An Algorithm for Fuzzy Concept Lattices Building with Application to Social Navigation

Chunzhi Xie Liangzhong Yi Yajun Du

School of Mathematics and Computer Engineering, Xihua University, Chengdu 610039, P.R.China

Abstract

Due to the fuzziness of information, fuzzy concept lattices (FCL) are proposed as effective tools for data analysis and knowledge discovery on the fuzzy formal context. But the mining of large database needs efficient algorithms for FCL building. In this paper, a fast and automatic algorithm for FCL building is proposed, called FCLB algorithm, for generating fuzzy concepts from a given fuzzy formal context and a set of fuzzy sets, automatically building a fuzzy concept lattice and representing it graphically. This work has been applied successfully to the social navigation system which aimed to aid users in finding information more effectively.

Keywords: Fuzzy concept lattices, Fuzzy formal context, Social navigation

1. Introduction

Formal Concept Analysis(FCA)[1] was first proposed by Mr. Wille in 1982, it has been an efficient way to acquire rules and express knowledge. As an efficient tool, FCA has been successfully applied to the domains such as decision systems, information retrieval, data mining and knowledge discovery. But many of the the information people facing are usual fuzzy and unprecise, so are the formal context FCA based on. By introducing Fuzzy sets [2] into formal context, it can express the fuzzy characteristic between the objects and attributes. In 1994, Burusco applied fuzzy sets theory to FCA, and proposed a model of fuzzy concept lattice (FCL) [3]. In his further work, he calculated the L-fuzzy concepts and represented graphically the L-fuzzy concept lattice [4]. After that, Belohlavek [5-9], Georgescu [10] and Shi-Qing Fan[11] have made a further study of the theory respectively. In their work, four different forms of fuzzy concept are proposed called fuzzy formal concepts, fuzzy property-oriented formal concepts, fuzzy object-oriented formal concepts and dual fuzzy formal concepts respectively. All of them are called fuzzy concepts uniformly[11]. In the work [12], a new FCL model is proposed to deal with the continuous membership degree in fuzzy formal context. By far, FCL theory have been applied into some applications, such as Citation-based Document Retrieval[13] and social navigation[14]. But the applications based FCL must find an efficient way to build the corresponding FCL. In this paper, a fast and automatic algorithm for FCL building is proposed, called FCLB algorithm, for generating fuzzy concepts from a given fuzzy formal context and a set of fuzzy sets, automatically building a fuzzy concept lattice and displaying it graphically. This work has been applied successfully to the social navigation system which aimed to aid users in finding information more effectively.

2. Preliminaries

Definition 1. A triple $(G, A, \tilde{I})$ is called a fuzzy formal context. Where $G = \{x_1, x_2, \cdots, x_n\}$ is a set of objects, and $A=\{a_1,a_2,\cdots,a_m\}$ is a set of attributes, $\tilde{I}$ is a fuzzy relation mapping from $G\times A$ to interval [0, 1] i.e. $\tilde{I}$: $G\times A\rightarrow[0, 1]$. If $\tilde{I}(x, a)=\alpha$ then the object x has the attribute a with the degree α.

Definition 2. Let $(G, A, \tilde{I})$ be a fuzzy formal context, "$\rightarrow$" is an implicative operator, for all $\tilde{X}$: $G\rightarrow[0, 1]$, $\tilde{B}$: $A\rightarrow[0, 1]$,we can define the operators as follows [15]:

$$\tilde{X}^*(a) = \bigwedge_{x\in G} (\tilde{X}(x) \rightarrow \tilde{I}(x, a)) \tag{1}$$

$$\tilde{B}^*(x) = \bigwedge_{a\in A} (\tilde{B}(a) \rightarrow \tilde{I}(x, a)) \tag{2}$$

A pair$(\tilde{X},\tilde{B})$ is called a fuzzy concept if and only if $\tilde{X}^*=\tilde{B}$,$\tilde{B}^*=\tilde{X}$. $\tilde{X}$ is called its extent, and $\tilde{B}$ is called its intent.

Definition 3. The subset $\mathfrak{L}(G,A,\tilde{I})=\{(\tilde{X},\tilde{B})|\ \tilde{X}^*=\tilde{B},\tilde{B}^*=\tilde{X}\}$ with the order relation: $(\tilde{X}_1,\tilde{B}_1)\leq(\tilde{X}_2,\tilde{B}_2)$ is called a complete lattice if and only if $\tilde{X}_1\subseteq\tilde{X}_2$(or equivalently $\tilde{B}_2\subseteq\tilde{B}_1$). The lattice $\mathfrak{L}((G,A,\tilde{I}),\leq)$ is called a complete fuzzy concept lattice.

3. Algorithm for fuzzy concept lattices building(FCLB)

The FCLB algorithm is to generate all the fuzzy concepts from a given fuzzy formal context $(G,A,\tilde{I})$ and a set of fuzzy sets $\tilde{L}=\{\tilde{X}(x)|x\in G\}$, build FCL and present it graphically. There are four problems we must resolve: (1) How to generate fuzzy concepts; (2) How to prune redundant fuzzy concepts; (3) How to seek out all the partial ordered sets; (4) How to locate fuzzy concepts. Here is the algorithm we mentioned.

Algorithm 1. Fuzzy Concept Lattice Building (FCLB).
Input: a file of the fuzzy formal context $FFCFile.txt$, a file of the set of fuzzy sets named $FSFile.txt$.
Output: Fuzzy concept lattice graphically.

```
TempCpt.next ← null
LastCpt.next ← null
Ĩ(x,y) ← ReadFile("FFCFile.txt")
m ← GetObjectNum Ĩ(x,y)
n ← GetAttribNum Ĩ(x,y)
while ReadFileRow(FSFile ≠ null)
    X̃(x) ← ReadFileRow("FSFile.txt"))
/ *generating fuzzy concepts * /
    Cpt ← CalConcept(Ĩ(x,y), X̃(x), m, n)
/ * pruning redundant fuzzy concepts * /
    PruReducon(TempCpt, Cpt)
end while
SortByExtent(TempCpt)
/*Sort Ascending the fuzzy concept by X̃(x) */
SortAscending(TempCpt, LastCpt)
OutPutFile(LastCpt,"LastCpt.txt")
NewNode ← ReadFileRow("LastCpt.txt")
nodes← ∅
while (NewNode ≠ null))
 /* building fuzzy concept lattice by inserting directly
    InsertNewConcept(Nodes, NewNode)
    ReadNodeMap()
end while
```

3.1. Generating all the fuzzy concepts

As we know, fuzzy concepts can be calculated from Eq.(1)(2). If we select the implication operator $R(a,b)$=(1-a+b)$\wedge$1[16], then Eq.(1)(2) can be substituted for the following:

$$\tilde{X}^*(a)=\bigwedge_{x\in G}((1-\tilde{X}(x)+\tilde{I}(x,a))\wedge 1) \quad (3)$$

$$\tilde{B}^*(x)=\bigwedge_{a\in A}((1-\tilde{B}(a)+\tilde{I}(x,a))\wedge 1) \quad (4)$$

That is, for a given $\tilde{I}(x,a)$ and $\tilde{X}(x)$ we can calculate all the fuzzy concepts from Eq.(3)(4).

Algorithm 2. Calculate one fuzzy concept(CalConcept).
Input: a fuzzy formal context $\tilde{I}(x,a)$, a fuzzy set $\tilde{X}(x)$, number of fuzzy formal context $\tilde{I}(x,a)$ columns m, number of fuzzy formal context $\tilde{I}(x,a)$ rows n;
Output: a fuzzy concept:$(\tilde{X},\tilde{B})$.

```
for all a_j ∈ A
  for all x_i ∈ X
    temp[i] ← min(1,1-X̃(x_i)+ Ĩ(x_i, a_j)
   end for
  for i ← 0 to m − 1
    temp ← min(temp[i])
  end for
  /* calculate the intent of a fuzzy concept */
  ConceptInte[j] ← temp
end for
for all x_i ∈ X
  for all a_j ∈ A
    temp[i] ← min (1, (1 − ConceptInte[j] + Ĩ(x_i, a_j))
  end for
  for j ← 0 to n-1
    temp ← min(temp[j])
  end for
  /* calculate the extent of a fuzzy concept */
 ConceptExte[i] ← temp
end for
```

3.2. Pruning redundant fuzzy concepts

From algorithm 2 we know, if given a fuzzy set $\tilde{X}(x)$, we can get a fuzzy concept. If given a whole fuzzy sets $\tilde{L}=\{\tilde{X}(x)|x\in G\}$, we can get a whole fuzzy concepts. But fuzzy concepts constructing is not only related to fuzzy sets, but also to the

fuzzy formal context and the implication operator. So, fuzzy sets presenting different natural language may be same, consequently the same of fuzzy concepts. Two different fuzzy sets may get the same fuzzy concepts. So redundant concepts must be pruned. Two fuzzy concepts $(\tilde{X}_1, \tilde{B}_1)$,$(\tilde{X}_2, \tilde{B}_2)$ are equal to each other if and only if $\tilde{X}_1 \subseteq \tilde{X}_2$(or equivalently$\tilde{B}_2 \subseteq \tilde{B}_1$).

Algorithm 3. Pruning redundant concepts (PruReducon).
Input: a temporary link list for fuzzy concepts $TempCpt$, a new fuzzy concept Cpt;
Output: if the new fuzzy concept Cpt is not equal to any of fuzzy concepts in $TempCpt$, then Cpt is added into $TempCpt$.

if $(TempCpt.next \neq null)$
  Insert$(TempCpt, Cpt)$
else
  $p \leftarrow Cpt1.next$
  while $(p \neq null)$
    if (Equal$(p.Cpt, Cpt)$) **then** exit
  end while
  if $(p = null)$ Insert$(TempCpt, Cpt)$
end if

3.3. Sorting by $\tilde{X}(x)$

After pruning redundant concepts, there are three relations: "¡", "¿" and neither "¡" nor " ¿" between two fuzzy concepts. If we sort ascending the concept by its extent, the relation of the two adjacent concepts can only two: "¡" and neither "¡" nor " ¿". we can make use of them to attain all father concepts of a given one. The necessary condition for the relation "¡" of two fuzzy concepts $(\tilde{X}_1,\tilde{B}_1)$,$(\tilde{X}_2,\tilde{B}_2)$ is

$$\tilde{X}_1(x_i) = \tilde{X}_2(x_i) \wedge \tilde{X}_1(x_{i+1}) < \tilde{X}_2(x_i)(1 \leq i \leq m)$$

or

$$\tilde{B}_1(a_j) = \tilde{B}_2(a_j) \wedge \tilde{B}_1(a_{j+1}) > \tilde{B}_2(a_j)(1 \leq j \leq n)$$

where m is the number of objects, n is the number of attributes in fuzzy formal context.
Algorithm 4. Sort ascending fuzzy concepts by $\tilde{X}(x)$(SortAscending)
Input: a temporary link list for fuzzy concepts $TempCpt$, a link list for fuzzy concepts after sorting $LastCpt$;
Output:Sort fuzzy formal concepts by $\tilde{X}(x)$.

$Cpt1 \leftarrow TempCpt.next$
while $(Cpt1 \neq null)$
  /*finding maximal fuzzy concepts */
  $preq \leftarrow$ FindPreMaxCon$(Cpt1)$
  /*insert maximal fuzzy concepts into list $LastCpt$ */
  Insert$(LastCpt, preq)$
  /*delete maximal fuzzy concepts from list $TempCpt$ */
  ReDelete$(TempCpt, preq)$
end while

3.4. Partial ordered relation in FCL

Algorithm 4 makes the two adjacent concepts only have two relations. For a new fuzzy concept S, let K presents a fuzzy concept in FCL, then the partial ordered relations of it can be attained from following:

(1) If no node in FCL, S is added as a father concept(seen in Fig.1);

(2) $\forall$ K, if K ¡ S and there is no children for K, then S is the child of K, K is the father of S(seen in Fig. 2);

(3) $\forall$ K, if K ¡ S, K has children but $\neg\exists$ $K.children$ ¡S , then S is the children of K, K is the father of S(seen in Fig. 3);

(4) $\forall$ K, if $K < S$, K has children, and $\exists$ $K.children < S$, then S is the children of $K.children$, K is the super father of S(seen Fig 4);

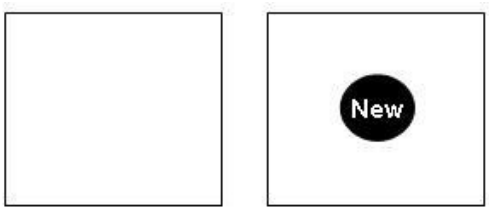

Fig. 1: S is added when the FCL is empty

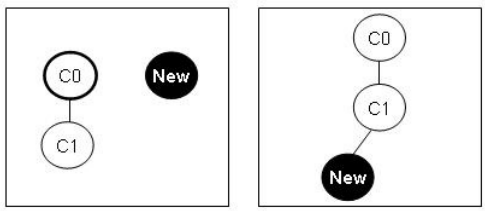

Fig. 2: S is added when K¡S and K has no children.

In this paper, fist in first out-branch and bound methods are applied to seek all the possible father concepts of the new concept S, which can make sure

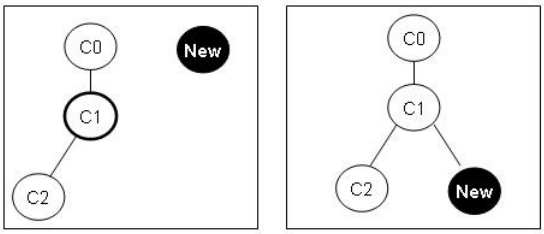

Fig. 3: S is added when $K < S$, K has some children, but $\neg \exists\ K.children < S$.

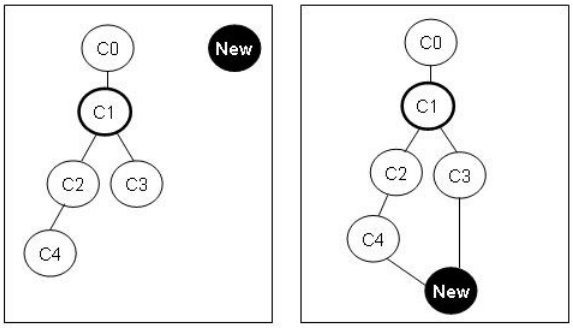

Fig. 4: S is added when $K < S$, K has some children, and $\exists\ K.children < S$.

not to search the space with root of a logjam. Algorithm 5 gives the process of seek all father concepts of S in FCL.

3.5. Hierarchy structure of FCL

The hierarchy of a new fuzzy formal concept is determined by its father concepts, but the number of the father concepts is likely more than one. So, let m be maximal hierarchy of the father concepts, then the hierarchy of the new one is m+1. Based on the partial ordered relation in the FCL mentioned above, we can get all the father concepts of a new concept. For example, the hierarchy of the new concept New in Fig.4 is 4, and hierarchy will not change anymore when a new concept is added in.

Algorithm 5.Insert a new concept into FCL (InsertNewConcept).
Input: a new fuzzy concept from the file "$LastCptFile.txt$" $NewNode$, an array to store the father concept $Nodes$, a queue for possible father nodes $NodesQueue$.
Output: prensent the fuzzy concept lattice graphically.

$NodesCount \leftarrow$ GetNodesSize($Nodes$)
/*when FCL is empty, new node is the father*/
if ($NodesCount$=0) Add($nodes, NewNode$)
push($NodesQueue$, $Nodes$[0])
while ($NodesQueue \neq$ null)
$Pnode \leftarrow$ Pop($NodesQueue$)
if (Count($Pnode.children$)$\neq$ 0)
if ($\exists$ ($Pnode'schildren < NewNode$)
/*$Pnode$ is the $NewNode$'s father fuzzy concept*/
Push($NodesQueue$,$Pnode.children[i]$)
else
/*$Pnode$ has children but none of them are smaller than $NewNode$*/
$Rnodes \leftarrow Pnode$
end if
else
/*$Pnode$ has no child*/
$Rnodes \leftarrow Pnode$
end if
end while
$HierarchyNew \leftarrow$ MaxParent($Renodes$)+1
$count \leftarrow$ GetNodes($Nodes$,$HierarchyNew$)+1
set relation between the $NewNode$ and $Renodes[i]$
get every node position by its hierarchy and the $count$
RedrawMap()

3.6. Example of FCLB

Given a fuzzy formal context as Fig.5, Where G= $\{g_1, g_2, g_3\}$, A=$\{a_1, a_2, a_3\}$, L ={0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1} which presents the fuzzy sets possible values, the fuzzy concept lattice we can get is as seen in Fig.6. Here the fuzzy concepts are the whole which can be obtained by L. In the FCL, the minimal element is c0 and the maximal element is c15, so it is a complete lattice.

Table 1: The fuzzy formal context.

X \ Y	y_1	y_2	y_3
x_1	0.6	0.6	0.0
x_2	0.9	0.5	0.3
x_3	1.0	0.2	0.7

Table 2: All fuzzy concepts.

c0= ((0.3, 0.3, 0.3), (0.7, 0.7, 0.7))	c8= ((0.4, 0.4, 0.7), (0.6, 0.3, 0.6))
c1= ((0.3, 0.3, 0.4), (0.7, 0.6, 0.7))	c9= ((0.5, 0.5, 0.5), (0.6, 0.5, 0.5))
c2= ((0.3, 0.3, 0.5), (0.7, 0.5, 0.7))	c10= ((0.5, 0.5, 0.6), (0.6, 0.4, 0.5))
c3= ((0.3, 0.3, 0.6), (0.7, 0.4, 0.7))	c11= ((0.5, 0.5, 0.7), (0.6, 0.3, 0.5))
c4= ((0.3, 0.3, 0.7), (0.7, 0.3, 0.7))	c12= ((0.6, 0.5, 0.5), (0.6, 0.5, 0.4))
c5= ((0.4, 0.4, 0.4), (0.6, 0.6, 0.6))	c13= ((0.6, 0.6, 0.6), (0.6, 0.4, 0.4))
c6= ((0.4, 0.4, 0.5), (0.6, 0.5, 0.6))	c14= ((0.6, 0.6, 0.7), (0.6, 0.3, 0.4))
c7= ((0.4, 0.4, 0.6), (0.6, 0.4, 0.6))	c15= ((0.6, 0.7, 0.7), (0.6, 0.3, 0.3))

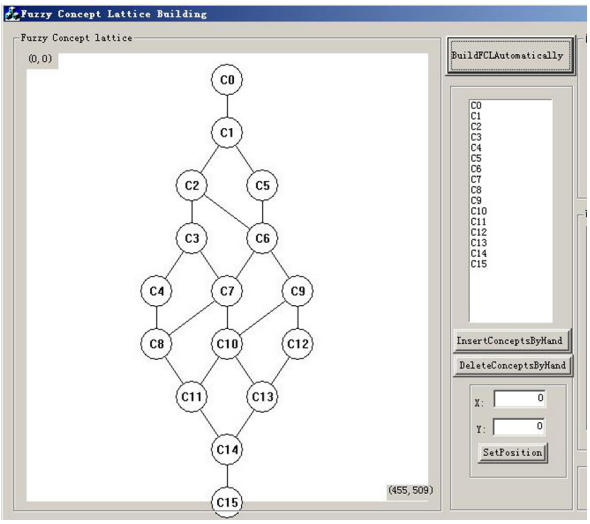

Fig. 5: The corresponding FCL .

4. Applying to Social navigation

Social navigation is a normal behavior in social daily life, which is not only impelled by the inherence aim, the cognitive process of his/her but also influenced by others and the society. By social navigation, people share knowledge, experience and leave them to their successors. With the rapid progress of Internet, it is more and more difficult for Web search engines to find information which can express the user's interests exactly [17]. Applying this real behavior to navigate in Internet will help people to find information more effectively.

Recently a social navigation system model based on FCL was proposed[14]. The system first obtains a fuzzy formal context by formalizing users' traces, then obtains fuzzy sets based on the query words people left, thirdly calculates the fuzzy formal concepts by the fuzzy sets and the fuzzy formal context and build the FCL, finally provides web pages with higher degree to people. Note that every fuzzy set in the social navigation system implies a special need, we call these fuzzy sets interesting fuzzy sets. It may be only a subset of L, so the minimal element or the maximal element may not exist in the FCL. Thus the lattice in social navigation is a special FCL. As seen in Fig.7, the number of interesting fuzzy sets is 64, and what number of the corresponding fuzzy concepts we can get is only 20 after pruning redundant ones. There is no maximal element in the FCL, but all the concept in FCL are interesting.

Table 3: The fuzzy formal context of social navigation.

	Page1	Page 2	Page 3	Page 4	Page 5
fuzzy	0.83	1.00	0.60	0.38	0.43
fuzzy concept lattice	0.80	0.90	0.73	0.38	0.43
fuzzy inference	0.81	1.00	0.00	0.58	0.71
social navigation research	0.90	0.83	0.73	0.80	0.85

Table 4: All fuzzy concepts of social navigation.

c0= ((0.38, 0.38, 0.00, 0.73), (1.00, 1.00, 1.00, 1.00, 1.00))
c1= ((0.38, 0.38, 0.40, 0.80), (1.00, 1.00, 0.60, 1.00, 1.00))
c2= ((0.38, 0.38, 0.50, 0.80), (1.00, 1.00, 0.50, 1.00, 1.00))
c3= ((0.39, 0.39, 0.08, 0.81), (1.00, 1.00, 0.92, 0.99, 1.00))
c4= ((0.40, 0.40, 0.00, 0.73), (1.00, 1.00, 1.00, 0.98, 1.00))
c5= ((0.40, 0.40, 0.50, 0.82), (1.00, 1.00, 0.50, 0.98, 1.00))
c6= ((0.43, 0.43, 0.70, 0.83), (1.00, 1.00, 0.30, 0.88, 1.00))
c7= ((0.57, 0.57, 0.00, 0.73), (1.00, 1.00, 1.00, 0.81, 0.86))
c8= ((0.57, 0.57, 0.40, 0.83), (1.00, 1.00, 0.60, 0.81, 0.86))
c9= ((0.57, 0.57, 0.81, 0.83), (1.00, 1.00, 0.19, 0.77, 0.86))
c10= ((0.58, 0.58, 0.27, 1.00), (0.90, 0.83, 0.73, 0.80, 0.85))
c11= ((0.60, 0.66, 0.00, 0.73), (1.00, 1.00, 1.00, 0.72, 0.77))
c12= ((0.66, 0.66, 0.40, 0.83), (1.00, 1.00, 0.60, 0.72, 0.77))
c13= ((0.66, 0.66, 0.81, 0.83), (1.00, 1.00, 0.19, 0.72, 0.77))
c14= ((0.68, 0.81, 0.08, 0.81), (0.99, 1.00, 0.92, 0.57, 0.62))
c15= ((0.70, 0.70, 0.50, 0.83), (1.00, 1.00, 0.50, 0.68, 0.73))
c16= ((0.72, 0.72, 1.00, 0.83), (0.81, 1.00, 0.00, 0.58, 0.71))
c17= ((0.81, 0.81, 0.50, 0.83), (0.99, 1.00, 0.50, 0.57, 0.62))
c18= ((1.00, 0.90, 0.70, 0.83), (0.83, 1.00, 0.30, 0.38, 0.43))
c19= ((1.00, 1.00, 0.40, 0.93), (0.80, 0.90, 0.60, 0.38, 0.43))

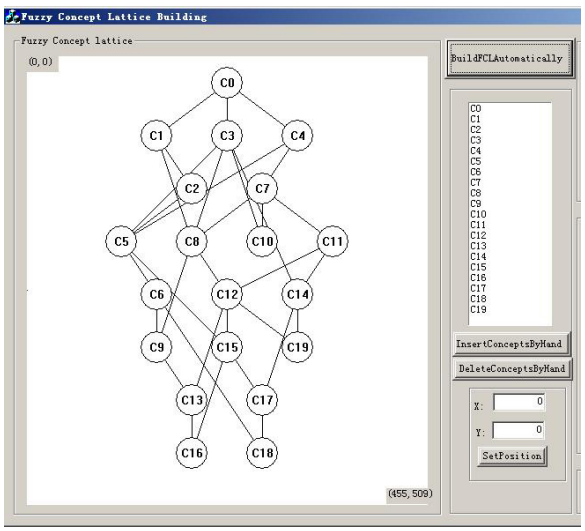

Fig. 6: The corresponding FCL of social navigation.

5. Conclusion

In this paper, a algorithm for fuzzy concept lattice building is proposed. It first generates fuzzy formal concepts from a given fuzzy formal context and a set of fuzzy sets, then prunes redundant concepts, again seeks out all the partial ordered sets and the hierarchy of all concepts, finally builds a fuzzy concept lattice and presents it graphically. This work has been applied successfully to the social navigation system which aimed to aid users in finding information more effectively. The time complexity is $\Theta(mn) + \Theta(p)$, where m is the number of objects, n is the number of attributes in fuzzy formal context and p is the size of fuzzy sets. Notice that the time complexity of FCLB is not only relative to the fuzzy formal context but also the size of fuzzy sets. So how to reduce fuzzy formal context and the size of fuzzy sets, how to build FCL more efficient and how to apply FCL to more applications effectively are our future work.

6. Acknowledgement

This work is supported by the Education Department Foundation of Sichuan Province (Grant No.2006A084 and 2006A086), the Application Foundation of Sichuan Province (Grant No.2006J13-056), The Cultivating Foundation of Science and Technology of Xihua University (Grant No.R0622611), The Cultivating Foundation of the Science and Technology Leader of Sichuan Province.

References

[1] B. Ganter and R. Wille, *Formal concept analysis: Mathematical foundations*, Springer, 1999.

[2] L.A. Zadeh, Fuzzy sets. *Inform. Control*, 8(3):338-353, Elsevier, 1965.

[3] A. Burusco and R. Fuentes, The study of L-fuzzy concept lattices. *Mathware Soft Comput*, 1(3):209-218, 1994.

[4] A. Burusco and R. Fuentes-Gonzfilez, Construction of the L-fuzzy concept lattice, *Fuzzy Sets and Systems*, 97:109-114, Elsevier, 1998.

[5] A. Burusco and R. Fuentes-Gonzáilez, Concept lattices defined from implication operators. *Fuzzy Sets and Systems*, 114 (2000):431-436, Elsevier, 2000.

[6] R. Belohlavek, Fuzzy Galois connections. *Math. Logic Quarterly*, 45(4):497–504, 1999.

[7] R. Belohlavek, *Fuzzy Relational Systems: Foundations and Principles*, Kluwer, NewYork, 2002.

[8] R. Belohlavek, Algorithms for fuzzy concept lattices. *prof. the Recent Advances in Soft Computing (RASC)*, *Nottingham (United Kingdom)*, pp.200-205, 2002.

[9] R. Belohlavek, Concept lattices and order in fuzzy logic. *Ann. Pure Appl. Logic* , 128 (1-3):277-298, 2004.

[10] G. Georgescu and A. Popescu, Non-dual fuzzy connections. *Archive Math. Logic*, 43(8):1009–1039,2004.

[11] S. Q. Fan, W. X. Zhang and W. Xu, Fuzzy inference based on fuzzy concept lattice. *Fuzzy Sets and Systems*, 157(24):3177-3187, 2006.

[12] Z. T. Liu, Y. Qiang and W. Zhou, A Fuzzy Concept Lattice Model and Its Incremental Construction Algorithm. *Chinese Journal of Computers*, 30(2):184-188, 2007.

[13] T. T. Quan, S. C. Hui and T. H. Cao, A Fuzzy FCA-Based Approach for Citation-based Document Retrieval. *Prof. the 2004 IEEE Conference on Cybernetics and Intelligent Systems* , *IEEE Computer Society*, *Singapore*, pp.578-583, 2004.

[14] C. Z. Xie, L. Z. Yi, Y. J. Du and Z. Pei, The Research of Social Navigation based on Fuzzy Concept Lattice, *Prof. the 6th International Conference on Computer and Information Science* (*ICIS*), *IEEE Computer Society, Melbourne (Australia),* 2007.

[15] W. X. Zhang and G. F. Qiu. Uncertain Decision MakingBased On Rough Sets. *Tsinghua University Press*, *Beijing*, 2005.

[16] G. J. Wang, *Non-Classical Mathematical Logic and Approximate Reasoning*, Science Press, Beijing, 2000.

[17] Y. J. Du H. M. Li, Z. Pei and H. Peng. Intelligent Spider's Algorithm of Search Engine Based on Keyword, *Transactions on computer and information theory*, 1(1):40-49, 2005.