different numbers of document clusters for different "similarity threshold values" α, where the "similarity threshold value" α ranges from 0 to 0.025, and there is no document cluster containing only one document until α reaches or is equal to 0.025. From Tables I and II, we can see that the SLINK method produced more document clusters containing only one document than the complete link method. In this case, the SLINK method has a poorer clustering result than the complete link method due to the fact that it produces too many document clusters containing only one document compared to the complete link method.

TABLE I
DOCUMENT CLUSTERS PRODUCED BY THE SINGLE LINK METHOD [23] FOR DIFFERENT SIMILARITY THRESHOLD VALUES α

Similarity Threshold Value α	Number of Document Clusters Containing More Than One Document	Number of Document Clusters Containing One Document
0.04	1 (Totally 247 Documents)	0
0.07	1 (Totally 221 Documents)	26
0.075	2 (Totally 205 Documents)	42
0.079	5 (Totally 193 Documents)	54
0.08	7 (Totally 190 Documents)	57

TABLE II
DOCUMENT CLUSTERS PRODUCED BY THE COMPLETE LINK METHOD [23] FOR DIFFERENT SIMILARITY THRESHOLD VALUES α

Similarity Threshold Value α	Number of Document Clusters Containing More Than One Document	Number of Document Clusters Containing One Document
0	1 (Totally 247 Documents)	0
0.003	5 (Totally 247 Documents)	0
0.005	12 (Totally 247 Documents)	0
0.01	24 (Totally 247 Documents)	0
0.025	46 (Totally 246 Documents)	1

In the following, we compare the document clusters generated by the hierarchical clustering method [23] with the ones generated by the proposed FAHC method. It is obvious that the document clusters generated by a good clustering method should have a maximum degree of within-cluster similarity and a minimum degree of between-cluster similarity [3]. That is, the degree of similarity between documents in the same document clusters should be high and the degree of similarity between documents in different document clusters should be low. Thus, we adopt the heuristic measure, called the *category binding* (CB) presented in [3] to evaluate the clustering results. Assume that $\{A_1, A_2, \ldots, A_c\}$ is a set of document clusters obtained by the clustering methods, then the value of CB of the set of document clusters is calculated by the following formula:

$$CB(A_1, A_2, \ldots, A_c) = \frac{DW(A_1, A_2, \ldots, A_c)}{DI(A_1, A_2, \ldots, A_c)} \tag{8}$$

where $DW(A_1, A_2, \ldots, A_c)$ denotes the average degree of within-cluster similarity and $DI(A_1, A_2, \ldots, A_c)$ denotes the average degree of between-cluster similarity of the set of document clusters. From (8), we can see that a good clustering result should get a large value of $CB(A_1, A_2, \ldots, A_c)$.

The average degree of within-cluster similarity $DW(A_1, A_2, \ldots, A_c)$ is calculated as follows:

$$DW(A_1, A_2, \ldots, A_c) = \frac{\sum\limits_{k=1,2,\ldots,c} \Phi_{A_k}}{c} \tag{9}$$

where Φ_{A_k} denotes the cohesion of documents in document cluster A_k and is calculated by (10), as shown at the bottom of the page, where n_k denotes the number of documents in document cluster A_k, $\mu_k(d_l)$ denotes the membership degree of document d_l belonging to document cluster A_k, $\mu_k(d_m)$ denotes the membership degree of document d_m belonging to document cluster A_k, $\mathrm{sim}(d_l, d_m)$ denotes the degree of similarity between documents d_l and d_m calculated by (7), and T_v denotes a parameter representing the cohesion of documents in a single-instance document cluster [3].

The average degree of between-cluster similarity $DI(A_1, A_2, \ldots, A_c)$ is calculated as follows:

$$DI(A_1, A_2, \ldots, A_c) = \frac{\sum\limits_{i=1,2,\ldots,c-1} \sum\limits_{j=i+1,i+2,\ldots,c} \mathrm{sim}(A_i, A_j)}{\frac{c(c-1)}{2}}. \tag{11}$$

Here, $\mathrm{sim}(A_i, A_j)$ denotes the degree of similarity between document clusters A_i and A_j and is calculated as follows:

$$\mathrm{sim}(A_i, A_j) = \frac{\sum\limits_{l=1,2,\ldots,n_i} \sum\limits_{m=1,2,\ldots,n_j} \mu_i(d_l) \times \mu_j(d_m) \times \mathrm{sim}(d_l, d_m)}{\sum\limits_{l=1,2,\ldots,n_i} \mu_i(d_l) \times \sum\limits_{m=1,2,\ldots,n_j} \mu_j(d_m)} \tag{12}$$

where n_i denotes the number of documents in document cluster A_i, n_j denotes the number of documents in document cluster A_j, $\mu_i(d_l)$ denotes the membership degree of document d_l belonging to document cluster A_i, $\mu_j(d_m)$ denotes the membership degree of document d_m belonging to document cluster A_j, and $\mathrm{sim}(d_l, d_m)$ denotes the degree of similarity between any two documents d_l and d_m calculated by (7).

By observing the 247 NSC research reports [30], we can see that it is appropriate to partition these documents into 23–30 document clusters. Therefore, we tune the "similarity threshold value" α of the complete link method [23] to partition the 247 NSC research reports into an appropriate number of document clusters and compute the CB value based on (8)–(12) for each clustering result as shown in Table III. Moreover, we also tune the "similarity threshold value" α of the proposed fuzzy agglomerative hierarchical clustering method to partition the 247 NSC documents into an appropriate number of document clusters and compute the CB value based on (8)–(12) for each clustering result as shown in Table IV (when the difference threshold value $\lambda = 0.001$) and Table V (when the difference threshold value $\lambda = 0.0005$), where the values of λ are heuristically set to 0.001 and 0.0005, respectively

The "difference threshold value" λ of the proposed fuzzy agglomerative hierarchical clustering method can be regarded as

$$\Phi_{A_k} = \begin{cases} \dfrac{\sum\limits_{l=1,2,\ldots,n_k-1} \sum\limits_{m=l+1,l+2,\ldots,n_k} \mu_k(d_l) \times \mu_k(d_m) \times \mathrm{sim}(d_l, d_m)}{\sum\limits_{l=1,2,\ldots,n_k-1} \mu_k(d_l) \times \sum\limits_{m=l+1,l+2,\ldots,n_k} \mu_k(d_m)}, & \text{if } n_k > 1 \\ T_v, & \text{otherwise} \end{cases} \tag{10}$$

TABLE III
CB VALUES OF THE DOCUMENT CLUSTERS PRODUCED BY THE COMPLETE LINK METHOD [23]

Similarity Threshold Value α	Number of Document Clusters	CB Value
0.0095	23	7.809
0.01	24	7.909
0.0125	25	8.094
0.0127	26	8.253
0.0133	27	8.315
0.0134	28	8.432
0.0135	29	8.561
0.0138	30	8.584

TABLE IV
CB VALUES OF THE DOCUMENT CLUSTERS PRODUCED BY THE PROPOSED FUZZY AGGLOMERATIVE HIERARCHICAL CLUSTERING METHOD WHEN THE DIFFERENCE THRESHOLD VALUE $\lambda = 0.001$

Similarity Threshold Value α	Number of Document Clusters	CB Value
0.01	23	7.684
0.0125	24	7.671
0.013	25	7.598
0.0135	26	7.958
0.01357	27	8.075
0.0136	28	8.126
0.01365	29	8.250
0.0137	30	8.448

TABLE V
CB VALUES OF THE DOCUMENT CLUSTERS PRODUCED BY THE PROPOSED FUZZY AGGLOMERATIVE HIERARCHICAL CLUSTERING METHOD WHEN THE DIFFERENCE THRESHOLD VALUE $\lambda = 0.0005$

Similarity Threshold Value α	Number of Document Clusters	CB Value
0.009	23	7.829
0.01	24	7.929
0.011	25	8.114
0.0125	26	8.272
0.013	27	8.333
0.0134	28	8.450
0.0135	29	8.586
0.0137	30	8.608

a parameter to control the "fuzziness" of the resulting document clusters. The larger the "difference threshold value" λ, the higher the "fuzziness" of the resulting document clusters, where $\lambda \in [0, 1]$. When the "difference threshold value" λ is set to 0.001, we found that many documents are partitioned into more than one document cluster. When the "difference threshold value" λ is set to 0.0005, only few documents are partitioned into more than one document cluster. From Tables III–V, we can see that the CB values of the document clusters produced by the complete link method [23] are larger than the ones of the document clusters produced by the proposed fuzzy agglomerative hierarchical clustering method when the "difference threshold value" λ is set to 0.001, but are smaller than the ones of the document clusters produced by the proposed fuzzy agglomerative hierarchical clustering method when the "difference threshold value" λ is set to 0.0005. The reason is that the resulting document clusters produced by the proposed fuzzy agglomerative hierarchical clustering method, when the "difference threshold value" λ is set to 0.001, are too "fuzzy" and the average degree of between-cluster similarity is high due to the fact that many document clusters contain common documents. Therefore, we should decrease the fuzziness of the document clusters. The experimental results show that the document clusters produced by the proposed fuzzy agglomerative hierarchical clustering method with a low degree of fuzziness (i.e., $\lambda = 0.0005$) are better than the crisp document clusters produced by the complete link method [23].

V. FUZZY LOGIC RULES DISCOVERY

In [20], Kraft *et al.* presented a method to construct fuzzy logic rules based on the document clusters and their cluster centers. The fuzzy logic rules constructed in [20] have the following format:

$$[t_i \geq w_i] \rightarrow [t_j \geq w_j] \tag{13}$$

where w_i and w_j denote the weights of index terms t_i and t_j in the document cluster center representation, $0 \leq w_i \leq 1$, and $0 \leq w_j \leq 1$. The meaning of the rule is that whenever term t_i's weight (in a document or query) is at least w_i, the related term t_j's weight (in the same document or query) should be at least w_j. A fuzzy logic rule is constructed only when terms t_i and t_j both have high weights in the same document cluster center representation. Therefore, the fuzzy logic rule can represent the association relationship between index terms t_i and t_j. These rules can then be applied to modify the user's original query and thus increase the retrieval effectiveness.

However, the fuzzy logic rules discovery method presented in [20] only considers the clustering results of documents obtained by the two clustering methods (i.e., the complete link clustering method [23] and the fuzzy c-means clustering method [1]). We believe that the document clusters formed in the middle of the clustering process can also provide some useful information. For example, in the clustering process using the agglomerative hierarchical clustering method (e.g., the complete link method and the proposed fuzzy agglomerative hierarchical clustering method), two document clusters are merged to form a larger document cluster. The resulting document cluster can be regarded as the parent document cluster of the original document clusters. The index terms having large degrees of weights in the parent document cluster center should be more general than the ones with large degrees of weights in the child document cluster center, due to the fact that the index terms having large weights in the parent document cluster center are contained in more documents.

In this section, we present a method for fuzzy logic rules discovery which constructs fuzzy logic rules representing more kinds of relationships between index terms than the ones presented in [20]. The fuzzy logic rules are constructed based on the document cluster centers. Similar to the fuzzy logic rule construction method presented in [20], for each document cluster center, terms are sorted in a descending sequence according to their weights in the document cluster center. Then, the first m terms, where $m \geq 2$, as well as their weights are extracted. However, unlike the method presented in [20], we build term pairs not only with chosen terms from the same document cluster center but also from document cluster centers that have parent–children relationships. Assume that term t_i is chosen from the document cluster center of document cluster A_k and term t_j is chosen from the document cluster center of document cluster A_l, then the term pair has the form of $\langle [t_i, w_{ki}], [t_j, w_{lj}] \rangle$, where w_{ki} is the weight of term t_i in the document cluster center of document cluster A_k, w_{lj} is the weight of term t_j in the document cluster center of document cluster A_l, $0 \leq w_{ki} \leq 1$, and $0 \leq w_{lj} \leq 1$. The generated term pairs can be categorized into three categories according to the relationship between source document cluster centers containing terms t_i and t_j. These three categories are as follows.

1) Positive Association Category: If terms t_i and t_j are in the same document cluster center, then the corresponding term pair belongs to this category. The term pairs in this category represent a positive association relationship [20] between terms t_i and t_j due to the fact that terms t_i and t_j are contained in similar documents and should describe similar concepts.
2) Generalization Category: If the document cluster center containing term t_j is the parent of the document cluster center containing t_i, then the corresponding term pair belongs to this category. The term pairs in this category denote that term t_j is more general than term t_i due to the fact that term t_j belongs to more documents than term t_i does.
3) Specialization Category: If the document cluster center containing term t_j is a child of the document cluster containing t_i, then the corresponding term pair belongs to this category. The term pairs in this category denote that term t_j is more specific than term t_i due to the fact that term t_j belongs to fewer documents than term t_i does.

If the same term pair occurs several times in the same category with different weights, then the minimal weight for each term among all term pairs will be taken as the aggregated weight of the term. Finally, for each term pair in the form of $\langle [t_i, w_{ki}], [t_j, w_{lj}] \rangle$, we can build rules according to the following cases.

Case 1) If the term pair $\langle [t_i, w_{ki}], [t_j, w_{lj}] \rangle$ belongs to the positive association category, then we build two fuzzy logic rules

$$[t_i \geq w_{ki}] \rightarrow [t_j \geq w_{lj}]$$

and

$$[t_j \geq w_{lj}] \rightarrow [t_i \geq w_{ki}]$$

for this term pair. The meaning of the pair of rules is that the occurrence of the term t_i with a weight at least w_{ki} should always be accompanied by the term t_j with a weight at least w_{lj}, and *vice versa.*

Case 2) If the term pair $\langle [t_i, w_{ki}], [t_j, w_{lj}] \rangle$ belongs to the generalization category and the term pair $\langle [t_j, w_{lj}], [t_i, w_{ki}] \rangle$ belongs to the specialization category, then we build two fuzzy logic rules

$$[t_i \geq w_{ki}] \rightarrow [t_j \geq w_{lj}]$$

and

$$[t_j \geq w_{lj}] \rightarrow [t_i \geq w_{ki}]$$

for these term pairs. The meaning of the above rules is that the occurrence of the term t_i with a weight at least w_{ki} should always be accompanied by the term t_j with a weight at least w_{lj}, and *vice versa.*

VI. Query Modification

After the fuzzy logic rules are constructed by the proposed fuzzy logic rule discovery method, we can use these fuzzy logic rules to modify the user's queries based on [4]. According to the definition of the fuzzy logic system [4], the fuzzy logic rule has the following form:

$$[t_i \geq w_{ki}] \rightarrow [t_j \geq w_{lj}]$$

which is a well-formed formula consisting of propositions of the form $[A \leq \alpha]$ or $[A \geq \alpha]$ and the logical connectives, i.e., $\wedge$, $\vee$, $\neg$ and $\rightarrow$.

A user's query q can be represented by a query descriptor vector $\overline{q}$ shown as follows:

$$\overline{q} = \langle w_{q1}, w_{q2}, \ldots, w_{qs} \rangle$$

where each element w_{qi} denotes the desired strength of term t_i in the retrieved documents, and $w_{qi} \in [0, 1]$ or $=$ " $-$ ". If $w_{qi} = 0$, then it indicates that the user hopes that the retrieved documents do not possess the term t_i. Furthermore, if the user considers that some terms may be neglected, i.e., to include those terms or not would have no substantial effects on the result, then the user does not have to assign degrees of strength with respect to such terms in the query descriptor vector. The symbol "$-$" is used for labeling a neglected term. If $w_{qi} = -$, then it indicates that the term t_i will not be considered in the document retrieval process.

Let R be a set of the generated fuzzy logic rules, $R = \{r_1, r_2, \ldots, r_m\}$, where each fuzzy logic rule has the following form:

$$[t_i \geq w_{ki}] \rightarrow [t_j \geq w_{lj}]$$

which is applied to modify the user's query descriptor vector $\overline{q}$ if $w_{qi} \geq w_{ki}$ and $w_{qj} \leq w_{lj}$ or if $w_{qi} \geq w_{ki}$ and $w_{qj} = -$. The modified user's query descriptor vector $\overline{q}^*$ is similar to the original user's query descriptor vector except that w_{qj} is set to w_{lj}. The modified user's query descriptor vector $\overline{q}^*$ will be used to retrieve relevant documents. Assume that the modified user's query descriptor vector $\overline{q}^*$ is as follows:

$$\overline{q}^* = \langle w_{q1}, w_{q2}, \ldots, w_{qs} \rangle$$

where $0 \leq w_{qi} \leq 1$ or $w_{qi} = -$. If $w_{qi} = -$, then it indicates that the term t_i is a neglected term and it will not be considered in the retrieval process. Furthermore, assume that each document d_i can be represented by a vector shown as follows:

$$d_i = \langle w_{i1}, w_{i2}, \ldots, w_{is} \rangle$$

where w_{ij} represents the weight of term t_j in document d_i, and $0 \leq w_{ij} \leq 1$. Then, the retrieval status value $RSV(d_i)$ of

[sit ≥ 0.28] ⟶ [execut ≥ 0.26],

[execut ≥ 0.26] ⟶ [sit ≥ 0.28],

[given ≥ 0.22] ⟶ [invers ≥ 0.18],

[invers ≥ 0.18] ⟶ [given ≥ 0.22],

[queri ≥ 0.23] ⟶ [includ ≥ 0.20],

[includ ≥ 0.20] ⟶ [queri ≥ 0.23],

[imag ≥ 0.31] ⟶ [restor ≥ 0.24],

[restor ≥ 0.24] ⟶ [imag ≥ 0.31],

[stochast ≥ 0.22] ⟶ [exampl ≥ 0.16],

[exampl ≥ 0.16] ⟶ [stochast ≥ 0.22],

[composit ≥ 0.22] ⟶ [methodolog ≥ 0.22],

[methodol ≥ 0.22] ⟶ [composit ≥ 0.22],

[relev ≥ 0.25] ⟶ [queri ≥ 0.23],

[queri ≥ 0.23] ⟶ [relev ≥ 0.25],

[object ≥ 0.24] ⟶ [queri ≥ 0.24],

[queri ≥ 0.24] ⟶ [object ≥ 0.24],

[multimed ≥ 0.24] ⟶ [manag ≥ 0.24],

[manag ≥ 0.24] ⟶ [multimedia ≥ 0.24],

[step ≥ 0.22] ⟶ [paramet ≥ 0.21],

[paramet ≥ 0.21] ⟶ [step ≥ 0.22],

[edg ≥ 0.55] ⟶ [threshold ≥ 0.35],

[threshold ≥ 0.35] ⟶ [edg ≥ 0.55],

[word ≥ 0.46] ⟶ [natur ≥ 0.29],

[natur ≥ 0.29] ⟶ [word ≥ 0.46],

[fuzzi ≥ 0.35] ⟶ [input ≥ 0.25],

[input ≥ 0.25] ⟶ [fuzzi ≥ 0.35],

[fact ≥ 0.24] ⟶ [rul ≥ 0.23],

[rul ≥ 0.23] ⟶ [fact ≥ 0.24],

[mpeg ≥ 0.47] ⟶ [standard ≥ 0.37],

[standard ≥ 0.37] ⟶ [mpeg ≥ 0.47],

[constraint ≥ 0.55] ⟶ [conceptu ≥ 0.50],

[conceptu ≥ 0.50] ⟶ [constraint ≥ 0.55],

[pixel ≥ 0.32] ⟶ [imag ≥ 0.26],

[imag ≥ 0.26] ⟶ [pixel ≥ 0.32],

[telephon ≥ 0.36] ⟶ [hospit ≥ 0.33],

[hospit ≥ 0.33] ⟶ [telephon ≥ 0.36],

[genet ≥ 0.38] ⟶ [learn ≥ 0.32],

[learn ≥ 0.32] ⟶ [genet ≥ 0.38],

[faster ≥ 0.34] ⟶ [unless ≥ 0.31],

[unless ≥ 0.31] ⟶ [faster ≥ 0.34],

[engin ≥ 0.41] ⟶ [expert ≥ 0.38],

[expert ≥ 0.38] ⟶ [engin ≥ 0.41],

[toler ≥ 0.41] ⟶ [difficulti ≥ 0.33],

[difficulti ≥ 0.33] ⟶ [toler ≥ 0.41],

[compress ≥ 0.45] ⟶ [cod ≥ 0.43],

[cod ≥ 0.43] ⟶ [compress ≥ 0.45],

[distanc ≥ 0.42] ⟶ [group ≥ 0.41],

[group ≥ 0.41] ⟶ [distanc ≥ 0.42],

[competit ≥ 0.46] ⟶ [market ≥ 0.43],

[market ≥ 0.43] ⟶ [competit ≥ 0.46],

[qualit ≥ 0.68] ⟶ [quantit ≥ 0.59],

[quantit ≥ 0.59] ⟶ [qualit ≥ 0.68],

[river ≥ 0.59] ⟶ [soil ≥ 0.59],

[soil ≥ 0.59] ⟶ [river ≥ 0.59],

[variabl ≥ 0.42] ⟶ [ann ≥ 0.41],

[ann ≥ 0.41] ⟶ [variabl ≥ 0.42],

[seek ≥ 0.79] ⟶ [instruct ≥ 0.60],

[instruct ≥ 0.60] ⟶ [seek ≥ 0.79],

[social ≥ 0.74] ⟶ [activ ≥ 0.47],

[activ ≥ 0.47] ⟶ [social ≥ 0.74],

Fig. 1. Set of generated fuzzy logic rules based on the term pairs belonging to the positive association category.

document d_i with respect to the user's query can be calculated as follows:

$$RSV(d_i) = \frac{\sum\limits_{j=1,2,\ldots,s \text{ and } w_{qj}\neq -} T(w_{qj}, w_{ij})}{k} \tag{14}$$

where k is the number of nonneglected terms. Here, T is a similarity function [8] to calculate the degree of similarity between two real values between zero and one and is given as

$$T(w_{qj}, w_{ij}) = 1 - |w_{qj} - w_{ij}|. \tag{15}$$

After the retrieval status value $RSV(d_i)$ of each document d_i with respect to the user's query is obtained, we normalize the value of $RSV(d_i)$ by dividing it by the maximum value among the values of $RSV(d_1), RSV(d_2), \ldots,$ and $RSV(d_N)$, where N is the number of collected documents. If a document d_i wants to be retrieved, then the retrieval status value $RSV(d_i)$ of document d_i should be larger than or equal to the "query threshold value" θ given by the user, where $0 \leq \theta \leq 1$.

We have implemented the proposed query modification method for document retrieval based on Delphi version 5.0 on a Pentium 4 PC using the 247 NSC research reports [30] as described in Section IV. Furthermore, we also implemented the query modification method proposed by Kraft *et al.* [20] for making a comparison. The experimental results show that although the modified queries by applying the query modification method presented in [20] can be useful to improve the precision rate of the retrieval results, the queries modified by the proposed method can achieve a higher precision rate. For example, assume that the user wants to retrieve documents about the topic "natural language processing" and assume that 13 documents among the 247 NSC research reports are relevant to this topic. Assume that weights of the terms "natural," "language," and "processing" in the user's query q_1 are 0.9, 0.9, and 0.8, respectively, and the other terms are neglected, denoted by the symbol $-$. Assume that the documents have been clustered by the proposed fuzzy hierarchical clustering method with the similarity threshold value $\alpha = 0.015$ and the difference threshold value $\lambda = 0.0005$. Then, the fuzzy logic rules are generated as shown in Figs. 1–3.

The query modification method presented in [20] uses the generated fuzzy logic rules based on the term pairs belonging to the "Positive Association Category" to modify the original query. On the other hand, the proposed query modification method uses the generated fuzzy logic rules based on the term pairs not only belonging to the "Positive Association Category," but also belonging to the "Generalization Category" and belonging to the "Specialization Category" to modify the original user's query q_1. By applying the query modification method presented in [20], the original user's query q_1 can be modified into the user's query q_1^+ which has the same weights of the

[tranch ≥ 0.50] ⟶ [variabl ≥ 0.42],
[tranch ≥ 0.50] ⟶ [ann ≥ 0.41],
[financi ≥ 0.77] ⟶ [variabl ≥ 0.42],
[financi ≥ 0.77] ⟶ [ann ≥ 0.41],
[insur ≥ 0.50] ⟶ [variabl ≥ 0.42],
[insur ≥ 0.50] ⟶ [ann ≥ 0.41],
[easier ≥ 0.71] ⟶ [qualit ≥ 0.68],
[easier ≥ 0.71] ⟶ [quantit ≥ 0.59],
[industri ≥ 0.59] ⟶ [competit ≥ 0.46],
[industri ≥ 0.59] ⟶ [market ≥ 0.43],
[euclidean ≥ 0.50] ⟶ [distanc ≥ 0.42],
[euclidean ≥ 0.50] ⟶ [group ≥ 0.41],
[fractal ≥ 0.56] ⟶ [compress ≥ 0.45],
[fractal ≥ 0.56] ⟶ [cod ≥ 0.43],
[neural ≥ 0.38] ⟶ [toler ≥ 0.41],
[neural ≥ 0.38] ⟶ [difficulti ≥ 0.33],
[infer ≥ 0.45] ⟶ [engin ≥ 0.41],
[infer ≥ 0.45] ⟶ [expert ≥ 0.38],
[chess ≥ 1.00] ⟶ [engin ≥ 0.41],
[chess ≥ 1.00] ⟶ [expert ≥ 0.38],
[play ≥ 0.67] ⟶ [engin ≥ 0.41],
[play ≥ 0.67] ⟶ [expert ≥ 0.38],
[overlap ≥ 0.44] ⟶ [faster ≥ 0.34],
[overlap ≥ 0.44] ⟶ [unless ≥ 0.31],
[subimag ≥ 0.50] ⟶ [faster ≥ 0.34],
[subimag ≥ 0.50] ⟶ [unless ≥ 0.31],
[weight ≥ 0.48] ⟶ [genet ≥ 0.38],
[weight ≥ 0.48] ⟶ [learn ≥ 0.32],
[crossov ≥ 0.41] ⟶ [genet ≥ 0.38],
[crossov ≥ 0.41] ⟶ [learn ≥ 0.32],
[mobil ≥ 0.41] ⟶ [genet ≥ 0.38],
[mobil ≥ 0.41] ⟶ [learn ≥ 0.32],
[protocol ≥ 0.46] ⟶ [telephon ≥ 0.36],
[protocol ≥ 0.46] ⟶ [hospit ≥ 0.33],
[softwar ≥ 0.40] ⟶ [telephon ≥ 0.36],
[softwar ≥ 0.40] ⟶ [hospit ≥ 0.33],
[neighbor ≥ 0.45] ⟶ [pixel ≥ 0.32],
[neighbor ≥ 0.45] ⟶ [imag ≥ 0.26],
[evid ≥ 0.52] ⟶ [pixel ≥ 0.32],
[evid ≥ 0.52] ⟶ [imag ≥ 0.26],
[belief ≥ 0.45] ⟶ [pixel ≥ 0.32],
[belief ≥ 0.45] ⟶ [imag ≥ 0.26],
[di ≥ 0.50] ⟶ [mpeg ≥ 0.47],
[di ≥ 0.50] ⟶ [standard ≥ 0.37],
[edi ≥ 0.50] ⟶ [mpeg ≥ 0.47],
[edi ≥ 0.50] ⟶ [standard ≥ 0.37],
[lossless ≥ 0.57] ⟶ [mpeg ≥ 0.47],
[lossless ≥ 0.57] ⟶ [standard ≥ 0.37],
[expert ≥ 0.39] ⟶ [fact ≥ 0.24],
[expert ≥ 0.39] ⟶ [rul ≥ 0.23],
[occurr ≥ 0.34] ⟶ [fact ≥ 0.24],
[occurr ≥ 0.34] ⟶ [rul ≥ 0.23],
[fundame ≥ 0.33] ⟶ [fact ≥ 0.24],
[fundame ≥ 0.33] ⟶ [rul ≥ 0.23],
[number ≥ 0.25] ⟶ [fuzzi ≥ 0.35],
[number ≥ 0.25] ⟶ [input ≥ 0.25],
[trad ≥ 0.60] ⟶ [fuzzi ≥ 0.35],
[trad ≥ 0.60] ⟶ [input ≥ 0.25],
[daili ≥ 0.45] ⟶ [fuzzi ≥ 0.35],
[daili ≥ 0.45] ⟶ [input ≥ 0.25],
[dictionari ≥ 0.44] ⟶ [word ≥ 0.46],
[dictionari ≥ 0.44] ⟶ [natur ≥ 0.29],
[corpu ≥ 0.39] ⟶ [word ≥ 0.46],
[corpu ≥ 0.39] ⟶ [natur ≥ 0.29],
[speech ≥ 0.37] ⟶ [word ≥ 0.46],
[speech ≥ 0.37] ⟶ [natur ≥ 0.29],
[adopt ≥ 0.41] ⟶ [edg ≥ 0.55],
[adopt ≥ 0.41] ⟶ [threshold ≥ 0.35],
[region ≥ 0.51] ⟶ [edg ≥ 0.55],
[region ≥ 0.51] ⟶ [threshold ≥ 0.35],
[control ≥ 0.34] ⟶ [step ≥ 0.22],
[control ≥ 0.34] ⟶ [paramet ≥ 0.21],
[tun ≥ 0.30] ⟶ [step ≥ 0.22],
[tun ≥ 0.30] ⟶ [paramet ≥ 0.21],
[br ≥ 0.32] ⟶ [step ≥ 0.22],
[br ≥ 0.32] ⟶ [paramet ≥ 0.21],
[think ≥ 0.28] ⟶ [step ≥ 0.22],
[think ≥ 0.28] ⟶ [paramet ≥ 0.21],
[hypermedia ≥ 0.27] ⟶ [multimedia ≥ 0.24],
[hypermedia ≥ 0.27] ⟶ [manag ≥ 0.24],
[product ≥ 0.29] ⟶ [multimedia ≥ 0.24],
[product ≥ 0.29] ⟶ [manag ≥ 0.24],
[area ≥ 0.27] ⟶ [multimedia ≥ 0.24],
[area ≥ 0.27] ⟶ [manag ≥ 0.24],
[schema ≥ 0.26] ⟶ [object ≥ 0.24],
[schema ≥ 0.26] ⟶ [queri ≥ 0.24],
[video ≥ 0.34] ⟶ [object ≥ 0.24],
[video ≥ 0.34] ⟶ [queri ≥ 0.24],
[multimedia ≥ 0.33] ⟶ [object ≥ 0.24],
[multimedia ≥ 0.33] ⟶ [queri ≥ 0.24],
[fuzzi ≥ 0.32] ⟶ [relev ≥ 0.25],
[fuzzi ≥ 0.32] ⟶ [queri ≥ 0.23],
[repres ≥ 0.31] ⟶ [relev ≥ 0.25],
[repres ≥ 0.31] ⟶ [queri ≥ 0.23],
[ir ≥ 0.26] ⟶ [relev ≥ 0.25],
[ir ≥ 0.26] ⟶ [queri ≥ 0.23],
[librari ≥ 0.45] ⟶ [composit ≥ 0.22],
[librari ≥ 0.45] ⟶ [methodolog ≥ 0.22],
[faculti ≥ 0.43] ⟶ [composit ≥ 0.22],
[faculti ≥ 0.43] ⟶ [methodolog ≥ 0.22],
[concurr ≥ 0.33] ⟶ [stochast ≥ 0.22],
[concurr ≥ 0.33] ⟶ [exampl ≥ 0.16],
[avoid ≥ 0.27] ⟶ [stochast ≥ 0.22],
[avoid ≥ 0.27] ⟶ [exampl ≥ 0.16],
[solv ≥ 0.37] ⟶ [stochast ≥ 0.22],
[solv ≥ 0.37] ⟶ [exampl ≥ 0.16],
[nois ≥ 0.45] ⟶ [imag ≥ 0.31],
[nois ≥ 0.45] ⟶ [restor ≥ 0.24],
[filter ≥ 0.35] ⟶ [imag ≥ 0.31],
[filter ≥ 0.35] ⟶ [restor ≥ 0.24],
[motion ≥ 0.35] ⟶ [imag ≥ 0.31],
[motion ≥ 0.35] ⟶ [restor ≥ 0.24],
[oo ≥ 0.31] ⟶ [queri ≥ 0.23],
[oo ≥ 0.31] ⟶ [includ ≥ 0.20],
[text ≥ 0.41] ⟶ [queri ≥ 0.23],
[text ≥ 0.41] ⟶ [includ ≥ 0.20],
[ir ≥ 0.40] ⟶ [includ ≥ 0.20],
[hcc ≥ 0.25] ⟶ [given ≥ 0.22],
[hcc ≥ 0.25] ⟶ [invers ≥ 0.18],
[strok ≥ 0.25] ⟶ [given ≥ 0.22],
[strok ≥ 0.25] ⟶ [invers ≥ 0.18],
[coeffici ≥ 0.37] ⟶ [given ≥ 0.22],
[coeffici ≥ 0.37] ⟶ [invers ≥ 0.18],
[transact ≥ 0.46] ⟶ [sit ≥ 0.28],
[transact ≥ 0.46] ⟶ [execut ≥ 0.26],
[www ≥ 0.45] ⟶ [sit ≥ 0.28],
[www ≥ 0.45] ⟶ [execut ≥ 0.26],
[server ≥ 0.41] ⟶ [sit ≥ 0.28],
[server ≥ 0.41] ⟶ [execut ≥ 0.26],

Fig. 2. Set of generated fuzzy logic rules based on the term pairs belonging to the generalization category.

[variabl	≥	0.42]	⟶	[tranch	≥	0.50],
[ann	≥	0.41]	⟶	[tranch	≥	0.50],
[variabl	≥	0.42]	⟶	[financi	≥	0.77],
[ann	≥	0.41]	⟶	[financi	≥	0.77],
[variabl	≥	0.42]	⟶	[insur	≥	0.50],
[ann	≥	0.41]	⟶	[insur	≥	0.50],
[qualit	≥	0.68]	⟶	[easier	≥	0.71],
[quantit	≥	0.59]	⟶	[easier	≥	0.71],
[competit	≥	0.46]	⟶	[industri	≥	0.59],
[market	≥	0.43]	⟶	[industri	≥	0.59],
[distanc	≥	0.42]	⟶	[euclidean	≥	0.50],
[group	≥	0.41]	⟶	[euclidean	≥	0.50],
[compress	≥	0.45]	⟶	[fractal	≥	0.56],
[cod	≥	0.43]	⟶	[fractal	≥	0.56],
[toler	≥	0.41]	⟶	[neural	≥	0.38],
[difficulti	≥	0.33]	⟶	[neural	≥	0.38],
[engin	≥	0.41]	⟶	[infer	≥	0.45],
[expert	≥	0.38]	⟶	[infer	≥	0.45],
[engin	≥	0.41]	⟶	[chess	≥	1.00],
[expert	≥	0.38]	⟶	[chess	≥	1.00],
[engin	≥	0.41]	⟶	[play	≥	0.67],
[expert	≥	0.38]	⟶	[play	≥	0.67],
[faster	≥	0.34]	⟶	[overlap	≥	0.44],
[unless	≥	0.31]	⟶	[overlap	≥	0.44],
[faster	≥	0.34]	⟶	[subimag	≥	0.50],
[unless	≥	0.31]	⟶	[subimag	≥	0.50],
[genet	≥	0.38]	⟶	[weight	≥	0.48],
[learn	≥	0.32]	⟶	[weight	≥	0.48],
[genet	≥	0.38]	⟶	[crossov	≥	0.41],
[learn	≥	0.32]	⟶	[crossov	≥	0.41],
[genet	≥	0.38]	⟶	[mobil	≥	0.41],
[learn	≥	0.32]	⟶	[mobil	≥	0.41],
[telephon	≥	0.36]	⟶	[protocol	≥	0.46],
[hospit	≥	0.33]	⟶	[protocol	≥	0.46],
[telephon	≥	0.36]	⟶	[softwar	≥	0.40],
[hospit	≥	0.33]	⟶	[softwar	≥	0.40],
[pixel	≥	0.32]	⟶	[neighbor	≥	0.45],
[imag	≥	0.26]	⟶	[neighbor	≥	0.45],
[pixel	≥	0.32]	⟶	[evid	≥	0.52],
[imag	≥	0.26]	⟶	[evid	≥	0.52],
[pixel	≥	0.32]	⟶	[belief	≥	0.45],
[imag	≥	0.26]	⟶	[belief	≥	0.45],
[mpeg	≥	0.47]	⟶	[di	≥	0.50],
[standard	≥	0.37]	⟶	[di	≥	0.50],
[mpeg	≥	0.47]	⟶	[edi	≥	0.50],
[standard	≥	0.37]	⟶	[edi	≥	0.50],
[mpeg	≥	0.47]	⟶	[lossless	≥	0.57],
[standard	≥	0.37]	⟶	[lossless	≥	0.57],
[fact	≥	0.24]	⟶	[expert	≥	0.39],
[rul	≥	0.23]	⟶	[expert	≥	0.39],
[fact	≥	0.24]	⟶	[occurr	≥	0.34],
[rul	≥	0.23]	⟶	[occurr	≥	0.34],
[fact	≥	0.24]	⟶	[fundament	≥	0.33],
[rul	≥	0.23]	⟶	[fundament	≥	0.33],
[fuzzi	≥	0.35]	⟶	[number	≥	0.25],
[input	≥	0.25]	⟶	[number	≥	0.25],
[fuzzi	≥	0.35]	⟶	[trad	≥	0.60],
[input	≥	0.25]	⟶	[trad	≥	0.60],
[fuzzi	≥	0.35]	⟶	[daili	≥	0.45],
[input	≥	0.25]	⟶	[daili	≥	0.45],
[word	≥	0.46]	⟶	[dictionari	≥	0.44],
[natur	≥	0.29]	⟶	[dictionari	≥	0.44],
[word	≥	0.46]	⟶	[corpu	≥	0.39],
[natur	≥	0.29]	⟶	[corpu	≥	0.39],
[word	≥	0.46]	⟶	[speech	≥	0.37],
[natur	≥	0.29]	⟶	[speech	≥	0.37],
[edg	≥	0.55]	⟶	[adopt	≥	0.41],
[threshold	≥	0.35]	⟶	[adopt	≥	0.41],
[edg	≥	0.55]	⟶	[region	≥	0.51],
[threshold	≥	0.35]	⟶	[region	≥	0.51],
[step	≥	0.22]	⟶	[control	≥	0.34],
[paramet	≥	0.21]	⟶	[control	≥	0.34],
[step	≥	0.22]	⟶	[tun	≥	0.30],
[paramet	≥	0.21]	⟶	[tun	≥	0.30],
[step	≥	0.22]	⟶	[br	≥	0.32],
[paramet	≥	0.21]	⟶	[br	≥	0.32],
[step	≥	0.22]	⟶	[think	≥	0.28],
[paramet	≥	0.21]	⟶	[think	≥	0.28],
[multimedia	≥	0.24]	⟶	[hypermedia	≥	0.27],
[manag	≥	0.24]	⟶	[hypermedia	≥	0.27],
[multimedia	≥	0.24]	⟶	[product	≥	0.29],
[manag	≥	0.24]	⟶	[product	≥	0.29],
[multimedia	≥	0.24]	⟶	[area	≥	0.27],
[manag	≥	0.24]	⟶	[area	≥	0.27],
[object	≥	0.24]	⟶	[schema	≥	0.26],
[queri	≥	0.24]	⟶	[schema	≥	0.26],
[object	≥	0.24]	⟶	[video	≥	0.34],
[queri	≥	0.24]	⟶	[video	≥	0.34],
[object	≥	0.24]	⟶	[multimedia	≥	0.33],
[queri	≥	0.24]	⟶	[multimedia	≥	0.33],
[relev	≥	0.25]	⟶	[fuzzi	≥	0.32],
[queri	≥	0.23]	⟶	[fuzzi	≥	0.32],
[relev	≥	0.25]	⟶	[repres	≥	0.31],
[queri	≥	0.23]	⟶	[repres	≥	0.31],
[relev	≥	0.25]	⟶	[ir	≥	0.26],
[queri	≥	0.23]	⟶	[ir	≥	0.26],
[composit	≥	0.22]	⟶	[librari	≥	0.45],
[methodologi	≥	0.22]	⟶	[librari	≥	0.45],
[composit	≥	0.22]	⟶	[faculti	≥	0.43],
[methodologi	≥	0.22]	⟶	[faculti	≥	0.43],
[stochast	≥	0.22]	⟶	[concurr	≥	0.33],
[exampl	≥	0.16]	⟶	[concurr	≥	0.33],
[stochast	≥	0.22]	⟶	[avoid	≥	0.27],
[exampl	≥	0.16]	⟶	[avoid	≥	0.27],
[stochast	≥	0.22]	⟶	[solv	≥	0.37],
[exampl	≥	0.16]	⟶	[solv	≥	0.37],
[imag	≥	0.31]	⟶	[nois	≥	0.45],
[restor	≥	0.24]	⟶	[nois	≥	0.45],
[imag	≥	0.31]	⟶	[filter	≥	0.35],
[restor	≥	0.24]	⟶	[filter	≥	0.35],
[imag	≥	0.31]	⟶	[motion	≥	0.35],
[restor	≥	0.24]	⟶	[motion	≥	0.35],
[queri	≥	0.23]	⟶	[oo	≥	0.31],
[includ	≥	0.20]	⟶	[oo	≥	0.31],
[queri	≥	0.23]	⟶	[text	≥	0.41],
[includ	≥	0.20]	⟶	[text	≥	0.41],
[includ	≥	0.20]	⟶	[ir	≥	0.40],
[given	≥	0.22]	⟶	[hcc	≥	0.25],
[invers	≥	0.18]	⟶	[hcc	≥	0.25],
[given	≥	0.22]	⟶	[strok	≥	0.25],
[invers	≥	0.18]	⟶	[strok	≥	0.25],
[given	≥	0.22]	⟶	[coeffici	≥	0.37],
[invers	≥	0.18]	⟶	[coeffici	≥	0.37],
[sit	≥	0.28]	⟶	[transact	≥	0.46],
[execut	≥	0.26]	⟶	[transact	≥	0.46],
[sit	≥	0.28]	⟶	[www	≥	0.45],
[execut	≥	0.26]	⟶	[www	≥	0.45],
[sit	≥	0.28]	⟶	[server	≥	0.41],
[execut	≥	0.26]	⟶	[server	≥	0.41],

Fig. 3. Set of generated fuzzy logic rules based on the term pairs belonging to the specialization category.

TABLE VI

PRECISION RATES AND THE RECALL RATES WITH RESPECT TO DIFFERENT USER'S QUERIES FOR DIFFERENT RETRIEVAL THRESHOLD VALUES

Retrieval Threshold	The Original User's Query q_1		The Modified User's Query q_1^+ Based on [20]		The Modified User's Query q_1* Based on the Proposed Method	
Value θ	Precision Rate	Recall Rate	Precision Rate	Recall Rate	Precision Rate	Recall Rate
0.1	10%	100%	9%	100%	25%	100%
0.2	11%	100%	26%	100%	50%	100%
0.3	27%	100%	33%	100%	63%	92%
0.4	33%	100%	52%	100%	80%	92%
0.5	45%	100%	63%	92%	83%	77%
0.6	60%	92%	80%	92%	91%	77%
0.7	50%	54%	80%	92%	100%	54%
0.8	57%	31%	92%	85%	100%	38%
0.9	100%	15%	100%	31%	100%	8%

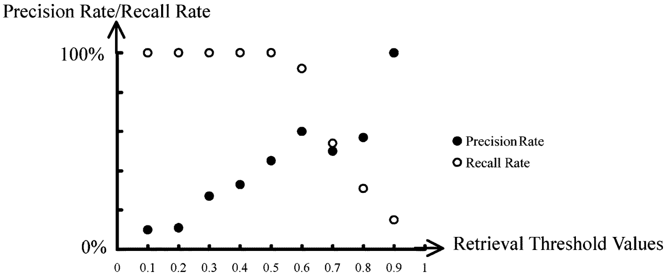

Fig. 4. Precision rate and the recall rate with respect to the original user's query q_1 for different query threshold values.

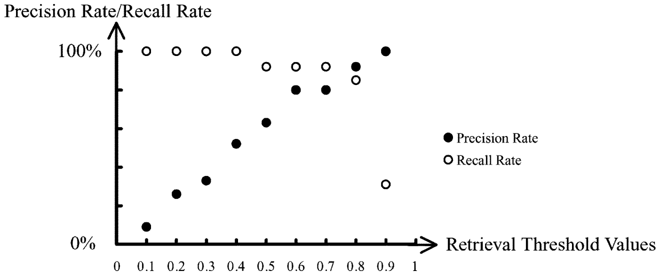

Fig. 5. Precision rate and the recall rate with respect to the modified user's query q_1^+ for different retrieval threshold values.

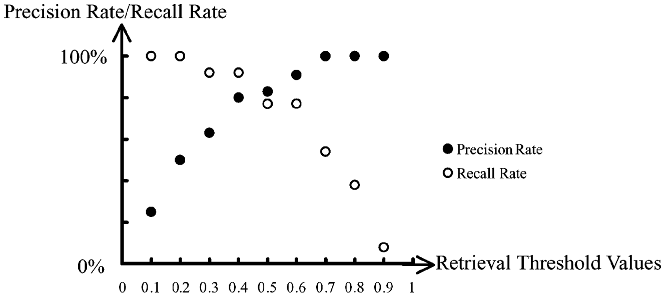

Fig. 6. Precision rate and the recall rate with respect to the modified user's query q_1^* for different retrieval threshold values.

terms "natural," "language," and "processing" as q_1 (i.e., 0.9, 0.9, and 0.8) and one additional term "word" with the weight 0.46. By applying the proposed query modification method, the original user's query q_1 can be modified into query q_1^* which also has the same weights of the terms "natural," "language," and "processing" as q_1 (i.e., 0.9, 0.9 and 0.8) and four additional terms "word," "dictionari" (root of "dictionary"), "corpu" (root of "corpus"), and "speech" with the weights 0.46, 0.44, 0.39, and 0.37, respectively. Based on the queries q_1, q_1^+ and q_1^* for document retrieval, the precision rates and the recall rates with respect to these three queries are shown in Table VI, where the retrieval threshold value θ is given by the user and $0 \leq \theta \leq 1$. If a document d_i wants be retrieved, then the retrieval status value $RSV(d_i)$ of document d_i should be larger than or equal to the "retrieval threshold value" given by the user. The curves of the precision rate and the recall rate with respect to the user's queries q_1, q_1^+ and q_1^* are shown in Figs. 4–6, respectively.

From Table VI and Figs. 4–6, we can see that the precision rates with respect to the modified user's query q_1^* are larger than the ones with respect to the original user's query q_1 and the modified user's query q_1^+ for each query threshold value.

In the following, we compare the precision rate and the recall rate for the top p retrieved documents with respect to the original user's query q_1, the modified user's query q_1^+ and the modified user's query q_1^*, respectively, where p is a positive integer. In our experiment, we let the query threshold value be 0.2. Then, 116 documents, 55 documents, and 25 documents are retrieved with respect to the original user's query q_1, the modified user's query q_1^+ and the modified user's query q_1^*, respectively. Let us consider the following cases, remembering that there are 13 relevant documents in the collection.

Case 1) When $p \leq 10$, both the precision rate and the recall rate with respect to the modified user's query q_1^* are the largest. For example, when $p = 10$, the original user's query q_1 gets five relevant documents among the ten retrieved documents. In this case, we can see that the precision rate is 50% and the recall rate is 38%. The modified user's query q_1^+ gets nine relevant documents among the ten retrieved documents. In this case, the precision rate is 90% and the recall rate is 69%. The modified user's query q_1^* gets ten relevant documents among the ten retrieved documents. In this case, we can see that the precision rate is 100% and the recall rate is 77%.

TABLE VII
NUMBER OF RELEVANT DOCUMENTS, THE PRECISION RATES AND THE RECALL RATES WITH RESPECT TO THE RETRIEVED TOP p DOCUMENTS OF THE USER'S QUERY q_1, q_1^+ AND q_1^*, RESPECTIVELY, WHERE THE RETRIEVAL THRESHOLD VALUE IS 0.2

p	The Original User's Query q_1			The Modified User's Query $q_1{}^+$ Based on [20]			The Modified User's Query q_1* Based on the Proposed Method		
	Number of Relevant Documents	Precision Rate	Recall Rate	Number of Relevant Documents	Precision Rate	Recall Rate	Number of Relevant Documents	Precision Rate	Recall Rate
5	4	80%	31%	4	80%	31%	5	100%	38%
10	5	50%	38%	9	90%	69%	10	100%	77%
15	8	53%	62%	12	80%	92%	12	80%	92%
20	12	60%	92%	12	60%	92%	12	60%	92%
25	13	52%	100%	13	52%	100%	13	52%	100%

Case 2) When $10 < p \leq 19$, the precision rate and the recall rate of the modified user's query q_1^* are equal to the precision rate and the recall rate of the modified user's query q_1^+, respectively, and are larger than the precision rate and the recall rate of the original user's query q_1, respectively. For example, when $p = 15$, the original user's query q_1 gets eight relevant documents among the 15 retrieved documents. In this case, the precision rate is 53% and the recall rate is 62%. Both the modified user's query q_1^+ and the modified user's query q_1^* get 12 relevant documents among the 15 retrieved documents. In this case, the precision rate is 80% and the recall rate is 92%.

Case 3) When $19 < p \leq 25$, the precision rate and the recall rate of the modified user's query q_1^* are equal to the precision rate and the recall rate of the modified user's query q_1^+, respectively, and are equal to the precision rate and the recall rate of the original user's query q_1, respectively. For example, when $p = 20$, the original user's query q_1, the modified user's query q_1^+ and the modified user's query q_1^* all get 12 relevant documents among the 20 retrieved documents. In this case, the precision rate is 60% and the recall rate is 92%.

The number of relevant documents, the precision rates and the recall rates with respect to the retrieved top p documents of the user's queries q_1, q_1^+, and q_1^*, respectively, are summarized in Table VII, where the query threshold value is 0.2.

From Table VII, we can see that the retrieval results with respect to the modified user's query q_1^* based on the proposed query modification method are better than those with respect to the user's queries q_1 and q_1^+ from the point of view that most of the top ranked documents are relevant.

VII. CONCLUSION

In this paper, we have extended the work of Kraft *et al.* [20] to present a new method for fuzzy information retrieval based on fuzzy hierarchical clustering and fuzzy inference techniques. We have presented a fuzzy agglomerative hierarchical clustering algorithm for clustering documents and to get the document cluster center of each document cluster. We also have presented a method to construct fuzzy logic rules based on the document clusters and their document cluster centers. We also have applied the constructed fuzzy logic rules to modify a user's query for query expansion and to guide the information retrieval system to retrieve documents relevant to the user's request. The proposed fuzzy information retrieval method is more flexible and more intelligent than the existing methods due to the fact that it can expand users' queries for fuzzy information retrieval in a more effective manner.

REFERENCES

[1] J. C. Bezdek, R. J. Hathaway, M. J. Sabin, and W. T. Tucker, "Convergence theory for fuzzy c-means: Counterexamples and repairs," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-17, no. 5, pp. 873–877, Sep.-Oct. 1987.

[2] G. Bordogna and G. Pasi, "A user-adaptive neural network supporting a rule-based relevance feedback," *Fuzzy Sets Syst.*, vol. 82, no. 2, pp. 201–211, 1996.

[3] C. L. P. Chen and Y. Lu, "FUZZ: A fuzzy-based concept formation system that integrates human categorization and numerical clustering," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 27, no. 1, pp. 79–94, Feb. 1997.

[4] J. Chen and S. Kundu, "A sound and complete fuzzy logic system using Zadeh's implication operator," in *Proc. 9th Int. Symp. Methodologies for Intelligent Systems*, Zakopane, Poland, 1996, pp. 233–242.

[5] S. M. Chen and Y. J. Horng, "Fuzzy query processing for document retrieval based on extended fuzzy concept networks," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 29, no. 1, pp. 126–135, Feb. 1999.

[6] S. M. Chen, Y. J. Horng, and C. H. Lee, "Document retrieval using fuzzy-valued concept networks," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 31, no. 1, pp. 111–118, Feb. 2001.

[7] S. M. Chen, W. H. Hsiao, and Y. J. Horng, "A knowledge-based method for fuzzy query processing for document retrieval," *Cybern. Syst.: Int. J.*, vol. 28, no. 1, pp. 99–119, 1997.

[8] S. M. Chen and J. Y. Wang, "Document retrieval using knowledge-based fuzzy information retrieval techniques," *IEEE Trans. Syst., Man, Cybern.*, vol. 25, no. 5, pp. 793–803, May 1995.

[9] S. M. Chen, Y. J. Horng, and C. H. Lee, "Fuzzy information retrieval based on multi-relationship fuzzy concept networks," *Fuzzy Sets Syst.*, vol. 140, no. 1, pp. 183–205, 2003.

[10] W. B. Frakes, "Stemming algorithms," in *Information Retrieval: Data Structure & Algorithms*, W. B. Frakes and R. Baeza-Yates, Eds. Upper Saddle River, NJ: Prentice-Hall, 1992.

[11] Y. J. Horng and S. M. Chen, "Fuzzy query processing for document retrieval based on extended fuzzy concept networks," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 29, no. 1, pp. 96–104, Feb. 1999.

[12] ——, "Finding inheritance hierarchies in fuzzy-valued concept-networks," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 29, no. 1, pp. 126–135, Feb. 1999.

[13] Y. J. Horng, S. M. Chen, and C. H. Lee, "A new fuzzy information retrieval method based on document terms reweighting techniques," *Int. J. Inf. Manage. Sci.*, vol. 14, no. 4, pp. 63–82, 2003.

[14] ——, "Automatically constructing multi-relationship fuzzy concept networks for document retrieval," *Appl. Art. Intell.: Int. J.*, vol. 17, no. 4, pp. 303–328, 2003.

[15] ——, "Fuzzy information retrieval using fuzzy hierarchical clustering and fuzzy inference techniques," in *Proc. 13th Int. Conf. Information Management*, vol. 1, Taipei, Taiwan, R.O.C., 2002, pp. 215–222.

[16] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: A review," *ACM Comput. Surveys*, vol. 31, no. 3, pp. 264–323, 1999.

[17] Y. Jung, H. Park, and D. Du, "An effective term weighting scheme for information retrieval," Univ. Minnesota, Dept. Comput. Sci., Minneapolis, MN, Comput. Sci. Tech. Rep. TR008, 2000.

[18] K. J. Kim and S. B. Cho, "A personalized web search engine using fuzzy concept network with link structure," in *Proc. Joint 9th IFSA Congr. 20th NAFIPS Int. Conf.*, Vancouver, BC, Canada, 2001, pp. 81–86.

[19] B. M. Kim, J. Y. Kim, and J. Kim, "Query term expansion and reweighting using term co-occurrence similarity and fuzzy inference," in *Proc. Joint 9th IFSA World Congr. 20th NAFIPS Int. Conf.*, vol. 2, Vancouver, BC, Canada, 2001, pp. 715–720.

[20] D. H. Kraft, J. Chen, and A. Mikulcic, "Combining fuzzy clustering and fuzzy inferencing in information retrieval," in *Proc. 9th IEEE Int. Conf. Fuzzy Systems*, vol. 1, San Antonio, TX, 2000, pp. 375–380.

[21] H. M. Lee, S. K. Lin, and C. W. Huang, "Interactive query expansion based on fuzzy association thesaurus for web information retrieval," in *Proc. 10th IEEE Int. Conf. Fuzzy Systems*, vol. 2, Melbourne, Australia, 2001.

[22] S. Miyamoto, "Information retrieval based on fuzzy associations," *Fuzzy Sets Syst.*, vol. 38, no. 2, pp. 191–205, 1990.

[23] E. Rasmussen, "Clustering algorithms," in *Information Retrieval: Data Structure and Algorithms*, W. B. Frakes and R. Baeza-Yates, Eds. Upper Saddle River, NJ: Prentice-Hall, 1992, pp. 419–442.

[24] F. J. Rohlf, "Single-link clustering algorithms," in *Classification, Pattern Recognition, and Reduction of Dimensionality*, P. R. Krishnaiah and J. N. Kanal, Eds. Amsterdam, The Netherlands: North-Holland, pp. 267–284.

[25] G. Salton, *The SMART Retrieval System: Experiments in Automatic Document Processing*. Upper Saddle River, NJ: Prentice-Hall, 1971.

[26] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Inf. Process. Manage.*, vol. 24, no. 5, pp. 513–523, 1988.

[27] G. Salton and M. J. Mcgill, *Introduction to Modern Information Retrieval*. New York: McGraw-Hill, 1983.

[28] R. Sibson, "SLINK: An optimally efficient algorithm for the single link cluster method," *Comput. J.*, vol. 16, pp. 30–34, 1973.

[29] L. A. Zadeh, "Fuzzy sets," *Inf. Control*, vol. 8, pp. 338–353, 1965.

[30] A subset of the collection of the research reports of the national science council. NTUST, Taiwan, R.O.C.. [Online]. Available: http://fuzzylab.et.ntust.edu.tw/NSC_Report_Database/247documents.html

Yih-Jen Horng received the B.S., M.S., and Ph.D. degrees, all in computer and information science, from National Chiao Tung University, Hsinchu, Taiwan, R.O.C., in 1994, 1996, 2003, respectively.

His current research interests include fuzzy systems, database systems, and artificial intelligence.

Dr. Horng was the winner of the 1996 Acer Dragon Outstanding M.S. Thesis Award, Republic of China. He was also the winner of the 2003 Acer Dragon Outstanding Ph.D. Dissertation Award, Republic of China.

Shyi-Ming Chen (M'88–SM'96) received the B.S. degree, and the M.S., and Ph.D. degrees in electrical engineering from the National Taiwan University of Science and Technology, Taipei, Taiwan, in 1982, 1986, and 1991, respectively.

From August 1987 to July 1989, and from August 1990 to July 1991, he was with the Department of Electronic Engineering, Fu-Jen University, Taipei, Taiwan. From August 1991 to July 1996, he was an Associate Professor in the Department of Computer and Information Science, National Chiao Tung University, Hsinchu, Taiwan. From August 1996 to July 1998, he was a Professor in the Department of Computer and Information Science, National Chiao Tung University, Hsinchu, Taiwan. From August 1998 to July 2001, he was a Professor in the Department of Electronic Engineering, National Taiwan University of Science and Technology, Taipei, Taiwan. Since August 2001, he has been a Professor in the Department of Computer Science and Information Engineering, National Taiwan University of Science and Technology. He was a Visiting Scholar in the Department of Electrical Engineering and Computer Science, University of California, Berkeley, in 1999, and with the Institute of Information Science, Academia Sinica, Republic of China, in 2003. His current research interests include fuzzy systems, database systems, knowledge-based systems, artificial intelligence, data mining, and genetic algorithms. He has published more than 200 papers in referred journals, book chapters, and conference proceedings.

Dr. Chen was the winner of the 1994 Outstanding Paper Award of the *Journal of Information and Education* and the winner of the 1995 Outstanding Paper Award of the Computer Society of the Republic of China. He was the winner of the 1997 Outstanding Youth Electrical Engineer Award of the Chinese Institute of Electrical Engineering, Republic of China. He was the winner of the Best Paper Award of the 1999 National Computer Symposium, Republic of China. He was the winner of the 1999 Outstanding Paper Award of the Computer Society of the Republic of China. He was the winner of the 2001 *Outstanding Talented Person Award*, Republic of China, for the contributions in Information Technology. He was the winner of the *Outstanding Electrical Engineering Professor Award* granted by the *Chinese Institute of Electrical Engineering* (CIEE), Republic of China, in 2002. He was the winner of the 2003 Outstanding Paper Award of the Technological and Vocational Education Society, Republic of China. He is a Member of the ACM, the International Fuzzy Systems Association (IFSA), and the Phi Tau Phi Scholastic Honor Society. He is currently the President of the Taiwanese Association for Artificial Intelligence (TAAI). He is also an Executive Committee Member of the Chinese Fuzzy Systems Association (CFSA). He is an Editor of the *Journal of the Chinese Grey System Association*, an Associate Editor of the *International Journal of New Mathematics and Natural Computation*, and an Associate Editor of the *International Journal of Fuzzy Systems*. He is listed in *International Who's Who of Professionals*, *Marquis Who's Who in the World*, and *Marquis Who's Who in Science and Engineering*.

Yu-Chuan Chang received the B.S. degree from the Department of Computer Science and Information Engineering, Fu-Jen University, Taipei, Taiwan, R.O.C., in 2000, and the M.S. degree in computer science and information engineering from National Taiwan University of Science and Technology, Taipei, Taiwan, R.O.C., in 2004. He is currently working toward the Ph.D. degree in computer science and information engineering at National Taiwan University of Science and Technology.

His current research interests include fuzzy information systems and artificial intelligence.

Chia-Hoang Lee received the Ph.D. degree in computer science from the University of Maryland, College Park, in 1983.

From 1984 to 1985, he was with the Department of Mathematics and Computer Science, University of Maryland. From 1985 to 1992, he was with the Department of Computer Science, Purdue University, West Lafayette, IN. He is currently a Professor in the Department of Computer and Information Science and also serves as the Deputy Director of MediaTek research center at National Chiao Tung University, Hsinchu, Taiwan, R.O.C. His current research interests include artificial intelligence, man machine interface systems, and natural language processing. He was an Associate Editor of the *International Journal of Pattern Recognition*.