Merging Textual Knowledge Represented by Element Fuzzy Cognitive Maps

Xiangfeng Luo
School of Computer Engineering and Science, Shanghai University, Shanghai, China
Email: luoxf@shu.edu.cn

Jun Zhang, Fangfang Liu,Yi Du, Zhian Yu and Weimin Xu
School of Computer Engineering and Science, Shanghai University, Shanghai, China
Email: { JunZhang1986.shu@gmail.com, ffliu@shu.edu.cn, cyrisdu@163.com , andyfern@163.com and wmxu@staff.shu.edu.cn}

***Abstract*—Importance degree and difference degree of keywords in different topics have been measured by the associated weights in Element Fuzzy Cognitive Maps (E-FCMs) which can represent textual knowledge effectively. Logic "and" operation is introduced to roughly evaluate the similarities between the mass E-FCMs in order to form the similar sets of textual knowledge. Based on the associated weight measuring and the logic operation, an E-FCMs-based knowledge merging algorithm is proposed to inspect the noisy and the redundancy information hidden in the original E-FCMs belonging to one similar set. A formula obtained through F-measure is employed as an indicator to measure the loss of textual information during the merging process of E-FCMs. The merging algorithm and the indicator provide a concise representation of textual knowledge that can be used in understanding-based automatic text classification and clustering, as well as relevant knowledge aggregation and integration. The proposed algorithm will have very good application prospects in future.**

***Index Terms*—E-FCMs; knowledge merging; knowledge representation**

I. INTRODUCTION

Concise representation of textual knowledge is one of the key issues of automatic text classification and clustering, relevant knowledge aggregation and integration in e-Science, scientific workflow, and e-Learning systems. It can highlight the relations between textual topics and keywords, and also can reduce the algorithm complexity of text analysis effectively. However, similar topics and same topics may be hidden in different information sources (e.g. one topic may be discussed in different documents), which leads to existence of mass noisy and redundant information in the original textual knowledge representations. Therefore, the problem is how to obtain the concise representations of texts. One choice is to merge the noisy and the redundant information hidden in the numerous original textual representations, which needs to maintain the maximum textual knowledge and highlight the relations between textual topics and keywords as far as possible.

Many methods of knowledge representations have been proposed. For example, vector space model [1], ontology-based knowledge representations (e.g. OIL, OWL and SHOE), probabilistic latent semantic analysis [2], latent dirichlet allocation [3], author-topic model [4], author-recipient- topic model[5], correlated topic model [6], symbolic logic model [7] and element fuzzy cognitive maps (E-FCMs) models [9-10], etc.

E-FCMs-based knowledge representation has been proposed by Zhuge and Luo [9-10], which has better capabilities of composition and decomposition that are indispensable to merging the same or similar textual knowledge. But the details of merging process of textual knowledge have not been discussed in [9-10].

Traditional techniques for information merging/ integration include knowledge based merging [11-14], implicit/explicit priorities and argumentation framework based merging [13], possibility and distance based merging [15], belief based merging [12, 16-17], etc. But the current methods of knowledge merging mainly focus on the knowledge that exist complements, dispositional /epistemic conflicts and ontological conflicts, which rarely consider the merging of knowledge represented by E-FCMs with mass noisy and redundant information. So, we particularly focus on this question in this paper.

In addition, because of the large scale of textual information, traditional techniques for information merging/integration aforementioned are not applicable to dealing with massive data. However, it is an inevitable trend that the textual information to be handled will get extremely large. C. Xiao, etc [8] proposed an algorithm called *ppjoin*, which combines positional filtering with the prefix filtering-based algorithm, to dramatically reduce the candidate sizes and hence to improve the efficiency when we meet massive data. This algorithm reduces the complexity of measuring similarity including *jaccard*, *cosine* and *overlap* similarity [8] so as to improve the performance of textual knowledge merging.

Although some related work has been done on the merging of Fuzzy Cognitive Map (FCMs) and Cognitive Maps (CMs), they focus on how to solve the conflict

Manuscript received March 31, 2009; revised July 18, 2009; accepted Sep 15, 2009.

doi:10.4304/jsw.5.2.225-234

knowledge while an edge is inconsistent with other FCMs/CMs [18-19].

The rest of this paper is organized as follows. Section 2 describes basic terms including E-FCMs, etc. Section 3 introduces our main algorithm used for textual knowledge merging, which can be divided into two steps. Section 4 shows our experiment results.

II. BASIC TERMS

Term 1: (Element Concept, C_i)

Element concept is a concept of FCM expressed by textual keyword.

The associated weight of the relation from element concept C_i to C_j is denoted by w_{ij}.

Term 2: (Theme Concept, C_j^0)

Theme concept is a concept of FCM using phrases or a short sentence to clearly represent the implied semantic information generated by the co-occurrence keywords (i.e. element concepts) appearing in paragraphs or a section.

Theme concept can be expressed by the title of textual fragment (e.g. paragraphs), which is denoted by C_j^0.

In a FCM, the associated weight of the relation from element concept C_i to theme concept C_j^0 is denoted by w_{ij}^0.

Term 3: (Element Fuzzy Cognitive Map, E-FCM)

Element Fuzzy Cognitive Map (E-FCM) [9-10] is a fuzzy cognitive map, whose element concepts are represented by keywords; state values of element concepts are computed by the function of keyword's frequency, position and font size in paragraphs or a section; theme concept is represented by the implied semantics of co-occurrence keywords appearing in paragraphs or a section; the definition of the relations and their associated weights are the same as in FCM.

For example, Figure 1 is an E-FCM generated by the algorithm in [9]. Semantic information of co-occurrence keywords expressed by E-FCM is more abundant than a set of separate keywords because E-FCM stores topic information instead of separate keywords.

Term 4 :(Equivalent Class)

If E-FCM$_n$ is similar to E-FCM$_m$ and E-FCM$_m$ is also similar to E-FCM$_n$ under a similarity in a similar set, we say that E-FCM$_n$ and E-FCM$_m$ belong to one equivalent class.

Figure 2 shows an example of two E-FCMs belonging to one equivalent class. In this figure, EQ (n, m) is the similarity of E-FCM$_n$ with E-FCM$_m$. ξ is a threshold of the similarity.

III. MERGING PROCESS OF E-FCMS

A. Main Steps of Merging E-FCMs

There are large numbers of same and similar topics hidden in texts, and authors have different habits and styles to describe one topic. Therefore, the original E-FCMs generated by the algorithm in [9] remain mass noisy and redundant information, which make it impractical for text analysis. To remove the noise and the redundancy information, the E-FCMs generated using the algorithm in [9] need to be merged. The main steps of merging E-FCMs are as follows.

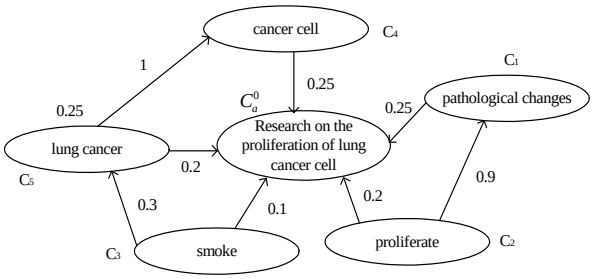


Figure 1.The topic of "*proliferation of lung cancer cell*" represented by E-FCM (denoted as E-FCM$_1$)

(1) Find out similar sets of E-FCMs in which E-FCMs have similar relationships measured by a method of roughly calculating the similarities between E-FCMs;

(2) With the results of step (1), precisely evaluate the similarities between E-FCMs considering the importance degree and the difference degree of keywords in different topics;

(3) Find equivalent classes based on the similarities calculated in previous steps;

(4) Merge E-FCMs belonging to the same equivalent class.

In these steps, 1) calculating of similarities between E-FCMs; and 2) finding and merging of equivalent classes in a similar set are the major steps of merging textual knowledge represented by E-FCM.

B. Method of Roughly Calculating the Similarities between E-FCMs

Vast topics exist in Web resources and each topic may appear in different resources. There are also mass E-FCMs representing topics in e-Science, and e-Learning systems. If we directly deal with the numberless topics represented by E-FCMs, it would lead to the high computational complexity in the computing of similarities between the mass E-FCMs. Therefore, we must roughly evaluate the similarities in order to form similar sets to decrease the number of E-FCMs drastically. On the other hand, in order to reduce the computation space and time, we introduce logic "and" operation to measure the similarities in a similar set. Herein, we ignore the importance degree and the difference degree of keywords in different topics, which will be discussed in the next sections. Further more, we set a threshold n, which means the percentage of reservation, in order to balance the elements belonging to each similar set. The main steps of the logic "and" operation are as follows.

(1) All the E-FCMs are stored in a high-dimensional sparse matrix, if the h^{th} element concept appear in the

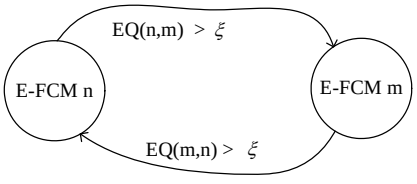


Figure 2.E-FCM$_m$ and E-FCM$_n$ belong to one Equivalent Class

i^{th} E-FCM, the value of the element (*i*,*h*) in the sparse matrix is 1, otherwise 0;

(2) Create the logic sparse matrix of the i^{th} E-FCM by computing with each row in this matrix using the logic "and" operation.

(3) Sum all the values in one row of the logic sparse matrix and *v*(*i*,*j*) is obtained, where *v*(*i*,*j*) is the sum of the j^{th} row using logic "and" operation with the i^{th} E-FCM;

(4) Calculate *g*(*i*,*j*)= *v*(*i*,*j*)/*K*(*i*), where *K*(*i*) is the number of the i^{th} E-FCM's element concepts with values nonzero;

(5) For each *g*(*i*,*j*), if it is bigger than threshold *m* ($m \in [0,1]$) and neither the i^{th} or the j^{th} E-FCM belongs to any similar set, then we add the j^{th} E-FCM to the i^{th} E-FCM's similar candidate set; otherwise go to step (7);

(6) Reserve the E-FCMs as the i^{th} E-FCM's similar set, which is ranked before *n* percent of the i^{th} E-FCM's similar candidate set by sorting the E-FCMs in descending order based on *g*(*i*,*j*); goto step (5);

(7) All the similar sets have been generated, end.

Through above steps, the mass E-FCMs can be divided into different similar sets quickly because all the computing is based on the logic "and" operations. In the following, we discuss the merging of E-FCMs in one similar set, which reduces the number of E-FCMs drastically.

C. Methods of Precisely Evaluating the Similarities between E-FCMs within one Similar Set

Element concepts in different E-FCMs have different importance degrees. The importance degrees of element concepts in different E-FCMs are reflected by the associated weights from co-occurrence element concepts to theme concept as well as the associated weights from one element concept to other element concepts. So the methods of precisely evaluating the similarities between E-FCMs belonging to one similar set should reflect the following factors.

(1) The associated weights from co-occurrence element concepts to theme concept in different E-FCMs.

(2) The associated weights between co-occurrence element concepts in different E-FCMs.

To better understand these two factors, we give two E-FCMs for studying the measuring of equivalent classes. Figure 1 is an E-FCM (denoted as E-FCM$_1$) representing the topic of "*proliferation of lung cancer cell*". Figure 3 is another topic "*cancer research*" represented by E-FCM (denoted as E-FCM$_2$).

a. Associated weights from co-occurrence concepts to theme concept

The same concept in different E-FCMs has different meanings. As a result, not only the number of co-occurrence concepts should be taken into account, but also the associated weights from co-occurrence concepts to theme concept when calculating the similarity between E-FCMs precisely.

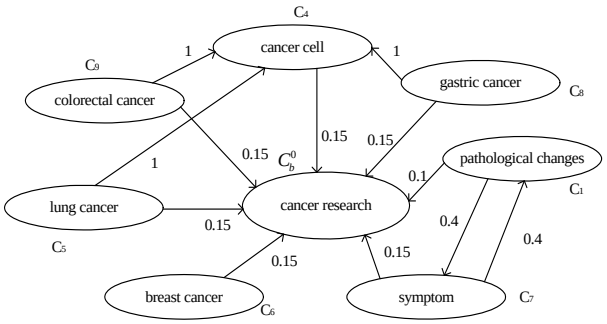


Figure 3.The topic of "*cancer research*" represented by E-FCM (denoted as E-FCM$_2$)

To precisely calculate the associated weights from co-occurrence element concepts to theme concept, the following factors should be reflected.

(1) The average of the difference degrees from co-occurrence element concepts to theme concepts in E-FCM$_m$ and E-FCM$_n$ should be reflected. So, $\sum_i^s \left| w_i^m - w_i^n \right| / 2$ is obtained to reflect the difference degree of E-FCM$_m$ and E-FCM$_n$. Herein, w_i^m and w_i^n represent the associated weights from co-occurrence element concept C_i to theme concepts C_m^0 and C_n^0, respectively.

$\sum_i^s \left| w_i^m - w_i^n \right| / 2$ represents the difference degree of C_i in E-FCM$_m$ and E-FCM$_n$; *s* is the common element concepts in E-FCM$_m$ and E-FCM$_n$.

For example, the associated weights from co-occurrence element concepts "*cancel cell*", "*pathological changes*", and "*lung cancer*" to the theme concepts are 0.25, 0.25, 0.2 in E-FCM$_1$, and 0.15, 0.1, 0.15 in E-FCM$_2$, respectively; so the difference degree between E-FCM$_1$ and E-FCM$_2$ is 0.15.

(2) The average of the sum of the associated weights from co-occurrence element concepts to theme concepts in E-FCM$_m$ and E-FCM$_n$ should be reflected. So, $\sum_i^s (w_i^m + w_i^n) / 2$ is obtained to reflect the importance degree between E-FCM$_m$ and E-FCM$_n$, in which $(w^m{}_i + w^n{}_i)/2$ is the importance degree of C_i between E-FCM$_m$ and E-FCM$_n$.

For example, the associated weights from co-occurrence element concepts "*cancel cell*", "*pathological changes*", and "*lung cancer*" to theme concepts are 0.25, 0.25, 0.2 in E-FCM$_1$, and 0.15, 0.1, 0.15 in E-FCM$_2$, respectively; so the importance degree between E-FCM$_2$ and E-FCM$_1$ is 0.55.

(3) The sum of the associated weights from co-occurrence element concepts to theme concepts in each E-FCM should be reflected. $(\sum_i^s w^m{}_i)$ and $(\sum_i^s w^n{}_i)$ $(\sum_i^s w^n{}_i)$ is obtained to reflect the impact, which is made by co-occurrence element concepts, on the computing of similarity between E-FCMs.

For example, the associated weights from the co-occurrence element concepts to the theme concepts are 0.25, 0.2, 0.25 in E-FCM$_1$ and 0.15, 0.15, 0.1 in E-FCM$_2$, so $\sum_{i=1}^{3} w_i^1$ is 0.7 and $\sum_{i=1}^{3} w_i^2$ is 0.4, respectively.

Taking the factors aforementioned into account, we propose three strategies to evaluate the similarities between E-FCMs. All of the three strategies are constructed based on the idea that each co-occurrence concept, which is very important for the similarity calculating between E-FCMs, should have a lower difference degree and a higher importance degree between E-FCMs. Strategy 1 adopts a simple arithmetic operation that subtracting difference degree from importance degree to reflect the contribution of each co-occurrence element concept to precisely calculate the similarity between E-FCMs. Strategy 2 employs a function $f(x)=\frac{e^x-e^{-x}}{e^x+e^{-x}}$ to improve on strategy 1. $f(x)=\frac{e^x-e^{-x}}{e^x+e^{-x}}$ is a monotonically increasing function that values from 0 to 1 when x is positive. Strategy 3 calculates the similarity through multiplying importance degree by difference degree, which is embedded in a monotonically function $f(x)=e^{-x}$. Through experiment results, we will select one of the three strategies to implement the merging process of textual knowledge.

Strategy 1:

$$EQ1(m,n)=\sqrt{\left(\sum_i^s w_i^m\right)*\left(\sum_i^s \frac{\left(w_i^m+w_i^n-\left|w_i^m-w_i^n\right|\right)}{2}\right)} \quad (1)$$

where s is the number of co-occurrence element concepts.

For example, $EQ1(1,2)=\sqrt{0.7*0.4}=0.529$, $EQ1(2,1)=\sqrt{0.4*0.4}=0.40$, which correspond to E-FCM$_1$ and E-FCM$_2$ shown in Figure 1 and Figure 3, respectively.

Strategy 2:

$$EQ1(m,n)=\sqrt{\left(\sum_i^s f\left(w_i^m\right)\right)*\left(\sum_i^s\left(f\left(\frac{w_i^m+w_i^n}{2}\right)-f\left(\frac{\left|w_i^m-w_i^n\right|}{2}\right)\right)\right)} \quad (2)$$

where $f(x)=\frac{e^x-e^{-x}}{e^x+e^{-x}}$ and s is the number of co-occurrence element concepts.

For example, $EQ1(1,2)=\sqrt{(0.6872)*(0.3746)}=0.5074$, $EQ1(2,1)=\sqrt{(0.3974)*(0.5074)}=0.3859$, which correspond to E-FCM$_1$ and E-FCM$_2$ shown in Figure 1 and Figure 3, respectively.

Strategy 3:

$$EQ1(m,n)=\sqrt{\left(\sum_i^s w_i^m\right)*\left(\sum_i^s\left(\frac{w_i^m+w_i^n}{2}\right)*\left(e^{-\frac{\left|w_i^m-w_i^n\right|}{2}}\right)\right)} \quad (3)$$

where s is the number of co-occurrence element concepts.

For example, $EQ1(1,2)=\sqrt{(0.7)*(0.5042)}=0.5941$, $EQ1(2,1)=\sqrt{(0.4)*(0.5042)}=0.4491$, which correspond to E-FCM$_1$ and E-FCM$_2$ shown in Figure 1 and Figure 3, respectively.

b. Associated weights between co-occurrence concepts

For each E-FCM there exits relations between concepts. The associated weights between two same concepts vary among E-FCMs. Therefore, despite the number of co-occurrence concepts and the associated weights from co-occurrence concepts to theme concept, associated weights between co-occurrence should also be considered when precisely calculating the similarity.

For precisely evaluating the associated weights between co-occurrence element concepts, the following factors should be reflected.

(1)The average of the associated weights from the co-occurrence element concept C_i to C_j in E-FCM$_m$ and E-FCM$_n$ should be reflected. So, $\left|w^m{}_{ij}-w^n{}_{ij}\right|/2$ is obtained, which reflects the difference degree of the edge from C_i to C_j in E-FCM$_m$ and E-FCM$_n$. Herein, $w^m{}_{ij}$ and $w^n{}_{ij}$ represent the associated weights from co-occurrence element concept C_i to C_j in E-FCM$_m$ and E-FCM$_n$, respectively.

For example, the associated weight from co-occurrence element concept "*lung cancer*" to "*cancel cell*" in E-FCM$_1$ is 1, and it is 1 in E-FCM$_2$ too, so the difference degree of the edge from C_i to C_j in E-FCM$_1$ and E-FCM$_2$ is 0.

(2)The average of the sum of the associated weights that are from co-occurrence element concept C_i to C_j in E-FCM$_m$ and E-FCM$_n$ should be reflected. So, $(w^m{}_{ij}+w^n{}_{ij})/2$ is obtained, which reflects the importance degree of the edge from C_i to C_j in E-FCM$_m$ and E-FCM$_n$.

For example, the associated weight from co-occurrence element concepts "*lung cancer*" to "*cancel cell*" in E-FCM$_1$ is 1, and it is 1 in E-FCM$_2$ too, so the importance degree of the edge from "*lung cancer*" to "*cancel cell*" in E-FCM$_1$ and E-FCM$_2$ is 1.

According to (1)-(2), we also propose three strategies to evaluate the further similarities between E-FCMs. Similarly, all of the three strategies should have the ability that strengthening the importance degree and weakening the difference degree when calculating the similarity between E-FCMs. Moreover, the following three strategies have similar forms to the strategies described by last section.

Strategy 1:

$$EQ2(m,n)=\sqrt{\left(\sum_{ij}^s\left(\frac{\left(w_{ij}^m+w_{ij}^n\right)-\left|w_{ij}^m-w_{ij}^n\right|}{2}\right)\right)\Big/ r^m} \quad (4)$$

where $w^m{}_{ij}$ is the associated weight from C_i to C_j in E-FCM$_m$. $w^n{}_{ij}$ is the associated weight from C_i to C_j in E-FCM$_n$. r^m is the number of associated relations existing in E-FCM$_m$'s element concepts; s is the number of the co-occurrence element concepts between E-FCM$_m$ and E-FCM$_n$.

Strategy 2:

$$EQ2(m,n)=\sqrt{\left(\sum_{ij}^{s}\left(f\left(\frac{w_{ij}^{m}+w_{ij}^{n}}{2}\right)-f\left(\frac{\left|w_{ij}^{m}-w_{ij}^{n}\right|}{2}\right)\right)\right)\Big/ r^{m}} \quad (5)$$

where $f(x)=\frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$, w_{ij}^{m} , w_{ij}^{n} , s and r^{m} have the same meaning to strategy 1.

Strategy 3:

$$EQ2(m,n)=\sqrt{\left(\sum_{ij}^{s}\left(\left(\frac{w_{ij}^{m}+w_{ij}^{n}}{2}\right)*e^{-\frac{\left|w_{ij}^{m}-w_{ij}^{n}\right|}{2}}\right)\right)\Big/ r^{m}} \quad (6)$$

where w_{ij}^{m} , w_{ij}^{n} , s and r^{m} have the same meaning to strategy 1.

Then the final precisely evaluating the similarities between E-FCM$_m$ and E-FCM$_n$ is determined by

$$EQ(m,n)=\alpha EQ1(m,n)+(1-\alpha)EQ2(m,n) \quad (7)$$

where $\alpha\in[0,1]$.

Formula (7) reflects the similarity between E-FCMs.

D. Generate the Equivalent Classes in Each Similar Set

a. Inspect Noisy and Redundancy Information

If *EQ (n, m)* and *EQ* (*m, n*) are bigger than threshold ξ, E-FCM$_n$ and E-FCM$_m$ belong to one equivalent class.

Term 5:(Weak Equivalent Class between E-FCM$_m$ and E-FCM$_n$)

If $EQ(m,n)<\xi$ and $EQ(n,m)<\xi$, E-FCM$_m$ and E-FCM$_n$ have a low probability to describe a topic or a similar topic. The two E-FCMs are defined as weak equivalent class.

Term 6:(Strong Equivalent Class between E-FCM$_m$ and E-FCM$_n$)

If $EQ(m,n)>=\xi$ and $EQ(n,m)>=\xi$, E-FCM$_m$ and E-FCM$_n$ have a high probability to describe a topic or a similar topic. The two E-FCMs are defined as strong equivalent class.

Term 7: (Weights Matrix of E-FCMs)

If the associated weight from element concept C_i to theme concept C_j^0 or from element concept C_i to C_j is stored in a matrix's element (*i,j*) , this matrix is called weights matrix of E-FCMs.

Term 8:(Equivalent class matrix of E-FCMs)

If the measurement of equivalent class between E-FCM$_i$ and E-FCM$_j$ is stored in the element (*i,j*) of a matrix , this matrix is called equivalent class matrix of E-FCMs.

The weight matrix and the equivalent class matrix of E-FCMs are the high-dimensional sparse matrix.

We know that with the increase of texts, the number of E-FCMs generated by the algorithm in [9] will increase dramatically. So there are combinatorial explosion in the finding process of equivalent classes. Considering that the number of keywords is limited in a specific domain, high-dimensional sparse matrix is proposed to store the associated weights between E-FCMs' concepts, thus the weights matrix of E-FCMs is formed, which greatly reduce the scale of merging times.

The equivalent class matrix is generated by the computing of formula (7). When E-FCM equivalent class matrix is generated, the threshold ξ is required to find equivalent classes. If ξ is great, E-FCMs with strong equivalence are possible to be merged. If the equivalence measurement *EQ* (*n, m*) is above the threshold and *EQ* (*m,n*) is below the threshold, there is no equivalent class between E-FCM$_n$ and E-FCM$_m$, which indicates E-FCM$_n$ and E-FCM$_m$ do not discuss one topic. The algorithm of finding equivalent classes is described as following.

Input: Similar Set SS_i

Output: Set of Equivalent Class SEC_i

For each similar set SS_i:

(1)Precisely evaluate the similarity $EQ(m,n)$ between E-FCM$_m$ and E-FCM$_n$, which are in SS_i;

(2)Generate concise equivalent class matrix, denoted as $CECM_i$;

While the number of max equivalent class is bigger than 1:

(3)Find out the max equivalent class in $CECM_i$;

(4)If the number of max equivalent class is bigger than 1, add max equivalent class to the set of equivalent class SEC_i;

End While;

End For.

b. Threshold of Equivalent Class based on High-Dimensional Sparse Matrix

In the finding step of equivalent class, ξ is an important parameter. If ξ is chosen improperly, the merging process of E-FCMs would be unsatisfied. Although there are common usages and the habits of keywords on a topic in a text; author's writing style may be different. Different authors or article styles show different frequency and position in the usage of keywords. Therefore, if ξ is determined without any change in the merging process, the number of E-FCM's equivalent class would be too few or too abundant, which affects the results of textual knowledge representations significantly. To solve this problem, we use a variable threshold which can be automatically adjusted according to the actual situation based on the similarities. So the merging process of E-FCMs has certain adaptability. Variable threshold formula is as follows.

$$\zeta_i=\begin{cases}Max(w_{i1}^{i},..,w_{ik}^{i})-T\times(Max(w_{i1}^{i},..,w_{ik}^{i})-Min(w_{i1}^{i},..,w_{ik}^{i})), & Max(w_{i1}^{i},..,w_{ik}^{i})\geq\lambda\\ \lambda, & Max(w_{i1}^{i},..,w_{ik}^{i})<\lambda\end{cases} \quad (8)$$

where ζ_i is the threshold of the i^{th} row in a high-dimensional sparse matrix ; *Max* and *Min* means the maximum and minimum of the equivalence measurements in row *i* of equivalent class matrix; λ is a static threshold that ensures ζ_i big enough.

ζ_i can be set in a particular position between the values of maximum to minimum, which can solve the problem that the fixed threshold brings. *T* represents the ratio which is selected according to the overall standard of the similarities in equivalent class matrix. On the other hand, ζ_i also can be set bigger than any w_k^i in the i^{th}

row while w_k^i in the i^{th} row is too small. Therefore, the proposed merging algorithm can merge the textual knowledge representations with different style texts, which really have strong similar relationships between them.

Term 9: (Concise equivalent class matrix of E-FCMs)

Element (i,j) in an equivalent class matrix is set zero while the value of element (i,j) is less than the threshold of the i^{th} row ζ_i. This matrix is called concise equivalent class matrix of E-FCMs.

In the concise equivalent class matrix of E-FCMs, if element (i,j) is bigger than zero, E-FCM$_i$ and E-FCM$_j$ belong to one equivalent class matrix.

E. Merging Algorithm of E-FCMs

With the increase of texts, the number of E-FCMs will be enormous. There are huge noisy and redundancy information hidden in the original E-FCMs. After merging in equivalent class, the number of E-FCMs may be reduced drastically. The merging process of E-FCM$_n$ and E-FCM$_m$ are as follows.

a. The merging of associated weight from co-occurrence element concept to theme concept

If E-FCM$_1$, E-FCM$_2$,…, and E-FCM$_n$ belong to one equivalent class, the new associated weight from C_i to $C_j^{0(new)}$ in the merged E-FCM can be computed by

$$w_{ij}^{0(new')} = \sum_{E-FCM_m \in S_i} w_{im}^{0(E-FCM_m)} \tag{9}$$

where $w_{ij}^{0(new')}$ is the new associated weight from C_i to $C_j^{0(new)}$; $C_j^{0(new)}$ is the theme concept of the merged E-FCMs which may be the theme concept of E-FCM$_1$, E-FCM$_2$, E-FCM$_3$, …, or E-FCM$_n$; $w_{im}^{0(E-FCM_m)}$ is the associated weight from C_i to C_m^0; C_m^0 is the theme concept of E-FCM$_m$. S_i is a set of E-FCMs that contain the element concept C_i.

The normalized associated weight of formula (9) is

$$w_{is}^{0(new'')} = \frac{w_{is}^{0(new')}}{\sum_{i=1}^{q} w_{is}^{0(new')}} \tag{10}$$

where q is the number of element concepts after the merging of E-FCMs; $w_{is}^{0(new'')}$ is the associated weight from element concept C_i to $C_s^{0(new)}$ that is the theme concept of the merged E-FCM.

In the merging process, if q >10, we delete the element concepts ranked after No.10 by sorting the concepts in descending order based on their associated weights because the number of keywords discussing a topic in a section is rarely bigger than 10.

By the analysis of formula (9) and (10), we know that the co-occurrence element concept in the merged E-FCM is highlighted because the associated weight from co-occurrence element concept to theme concept in the merged E-FCM becomes relatively larger than other element concepts without co-appearance in E-FCM$_m$ and E-FCM$_n$. So the common information hidden in E-FCM is enhanced.

b. The merging of associated weights between co-occurrence element concepts

If there are two co-occurrence element concepts that have relation in E-FCM$_1$, E-FCM$_2$, E-FCM$_3$, …, and E-FCM$_n$ that belong to one equivalent class, the merged weight between co-occurrence element concepts is

$$w_{iv}^{new'} = \left(\sum_{E-FCM_m \in S_i} w_{iv}^{E-FCM_m} \right) \Big/ n \tag{11}$$

where $w_{iv}^{new'}$ is the associated weight from C_i to C_v.; $w_{iv}^{E-FCM_m}$ is the associated weight from co-occurrence element concept C_i to C_v in E-FCM$_m$; S_i is a set of E-FCMs that contain the element concept C_i; n is the number of E-FCMs in S_i.

c. Eliminating the redundant information and noise in equivalent class

In the merged E-FCM, if $w_{is}^{0(new'')}$ or $w_{iv}^{new'}$ is small enough, the element concept may be noisy or redundant information which should be removed. So

$$\begin{cases} if\ w_{is}^{0(new'')} \le \overline{w_{is}^{0(new'')}} / n & remove\ C_i\ from\ the\ merged\ E\text{-}FCM \\ if\ w_{iv}^{new'} \le \overline{w_{iv}} / n & delect\ the\ relation \end{cases} \tag{12}$$

where $\overline{w_{is}^{0(new'')}}$ is the mean of the associated weights from element concept C_i to theme concept $C_s^{0(new)}$ in the merged E-FCM; $\overline{w_{iv}}$ is the mean of the associated weights from one element concept to another; n is a coefficient.

After removing the weights which have small values, the remainder weights should be normalized again.

$$w_{is}^{0(new)} = w_{is}^{0(new'')} \Big/ \sum_{i=1}^{n} w_{is}^{0(new'')} \tag{13}$$

where $w_{is}^{0(new)}$ is the associated weight of the merged E-FCM from element concept C_i to $C_s^{0(new)}$.

After the normalization of associated weights, the common information is enhanced. So the noisy and redundant information may be restrained or even be eliminated by the computing of formula (12) to (13).

d. The stop condition of the merging process

The stop condition of the merging process is as follows.

$$\begin{cases} Merging\ \ if\ \ (\exists i)(Set\ of\ Equivalent\ Class\ SEC_i\ is\ not\ empty) \\ Stop\ \ if\ \ (\forall i)(Set\ of\ Equivalent\ Class\ SEC_i\ is\ empty) \end{cases} \tag{14}$$

F. Measure the Loss of Textual Information in the Merging Process of E-FCMs

In the merging process of textual knowledge represented by E-FCM, a part of textual information may be lost accompanied with the removing of the noisy and the redundant information. As a result, it is important to measure the loss of textual information in the merging process of E-FCMs. Y Zhang, etc [20] use precision and

recall to evaluate the effectiveness of their novelty and redundancy detection. In this paper, as a result, it is reasonable that we adopt the alteration of the values of F-measure, which combines precision with recall, to measure the loss of textual information. In the following experiments, we propose two experimental methods to obtain the values of F-measure, which use keywords or E-FCM as domain core to compute the precision, recall and F-measure.

a. Measurement of textual information loss

Through the experimental methods described above, we calculate the precision and recall so as to measure the loss of information by

$$L = e^{p} \tag{15}$$

where $p = \dfrac{f_{Before\_Merge} - f_{Merged}}{f_{Before\_Merge}}$

Herein, either f_{Before_Merge} or f_{Merged} is the value of F-measure, which is defined by:

$$f = \frac{2 * precision * recall}{precision + recall} \tag{16}$$

For the equivalent classes of E-FCMs strongly depend on the co-occurrence element concepts in E-FCMs, after the merging, the distribution of element concepts on E-FCMs tends to be concentrated. If the loss of textual information is too much, textual information would be losing seriously. So the trend of textual information loss should increase slowly. Through the change of textual information loss, the below experiments will illustrate that formula (15) can effectively measure the loss of textual information in the merging process of textual knowledge represented by E-FCMs.

b. Compute the precision, recall and F-measure using keywords as domain core

The main steps of the process are as following.

(1) Select a set of *n* keywords in each domain by random as the domain core;

(2) Cosine similarity is used to calculate the similarity between the domain core and each E-FCM in domains. For the domain core *dc* and E-FCM *ef*, their similarity is calculated by

$$similarity(dc, ef) = \sum_{w \in dc \cap ef} weight(dc, w) * weight(ef, w) \tag{17}$$

where *weight(dc,w)* is denoted as 1/*n*. *weight(ef,w)* represents the associated weight from word *w* to E-FCM *ef*;

(3)Add E-FCM *ef* to domain d_k, if the similarity between *ef* and the core of domain d_k is the maximal value among all domains;

(4)If there exists an E-FCM that has not been added to any domain, go to step (2); otherwise end.

c. Compute the precision, recall and F-measure using E-FCM as domain core

The main steps of the process are as following.

(1)For each domain, select an E-FCM as its domain core, respectively;

(2)Calculate the similarity between the domain core *dc*, which is represented by an E-FCM and E-FCM *ef* using formula (17);

(3)Add E-FCM *ef* to domain *k*, if the similarity between *ef* and the core of domain *k* is the maximal value among all domains;

(4)If there exists an E-FCM that has not been added to any domain, go to step (2); otherwise end.

G. Evaluate the quality of the Merging Process of E-FCMs

To evaluate the quality of the merging process, the following two factors should be considered.

(1)The loss of textual information in the merging process should be as less as possible;

(2)The number of E-FCMs that have been merged in the merging process should be as more as possible.

In order to obtain the quality of the merging process, a formula is defined as following.

$$QoM = \left(e^{\frac{N_{Before\_Merge} - N_{Merged}}{N_{Before\_Merge}}} \right) \Big/ L \tag{18}$$

where N_{Before_Merge} is the number of the total E-FCMs before the merging and N_{Merged} is the number after the merging. *L* is the loss of textual information.

Therefore, the bigger the value of *QoM* is, the better the quality of the merging process of E-FCM is.

IV. Experiments and Analysis

Experiments are firstly done to evaluate the method based on different domain cores. Then we use cosine-similarity metric whose roughly similarity calculating uses the algorithm of *ppjoin* [8] to get the merging process start and its result is compared with ours. Finally we show the results using different parameters both in roughly and precisely evaluating similarities.

A. Data set

We selected the Web site called Reuters (www.reuters.com) as our data source and chose three domains in it, which include *environment*, *health* and *internet*. By the Web crawler, we downloaded 6690 Web

TABLE I.
PERFORMANCE OF DIFFERENT NUMBERS OF KEYWORDS

K-num	AvgRecall	AvgPrecision	AvgF-Measure
10	0.337606774	0.390139212	0.228750762
20	0.377054528	0.431128093	0.330117425
30	0.367330758	0.401764068	0.344162428
40	0.332920742	0.32896658	0.309087979
50	0.381307769	0.387858681	0.354043796
60	0.41780862	0.431573977	0.41144752
70	0.494468191	0.515839933	0.479395113
80	0.492345539	0.499145377	0.487235795
90	0.508350511	0.506632449	0.501744591
100	0.51814685	0.52011294	0.512033864

pages from March 2007 to September 2008 belonging to the domain of *environment*, 8168 Web pages from January 2007 to September 2008 belonging to the domain of *health* and 4158 Web pages from January 2007 to September 2008 belonging to the domain of *Internet*. Then with the Web pages we crawled, 19016 E-FCMs have been generated, which belong to the domains of *environment*, *health* and *internet*.

B. Select domain core

We firstly select keyword sets with the number of 10, 20, …, 100 from each domain as its domain core. As seen in Table 1, the result indicates that recall and precision will achieve better when the number of keywords get bigger. As a result, we choose 10 sets of 100 keywords in each domain as cores to compare with the cores represented by E-FCM and the results are shown in Table 2. From Table 2, we find that the value of F-Measure achieves better when the cores are represented by E-FCM than by set of keywords.

C. The algorithm of ppjoin

Table 2 shows that when the domain core is represented by E-FCM, the value of Core 6's F-measure achieves best. Therefore, in the following experiments, we select the 6^{th} core as the domain core.

In the merging process of E-FCMs, the key step of merging is the similarity calculating both roughly and precisely. In this section, we use the algorithm of *ppjoin* [8] to compute the similarities between E-FCMs and using our merging algorithm to get the candidate merged. Finally, the experiment results are shown in Figure 4.

As shown in Figure 4, we can see that when the merging threshold is tuned lower than 0.6, the value of *QoM* gets very higher; however, it makes no sense when the threshold is too low because there is much irrelevant textual knowledge having been merged when the merging threshold is tuned very low. As a result, it is reasonable that we only take care of the values after 0.6. From Figure 4 we know that the value of *QoM* is not bigger than 1.1 when merging threshold is no more than 0.6. Moreover, as shown in Figure 9, the experiment results using our algorithm performs better for the values of *QoM* are all bigger than 1.18.

D. Roughly calculating

In the process of roughly calculating the similarities between E-FCMs, we have two thresholds to evaluate. One is *m*, which means that if not less than *m* percent of element concepts are the same between two E-FCMs, they are belong to one similar candidate set. Another is *n*, which means that *n* percent of E-FCMs in the similar candidate set will be added to a similar set. The results of *m* and *n* are shown in Figure 5 and Figure 6, respectively.

In Figure 5 and Figure 6, we can see that when *m*=0.5 and *n*=1, the quality of the merging process get the highest score. In addition, the quality changes much when *m* is tuned, whereas it changes little when *n* is tuned. It indicates that when the merging threshold *m* is set too low or too high, the effectiveness of merging process achieves unsatisfactory results. When the threshold *m* is set too low, the size of the similar sets we get after roughly similar calculating will become very large. It makes no sense of the process of roughly similar calculating. However, if *m* is set too bigger, only a few noisy and redundancy textual information will be merged, whereas most of other noisy and redundancy textual information has not been merged. In addition, results of parameter *n* reflects that when reserving all of the elements in one similar candidate set, the quality of merging achieves best results.

TABLE II.
PERFORMANCE OF DIFFERENT DOMAIN CORES

Core	F-Measure(KW)	F-Measure(E-FCM)
1	0.396799419	0.377031626
2	0.501588351	0.538662471
3	0.480714994	0.447162796
4	0.450223244	0.605151643
5	0.361911119	0.501922842
6	0.410504405	0.639186893
7	0.436868602	0.547632355
8	0.521751974	0.560767449
9	0.440945571	0.568287708
10	0.512033864	0.533815714
AVG	0.451334154	0.53196215

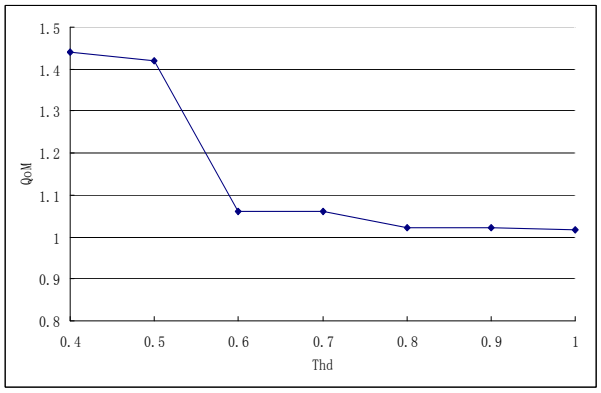


Figure 4. Performance of the merging using the algorithm of PPJoin

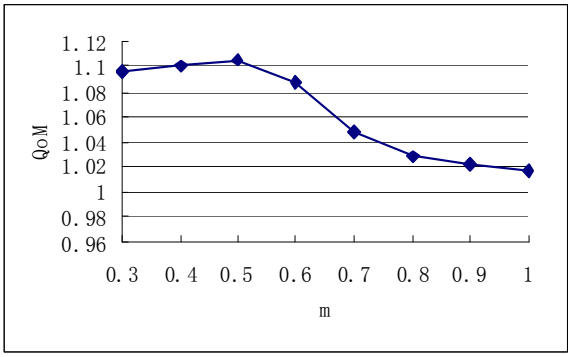


Figure 5. Performance of different *m*

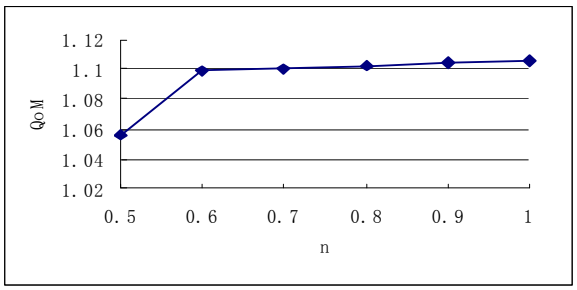


Figure 6. Performance of different *n*

E. *Precise calculation*

In the process of precisely calculation, we have proposed three strategies to calculate the similarities between E-FCMs in the same similar set. Herein, we'll evaluate these three strategies, and choose the best strategy to do the next experiments.

There are three parameters to be evaluated during the process of precisely calculation, which are α, λ and *T*.

Figure 7 shows that the quality of the merging process between E-FCMs achieves best results when we select strategy 3, whereas the qualities using strategy 1 and 2 perform almost the same results but much lower than using strategy 3. Additionally when α <0.5, the performance of these three strategies is nearly the same results for the reason that the associated weights between co-occurrence concepts have less effects on the merging process than the associated weights from co-occurrence concepts to theme concept.

Figure 8 and Figure 9 shows the performance of different λ and *T* (λ and *T* see formula (8)). λ and *T* are static and dynamic thresholds, respectively, which determine the merging threshold ξ. Experiment results of the parameters λ and *T* reflect that when the merging threshold ξ is set very low, the merging performance achieves better. The reason for the results may have something to do with the scale of our dataset as there does not exist much noisy or redundancy information. However, when the scale of textual information becomes extremely large, the merging algorithm we proposed method get an excellent performance.

F. *The variation of F-measure*

Figure 10 shows the variation of the values of F-measure as λ is tuned. We can see that the values of F-measure changes little when the static merging threshold changes. As a result, we'll get the conclusion that the loss of textual information is little after removing the noisy and redundancy information with our merging algorithm.

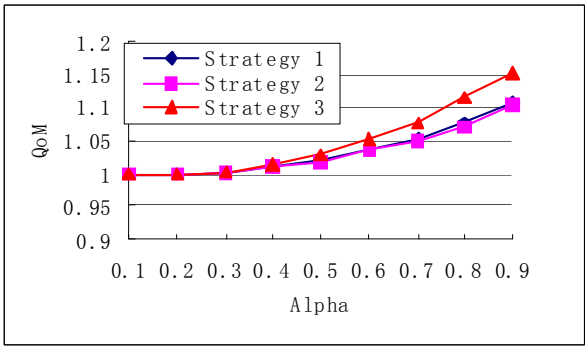


Figure 7. Performance comparison of different strategy

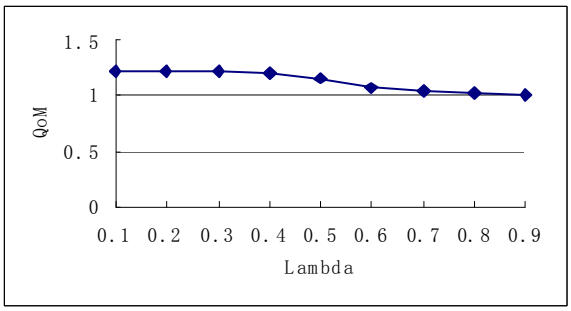


Figure 8. Performance of different λ

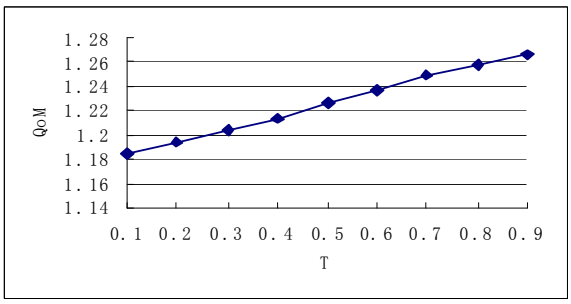


Figure 9. Performance of different *T*

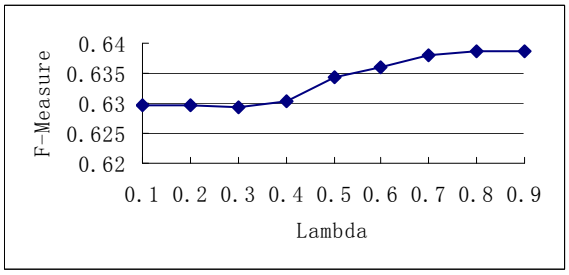


Figure 10. F-measure of different λ

V. Conclusions and Future Work

The noisy and the redundant information of textual knowledge have significant impacts on the complexity of the algorithm of text automatic classification, clustering as well as relevant knowledge aggregation and integration in e-Science, scientific workflow and e-Learning systems. The merging algorithm of textual knowledge represented by element fuzzy cognitive maps is proposed to reduce the noisy and redundant information in a similar set, which effectively decrease the qualitative requirements of the training texts according to the algorithm in [9] and lowers the noisy and redundant information hidden in the original E-FCMs. A formula consists of the changes of F-measure after merging process is employed as an indictor to measure the loss of textual information during the merging process of E-FCMs, and *QoM* is defined to

measure the quality of the merging process between E-FCMs. Through the experiments comparing with the algorithm of *ppjoin* that adopts *cosine* similarity as its similarity measurement, it is obviously that the merging algorithm we proposed can restrain the noise and eliminate the redundant information hidden in original E-FCMs effectively.

Our work has a broad way for future improvements and extensions. For an instance, we will make efforts to improve the precision and efficiency of proposed algorithm and apply it to promote the robust and dynamic verification of scientific workflow systems [21-22].

ACKNOWLEDGEMENT

Research work is supported by the Shanghai Science and Technology Commission (grants 09JC1406200), National Science Foundation of China (grants 90612010, and 60402016), the National Basic Research Program of China (grants. 2003CB317008) and the Shanghai Leading Academic Discipline Project (J50103).

REFERENCE

[1]. G. Salton, M.J. McGill, Introduction to Modern Information Retrieval. McGraw-Hill Book Company, New York. 1983.120-123.

[2]. T. Hofmann (1999). Probabilistic latent semantic indexing. in Proceedings of the 22nd International Conference on Research and Development in Information Retrieval (SIGIR"99), 50-57.

[3]. D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet Allocation. Journal of Machine Learning Research, 2003, 3: 993-1022.

[4]. R. Z. Michal, G. Thomas. The Author-Topic Model for Authors and Documents. http://www. datalab. uci.edu/ author-topic/398.pdf

[5]. A. McCallum, A. C. Emmanuel,et.al. The author-recipient -topic model for topic and role discovery in social networks: experiments with Enron and Academic email. http://www.cs.umass. edu/ ~mccallum/papers /art04tr.pdf.

[6]. D. M. Blei, J.D. Lafferty. Correlated Topic Models. http://www.cs.princeton.edu/~blei/ papers/ BleiLafferty 2006.pdf.

[7]. D.Cohn, T. Hofmann. The missing link: A probabilistic model of document content and hypertext connectivity. Neural Information Processing Systems, 2001, 13, 430-436.

[8]. C Xiao, W Wang, X Lin, JX Yu. Efficient Similarity Joins for Near Duplicate Detection. Proceeding of the 17th international conference on World Wide Web, April, 2008, 131-140.

[9]. H.Zhuge, X.F. Luo. Automatic generation of document semantics for the e-Science Knowledge Grid. Journal of Systems and Software, 2006, 79, 969-983.

[10]. X.F. Luo, N.Fang, et.al. Semantic Representation of Scientific Documents for the e-Science Knowledge Grid. International Journal of Concurrency and Computation: Practice and Experience, 2008, 20(7): 839-862.

[11]. H. Anthony, S. Rupert. A knowledge-based approach to merging information Knowledge-Based Systems, Volume 19, Issue 8, December 2006, 647-674.

[12]. H. Anthony, Merging structured text using temporal knowledge, Data & Knowledge Engineering, Volume 41, Issue 1, April 2002, 29-66.

[13]. A. Leila and K. Souhila. An argumentation framework for merging conflicting knowledge bases. International Journal of Approximate Reasoning, Volume 45, Issue 2, July 2007, 321-340.

[14]. W. Z. Christopher, P. R. Loren and R. R. Terry. Automated merging of conflicting knowledge bases, using a consistent, majority-rule approach with knowledge-form maintenance. Computers & Operations Research, Volume 32, Issue 7, July 2005, 1809-1829.

[15]. S. Benferhat, D. Dubois, S. Kaci, H. Prade, Possibilistic merging and distance-based fusion of propositional information, Annals of Mathematics and Artificial Intelligence 34 (1–3) (2002) 217–252.

[16]. S. Benferhat, D. Dubois, H. Prade, M. Williams, A practical approach to fusing and revising prioritized belief bases, in: Proceedings of the 9th Portuguese Conference on Artificial Intelligence (EPIA'99), 1999, 222–236.

[17]. J. P. Delgrande and S.Torsten. A consistency-based framework for merging knowledge bases. Journal of Applied Logic, 5(3), September 2007, 459-477.

[18]. B.Chaib-Draa. Causal Maps: Theory, Implementation and Practical Applications in Multi-agent Environments. IEEE Trans. on Knowledge and Data Engineering, 2002, 14(6): 1 - 17.

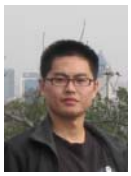
[19]. P. C. Silva. New Forms of Combinated Matrices in Fuzzy Cognitive Maps. In: Proc of the IEEE International Conference on Neural Network, New York, 1995, 771-776.

[20]. Y. Zhang, P. Callan, T. Minka. Novelty and Redundancy Detection in Adaptive Filtering, Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval, Aug. 2002,81-88.

[21]. J. Chen and Y. Yang, Temporal Dependency based Checkpoint Selection for Dynamic Verification of Temporal Constraints in Scientific Workflow Systems. ACM Transactions on Software Engineering and Methodology, 2009.

[22]. M. Wang, R. Kotagiri and J. Chen, Trust-based Robust Scheduling and Runtime Adaptation of Scientific Workflow, Concurrency and Computation: Practice and Experience, ISSN: 1532-0626, Wiley, 2009.

Xiangfeng Luo received the master degree in Hefei University of Technology in 2000 and the Ph.D. degree in the same school in 2003. He was a post doctor at the China Knowledge Grid Research Group of Institute of Computing Technology (ICT) in Chinese Academy of Sciences (CAS) from 2003 to 2005. He is currently an associate professor at School of Computers in Shanghai University, and his main research interests include the Web content analysis and cognitive informatics.



Jun Zhang received the bachelor degree in Shanghai University in 2008. He is currently a graduate student at School of Computers in Shanghai University and his main research interests include online word relation discovery and topic detection and tracking.