Mining Negative Fuzzy Sequential Patterns

NANCY P. LIN[1], HUNG-JEN CHEN[1,2], WEI-HUA HAO[1],
HAO-EN CHUEH[1], CHUNG-I CHANG[1]
[1]Department of Computer Science and Information Engineering
Tamkang University,
151 Ying-Chuan Road, Tamsui, Taipei,
TAIWAN, R.O.C.
[2]Department of Industrial Engineering and Management
St. John's University,
499, Sec. 4, Tam-King Road, Tamsui, Taipei,
TAIWAN, R.O.C.

Abstract: - Many methods have been proposed for mining fuzzy sequential patterns. However, most of conventional methods only consider the occurrences of fuzzy itemsets in sequences. The fuzzy sequential patterns discovered by these methods are called as positive fuzzy sequential patterns. In practice, the absences of frequent fuzzy itemsets in sequences may imply significant information. We call a fuzzy sequential pattern as a negative fuzzy sequential pattern, if it also expresses the absencesof fuzzy itemsets in a sequence. In this paper, we proposed a method for mining negative fuzzy sequential patterns, called NFSPM. In our method, the absences of fuzzy itemsets are also considered. Besides, only sequences with high degree of interestingness can be selected as negative fuzzy sequential patterns. An example was taken to illustrate the process of the algorithm NFSPM. The result showed that our algorithm could prune a lot of redundant candidates, and could extract meaningful fuzzy sequential patterns from a large number of frequent sequences.

Key-Words: - Itemset, Fuzzy itemset, Large sequence, Sequential pattern, Fuzzy sequential pattern, Negative sequential pattern, Quantitative database

1 Introduction

Sequential pattern mining is to discover all frequent subsequences from a given sequence database, and it can be applied in divers applications such as basket analysis, web access patterns and quality control in manufactory engineering, etc. For example, users' web pages access sequential patterns can be used to improve a company's website structure in order to provide more convenient access to the most popular links. Thus, sequential pattern mining has become an important task in data mining field. Sequential patterns can be divided into Sequential Procurement [1, 2], and Cyclic Procurement [3, 4, 5, 6, 7, 8] by the sequence and the section of time.

A number of methods have been proposed to discover sequential patterns. Most of conventional methods for sequential pattern mining were developed to discover positive sequential patterns from database [1, 8, 9, 10, 11, 12]. Positive sequential patterns mining consider only the occurrences of itemsets in sequences. In practice, however, the absences of itemsets in sequences may imply valuable information. For example, web pages *A*, *B*, *C*, and *D* are accessed frequently by users, but *D* is seldom accessed after the sequence *A*, *B* and *C*. The web page access sequence can be denoted as $< A, B, C \neg D >$, and called a negative sequence. Such sequence could give us some valuable information to improve the company's website structure. For example, a new link between *C* and *D* could improve users' convenience to access web page *D* from *C*.

Moreover, most real world databases consist of numerical data. It is an important task to deal with the numerical data, and to discover information, which is suitable for human reasoning. To reach this goal, the fuzzy sets theory is commonly used, and the discovered sequential patterns are called fuzzy sequential patterns.

A fuzzy sequential pattern is called a positive fuzzy sequential pattern if it expresses only the occurrences of the fuzzy itemsets. In other words, a fuzzy sequential pattern is called a negative fuzzy sequential pattern if it also expresses the absences of fuzzy itemsets. Many methods have been proposed for mining fuzzy sequential patterns [13, 14, 15]. However, these methods only consider the appearances of fuzzy itemsets.

In this paper, we proposed a method for mining negative fuzzy sequential patterns, called NFSPM. In our method, absences of itemsets in sequences are also considered. Besides, only the sequences with high degree of interestingness can be selected as negative fuzzy sequential patterns.

2 Preliminary

In this section the basic concepts and derivatives of sequential pattern are described as follows.

2.1 Sequential Patterns

A sequence is an ordered list of itemsets. A positive sequence is denoted by $< s_1, s_2,..., s_n >$, and a negative sequence is denoted by $< s_1, s_2, ..., \neg s_n >$, where $\neg s_n$ represents the absence of itemset s_n. The length of a sequence is the number of itemsets in the sequence. A sequence with length l is called an l-sequence. We may note that a sequence $< s_1, s_2, ..., s_n >$ (or a negative sequence $< s_1, s_2, ..., \neg s_n >$ can also be written as $< <s_1, s_2, ...s_{n-1}>,< s_n >>$ (or $<< s_1, s_2, ...s_{n-1}.>,< \neg s_n >>$. That is a sequence can be regarded as an (n-1)-sequence $<s_1, s_2, ...s_{n-1}>$, denoted by s_{pre} and called a preceding subsequence, followed by a 1-sequence $<s_{n-1}>$(or $<s_{n-1}>$), denoted by s_{tar} and called a target subsequence. A sequence database D is a set of tuples (cid, s) with primary key cid that is a customer-id, and s that is a customer transaction sequence.

A positive sequence $< a_1, a_2, ..., a_n >$ is contained in a sequence $< s_1, s_2, ..., s_m >$ if there exist integers $1 \leqq i_1 < i_2 \ldots < i_n \leqq m$ such that $a_1 \subseteq s_{i_1}$, $a_2 \subseteq s_{i_2}$, …, $a_n \subseteq s_{i_n}$. A negative sequence $b = < b_1, b_2, ..., \neg b_n >$ is contained in a negative sequence $s = < s_1, s_2, ..., \neg s_m >$, if its positive counterpart $< b_1, b_2, ..., b_n >$ is not contained in s and the subsequence, $< b_1, b_2, ..., b_{n-1} >$, of b is contained in s.

The support of a sequence s, Supp(s), is α %, if α % of customer sequences in D contain s. A positive sequence a is called as sequential pattern (or large positive sequence) in D if Supp $(a) \geq \lambda_{ps}$, where λ_{ps} is the user-predefined threshold of the support of positive sequences. With the user-predefined threshold of the support of negative sequences, λ_{ns}, a negative sequence $b = < b_1, b_2, ..., \neg b_n >$ is called a negative sequential pattern (or large negative sequence) in D if Supp$(b) \geq \lambda_{ns}$ and the counterpart of the last itemset, b_n is a large 1-sequence. Note that the condition that b_n being a large 1-sequence is a must, which removes the trivial situation where sequences with itemset b_n occur infrequently.

2.2 Fuzzy Sequential Patterns

A fuzzy item is denoted by a.b where a is the item, and b is the fuzzy set associated with the item. For example, a fuzzy item may be denoted by AGE.YONG. A fuzzy itemset is a set of fuzzy items, and a fuzzy sequence is an ordered list of fuzzy itemsets. The fuzzy support, *fsupp*, of a fuzzy sequence is the percentage value of customers supporting this fuzzy sequence [13]. A fuzzy sequence is called a fuzzy sequential pattern if its *fsupp* is greater than or equal to a predefined threshold.

If a fuzzy sequential pattern expresses the absences of the fuzzy itemsets, we call it as a negative sequential pattern. In other word, a fuzzy sequential pattern is called a positive fuzzy sequential pattern if it expresses only the occurrences of the fuzzy itemset.

Algorithm: NFSPM
Input:
TD : Transaction database
λ_{ps} : Threshold of support of positive sequences
λ_{ns} : Threshold of support of negative sequences
λ_{ni} : Threshold of interestingness of negative sequences
Output:
NFS: Negative fuzzy sequential patterns
Method:
(1) Transform *TD* into *FTD* (i.e., transform each item in sequences in *TD* into fuzzy item.)
(2) Find the set of the large fuzzy itemsets, F = {The fuzzy itemsets whose $fsupp \geq \lambda_{ps}$}
(3) LP_1 = {All large fuzzy itemsets in F, each of which is recoded as an unique integer}
// **Finding all negative sequential patterns**
(4) $N = NSP(FTD, LP_1, \lambda_{ps}, \lambda_{ns}, \lambda_{ni})$
(5) *NFS* = {All negative sequential patterns in N, whose itemsets are mapped to the original fuzzy itemsets}
return *NFS* ;

Fig. 1. Algorithm NFSPM

3 Algorithm NFSPM

The algorithm is shown in fig. 1. There are five steps in the algorithm. In step 1, all the items in the transaction database are transformed into fuzzy items. In step 2, all the large fuzzy itemsets are found. In step 3, each large fuzzy itemset is recoded as a unique integer. Then, in step 4, the algorithm executes the procedure *NSP*, which discovers all negative sequential patterns from a given database. We describe the procedure *NSP* in subsection 3.3. After finding the negative sequential patterns, each code is mapped back to the original fuzzy itemset, Finally, in step 5, the results are obtained.

There are two functions, *p_gen* and *n_gen*, for generating candidates, and the measure of interestingness, *im*, in the procedure *NSP*, we describe them in subsection 3.1 and subsection 3.2, respectively.

```
Procedure:  n_gen(LP_{k-1}, LN_{k-1})
Parameters:
 LP_{k-1} : Large positive sequences with length  k-1
 LN_{k-1} : Large negative sequences with length  k-1
Output:
 CN_k : Negative sequence Candidates
Method:
// Generating new candidates
(1) for each sequence  p =< p_1, p_2, ..., p_{k-2}, p_{k-1} >
      in  LP_{k-1}  do
(2)   for each sequence
        q =< q_1, q_2, ..., q_{k-2}, ¬q_{k-1} >  in  LN_{k-1}  do
(3)     if  ((p_{j+1} = q_j), for all j = 1..k-2)  then
(4)       begin
(5)         new =< p_1, p_2, ..., p_{k-1}, ¬q_{k-1} >
(6)         CN_k = CN_k ∪ {new}
(7)       end
// Pruning redundant candidates
(8) CN_k = CN_k -
          { c | c ∈ CN_k  and any (k-1)-
          subsequence of  c  ∉ LN_{k-1} }
return  CN_k ;
```

Fig. 2. The procedure *n_gen*

3.1 Candidates Generation

The function, *p_gen*(), for generating candidates of positive sequences includes two phases: the first to generate new candidates and the second to prune redundant candidates [1]. In the first phase, the candidates of *k*–sequences are generated from the set of large positive (*k-1*)-sequences join with itself. For example, two candidates, $< s_1, s_2,...,s_{n-2},a_{n-1} ,b_{n-1} >$ and $< s_1, s_2,...,s_{n-},, b_{n-1} , a_{n-1} >$, are generated by combining two positive sequence, $< s_1, s_2, ..., s_{n-2}, a_{n-1} >$ and $< s_1,s_2 ...,s_{n-2},b_{n-1} >$. In the second phase, the candidates of positive *k*-sequences that contain any infrequent (*k-1*)-subsequence will be deleted. This is because the apriori-principle states the fact that *any super-pattern of an infrequent pattern cannot be frequent*.

The function, *n_gen*(), for generating candidates of negative sequences is shown in fig. 2. It includes two phases: the first to generate new candidates and the second to prune redundant candidates. In the first phase, the candidates of *k*–sequences are generated from the set of large positive (*k-1*)-sequences join with the set of large negative (*k-1*)-sequences. Note that, in *n_gen*(), the way to combine two sequences to generate a candidate of negative sequence is slightly different from *p_gen*(). For example, the candidate of negative sequence, $< a_1, s_2, ..., s_{n-1}, \neg b_{n-1} >$, is generated by combining the positive sequence $< a_1, s_2, ..., s_{n-1}, >$ and the negative sequence $< , s_1, ..., s_{n-2}, \neg b_{n-1} >$. In the second phase, candidates of negative *k*-sequences containing any infrequent (*k-1*)-subsequence will be deleted.

3.2 Measure of Interestingness

There may be a huge number of sequences generated during sequential pattern mining, and most of them are uninteresting. Therefore, defining a function to measure the degree of interestingness of a sequence is needed.

Suppose that $s =< s_1 ... s_n >$ (or $< s_1 ... \neg s_n >$), the preceding subsequence, s_{pre}, is $< s_1 ... s_{n-1} >$, the target subsequence, s_{tar}, is $< s_n >$ (or $< \neg s_n >$). And each s_k is a code, (i.e., a code represents a item) mapped from a fuzzy itemset (see step 3 in algorithm NFSPM). We define the measure of interestingness of sequence *s* as following equation:

$$im(s) = fsupp(s) / supp(s_{pre}) - fsupp(s_{tar}) \qquad (1)$$

Note that *fsupp*(*s*) is calculated from the membership values of original fuzzy itemset of s_k, and *supp*(*s*) is calculated by counting the number of the transactions support the sequence s.

3.3 Procedure NSP

The algorithm NSP is an iterative procedure as shown in fig. 2. In the algorithm, the iteration contains two phases: the phase of positive sequential pattern mining (line 5-6), and the phase of negative sequential pattern mining (line 7-10).

In the positive sequential pattern mining phase, the candidates of positive sequences with length k, CP_k, are generated from LP_{k-1} join with LP_{k-1} by p_gen function (line 5). Next, large k-sequences, LP_k, are selected if their supports are greater than or equal to a user-predefined threshold (line 6).

In the negative sequential pattern mining phase, the candidates of negative sequences with length k, CN_k, are generated from LP_{k-1} join with LN_{k-1} by n_gen function (line 7), Next, large negative sequences LN_k are selected if their supports are greater than or equal to a user-predefined threshold (line 8). Then, negative sequential patterns with high degree of interestingness, IN_k, are selected if their *im* are greater than or equal to a user-predefined threshold (line 9). Finally, IN_k are added into N, which contains all negative patterns with high degree of interestingness have already been mined so far (line10).

Procedure: *NSP*()
Input:
FTD : Fuzzy sets transformed from transaction database
LP_1 : 1-large positive sequences
λ_{ps} : Threshold of support of positive sequences
λ_{ns} : Threshold of support of negative sequences
λ_{ni} : Threshold of interestingness of negative sequences
Output:
N : Strong negative sequential patterns
Method:
(1) $LN_1 = \{<\neg i>) \mid i \in LP_1\}$
(2) $N = \phi$
(3) **for** ($k = 2$; $LP_{k-1} \neq \phi$; $k++$) **do**
(4) **begin**
// **Mining Positive sequential patterns**
(5) $CP_k = p\_gen(LP_{k-1})$
(6) $LP_k = \{<c>) \mid c \in CP_k,\ fsupp(c) \geq \lambda_{ps}\}$
// **Mining Negative sequential patterns**
(7) $CN_k = n\_gen(LP_{k-1}, LN_{k-1})$
(8) $LN_k = \{<c>) \mid c \in CN_k,\ fsupp(c) \geq \lambda_{ns}\}$
(9) $IN_k = \{<l>) \mid l \in LN_k, im(l) \geq \lambda_{ni}\}$
(10) $N = N \cup IN_k$
end
return N ;

Fig. 3 The procedure *NSP*

4 Example

Suppose a customer sequence database is given as shown in Table 1. Each row includes a CID (customer ID) and a customer's Purchase sequences. Each item in a sequence is represented by the form (item:quantity). The fuzzy membership functions for the fuzzy sets of items are shown in fig. 4.

Table 1. Transaction database

CID	Purchase sequences (item: quantity)
1	< (*A*: 12), (*C*: 18), (*A*: 13) >
2	< {(*B*: 18), (*C*: 20)}, {(*A*: 3), (*D*: 2)} >
3	< (*B*: 19), {(*A*: 2), (*D*: 3)}, (*C*: 2) >
4	< (*C*: 18), (*B*: 20), {(*A*: 3), (*D*: 2)} >
5	< (*C*: 17), (*B*: 20), (*E*: 10) >

The threshold of the *fsupp* of fuzzy positive sequences, λ_{ps}, the threshold of the *fsupp* of fuzzy negative sequences, λ_{ns}, and the threshold of *im* of negative fuzzy sequences, λ_{ni} are set to be 0.2, 0.4, and 0.4, respectively. The process of the algorithm is shown in table 2 to table 10. The discovered strong positive and negative sequential patterns are shown in table 11.

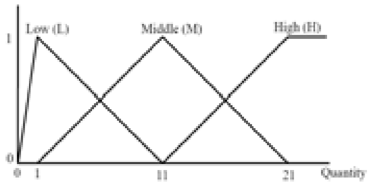

Fig. 4 The membership functions

In table 2, each item in transaction database is transform into fuzzy item.

Table 2. Fuzzy sets transformed from table 1

CID	Fuzzy sets (L: Low, M: Middle, H: High)
1	$(\frac{0.9}{A.M}+\frac{0.1}{A.H}),(\frac{0.3}{C.M}+\frac{0.7}{C.H}),(\frac{0.8}{A.M}+\frac{0.2}{A.H})$
2	$\{(\frac{0.3}{BM}+\frac{0.7}{BH}),(\frac{0.1}{CM}+\frac{0.9}{CH})\},\{(\frac{0.8}{AL}+\frac{0.2}{AM}),(\frac{0.9}{DL}+\frac{0.1}{DM})\}$
3	$(\frac{0.2}{BM}+\frac{0.8}{BH}),\{(\frac{0.9}{AL}+\frac{0.1}{AM}),(\frac{0.8}{DL}+\frac{0.2}{DM})\},(\frac{0.9}{CL}+\frac{0.1}{CM})$
4	$(\frac{0.3}{CM}+\frac{0.7}{CH}),(\frac{0.1}{BM}+\frac{0.9}{BH}),\{(\frac{0.8}{AL}+\frac{0.2}{AM}),(\frac{0.9}{DL}+\frac{0.1}{DM})\}$
5	$(\frac{0.4}{CM}+\frac{0.6}{CH}),(\frac{0.1}{BM}+\frac{0.9}{BH}),(\frac{0.1}{EL}+\frac{0.9}{EM})$

In table 3 and table 4, all candidates of fuzzy itemsets, and their fuzzy support (*fsupp*) are list. The large fuzzy itemsets are marked with boldface.

Table 3. 1-fuzzy itemsets

candidates	*fsupp*	candidates	*fsupp*
A.L	**0.50**	***D.L***	**0.52**
A.M	**0.28**	*D.M*	0.08
A.H	0.04	*D.H*	0.00
B.L	0.00	*E.L*	0.02
B.M	0.14	*E.M*	0.18
B.H	**0.66**	*E.H*	0.00
C.L	0.18		
C.M	**0.24**		
C.H	**0.58**		

Table 4. 2-fuzzy itemsets

candidate	*fsupp*
B.H,C.M	0.02
B.H,C.H	0.14
A.L,D.L	**0.48**
A.M,D.L	0.10

In table 5, each of large fuzzy itemsets is recoded as a unique integer.

Table 5. Codes of large fuzzy itemsets in table 3 and table4

fuzzy itemset	Code
A.L	1
A.M	2
B.H	3
C.M	4
C.H	5
D.L	6
A.L,D.L	7

In table 6, the codes mapped from fuzzy itemsets are set to be the 1-large positive sequences, LP_1. Then, 1-large negative sequences, LN_1 are initialed by copying from LP_1. And all of their *fsupp* are listed.

Table 6. 1-sequences

LP_1	*fsupp*	LN_1	*fupp*
<1>	0.50	<¬1>	0.50
<2>	0.28	<¬2>	0.72
<3>	0.66	<¬3>	0.34
<4>	0.24	<¬4>	0.76
<5>	0.58	<¬5>	0.42
<6>	0.52	<¬6>	0.48
<7>	0.48	<¬7>	0.52

Now, the procedure *NSP* is performed to find the negative sequential patterns. In table 7, the candidates of 2-positive sequences (CP_2) generated from the joint of LP_1 and LP_1, and their *fsupp* are listed. The large positive sequences, LP_2, (i.e., their *fsupp* are greater than or equal to the threshold λ_{ps}) are marked with boldface.

Table 7. 2-positive sequences

CP_2	*fsupp*	CP_2	*fsupp*	CP_2	*fsupp*
<1,1>	0	<3,4>	0.02	**<5,7>**	**0.30**
<1,2>	0	<3,5>	0	<6,1>	0
<1,3>	0	**<3,6>**	**0.52**	<6,2>	0
<1,4>	0.02	**<3,7>**	**0.46**	<6,3>	0
<1,5>	0	<4,1>	0.08	<6,4>	0.02
<1,6>	0	<4,2>	0.12	<6,5>	0
<1,7>	0	<4,3>	0.14	<6,6>	0
<2,1>	0	<4,4>	0	<6,7>	0
<2,2>	0.04	<4,5>	0	<7,1>	0
<2,3>	0	<4,6>	0.08	<7,2>	0
<2,4>	0	<4,7>	0.08	<7,3>	0
<2,5>	0	<5,1>	0.03	<7,4>	0.02
<2,6>	0	**<5,2>**	**0.22**	<7,5>	0
<2,7>	0	**<5,3>**	**0.26**	<7,6>	0
<3,1>	**0.46**	<5,4>	0	<7,7>	0
<3,2>	0.1	<5,5>	0		
<3,3>	0	**<5,6>**	**0.32**		

In table 8, the candidates of 2-negative sequences (CN_2) generated from the joint of LP_1 and LN_1, their *fsupp* and *im* are listed. The large negative sequences, LN_2, (i.e., their *fsupp* are greater than or equal to the threshold, λ_{ns}) are marked with boldface. And the large negative sequences with high degree of interestingness, IN_2, (i.e., their *im* are greater than or equal to the threshold, λ_{ni}) are underlined.

Table 8. 2-negative sequences

CN_2	*fsupp*	*im*	CN_2	*fsupp*	*im*	CN_2	*fsupp*	*im*
<1,¬1>	**0.5**	0.33	**<3,¬4>**	**0.66**	0.06	<5,¬7>	0.34	-0.1
<1,¬2>	**0.5**	0.11	**<3,¬5>**	**0.54**	0.26	**<6,¬1>**	**0.52**	0.37
<1,¬3>	**0.5**	**0.49**	<3,¬6>	0.26	-0.16	**<6,¬2>**	**0.52**	0.15
<1,¬4>	**0.5**	0.07	<3,¬7>	0.30	-0.15	**<6,¬3>**	**0.52**	**0.53**
<1,¬5>	**0.5**	**0.41**	<4,¬1>	0.22	-0.28	**<6,¬4>**	**0.52**	0.11
<1,¬6>	**0.5**	0.35	<4,¬2>	0.22	-0.5	**<6,¬5>**	**0.52**	**0.45**
<1,¬7>	**0.5**	0.31	<4,¬3>	0.14	-0.2	**<6,¬6>**	**0.52**	0.39
<2,¬1>	0.28	-0.15	<4,¬4>	0.24	-0.52	**<6,¬7>**	**0.52**	0.35
<2,¬2>	0.14	-0.55	<4,¬5>	0.24	-0.18	**<7,¬1>**	**0.48**	0.3
<2,¬3>	0.28	0.01	<4,¬6>	0.20	-0.28	**<7,¬2>**	**0.48**	0.08
<2,¬4>	0.28	-0.41	<4,¬7>	0.22	-0.3	**<7,¬3>**	**0.48**	**0.46**
<2,¬5>	0.28	-0.07	<5,¬1>	0.34	-0.08	**<7,¬4>**	**0.48**	0.04
<2,¬6>	0.28	-0.13	**<5,¬2>**	**0.46**	-0.15	**<7,¬5>**	**0.48**	0.38
<2,¬7>	0.28	-0.17	<5,¬3>	0.36	0.11	**<7,¬6>**	**0.48**	0.32
<3,¬1>	0.28	-0.15	**<5,¬4>**	**0.58**	-0.04	**<7,¬7>**	**0.48**	0.28
<3,¬2>	**0.64**	0.08	**<5,¬5>**	**0.58**	0.31			
<3,¬3>	**0.66**	**0.49**	**<5,¬6>**	**0.48**	0.12			

In table 9, the candidates of 3-positive sequences (CP_3) generated from the joint of LP_2 and LP_2, and their *fsupp* are listed. Note that no more 3-large positive sequences, LP_3, are generated because all of the *fsupp* of CP_3 are less than the threshold λ_{ps}. Since no candidate of 4-positive sequence can be generated from LP_3 (LP_3 is null), we stop mining positive pattern here.

Table 9. 3-positive sequences

CP_3	*fsupp*
<5,3,6>	0.14
<5,3,7>	0.14

In table 10, the candidates of 3-negative sequences (CN_3) generated from the joint of LP_2 and LN_2, their *fsupp* and *im* are listed. The large negative sequences, LN_3, are marked with boldface. And the large negative sequences with high degree of interestingness, IN_3, are underlined. Because no candidate of 4-positive sequence can be generated from LN_3 and LP_3 (LP_3 is null), we stop the procedure *NSP*.

Table 10. 3-negative sequences

CN_3	*fsupp*	*im*	CN_3	*fsupp*	*im*
<3, 1, ¬2>	**0.46**	0.05	<5, 3, ¬2>	0.26	-0.07
<3, 1, ¬3>	**0.46**	**0.43**	<5, 3, ¬4>	0.26	-0.11
<3, 1, ¬4>	**0.46**	0.01	<5, 3, ¬5>	0.26	0.23
<3, 1, ¬5>	**0.46**	0.35	<5, 6, ¬2>	0.32	0.08
<3, 6, ¬2>	**0.48**	0.08	<5, 6, ¬4>	0.32	0.04
<3, 6, ¬3>	**0.48**	**0.46**	<5, 6, ¬5>	0.32	0.38
<3, 6, ¬4>	**0.48**	0.04	<5, 6, ¬6>	0.32	0.32
<3, 6, ¬5>	**0.48**	0.38	<5, 7, ¬2>	0.3	0.03
<3, 7, ¬2>	**0.46**	0.05	<5, 7, ¬4>	0.3	-0.01
<3, 7, ¬3>	**0.46**	**0.43**	<5, 7, ¬5>	0.3	0.33
<3, 7, ¬4>	**0.46**	0.01	<5, 7, ¬6>	0.3	0.27
<3, 7, ¬5>	**0.46**	0.35			

Finally, the codes in discovered negative sequential patterns are mapped back to the original fuzzy itemsets. And the result is listed in table 11.The algorithm NFSM is stopped here.

Table 11. The discovered negative fuzzy sequential patterns

Fuzzy sequential patterns	*fsupp*	*im*
$<(A.L),(\neg B.H)>$	0.5	0.49
$<(A.L),(\neg C.H)>$	0.5	0.41
$<(B.H),(\neg B.H)>$	0.66	0.49
$<(D.L),(\neg B.H)>$	052	0.53
$<(D.L),(\neg C.H)>$	0.52	0.45
$<\{(A.L),(D.L)\},(\neg B.H)>$	0.48	0.46
$<(B.H),(A.L),(\neg B.H)>$	0.46	0.43
$<(B.H),(D.L),(\neg B.H)>$	0.48	0.46
$<(B.H),\{(A.L),(D.L)\},(\neg B.H)>$	0.46	0.43

5 Conclusion

The major challenges in mining sequential patterns, especially fuzzy negative ones, are that there may be huge number of the candidates generated, and most of them are meaningless. In this paper, we proposed a method, NFSPM, for mining negative fuzzy sequential patterns. In our method, the absences of itemsets in sequences are also considered. Besides, only the sequences with high degree of interestingness can be selected as negative fuzzy sequential patterns. The result showed that NFSPM could prune a lot of redundant candidates, and could extract meaningful sequential patterns from a large number of frequent sequences.

References:

[1] R. Agrawal and R. Srikant, Mining Sequential Patterns, *Proceedings of the Elventh International Conference on Data Engineering*, Taipei, Taiwan, March, 1995, pp. 3-14.

[2] R. Srikant and R. Agrawal, Mining Sequential Patterns: Generalizations and Performance Improvements, *Proceedings of the Fifth International conference, Extending Database Technology (EDBT'96)*, 1996, pp. 3-17.

[3] J. Han, G. Dong, Y. Yin, Efficient Mining of Partial Periodic Patterns in Time Series Database, *Proceedings of Fifth International Conference on Data Engineering*, Sydney, Australia, IEEE Computer Society, 1999, pp.106-115.

[4] F. Masseglia, F. Cathala, P. Ponelet, The PSP Approach for Mining Sequential Patterns, *Proceeding of the Second European Symposium on Principles of Data Mining and Knowledge Discovery*, Vol. 1510, 1998, pp. 176-184.

[5] J. S. Park, M. S. Chen, P. S. Yu, An Effective Hash Based Algorithm for Mining association rule, *Proceeding of the ACM SIGMOD Conference on management of data*, 1995, pp. 175-186.

[6] J. Pei, B. Motazavi-Asl, H. Pinto, Q. Chen, U. Dayal, M-C. Hsu, Prefixsapn Mining Sequential Patterns Efficiently by Prefix Prejected Pattern Growth, *Proceeding of the International Conference of Data Engineering*, 2001, pp. 215-224.

[7] R. Srikant, R. Agrwal, Mining Assoiation Rules with Item Constraints, *Proceedings of the Third International Conference on Knowledge Discovery in Database and Data Mining*, 1997.

[8] M. J. Zaki, Efficient Enumeration of Frequent Sequences, *Proceedings of the Seventh CIKM,* 1998.

[9] J. Ayres, J. E. Gehrke, T. Yiu, and J. Flannick, Sequential Pattern Mining Using Bitmaps, *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. Edmonton, Alberta, Canada, July 2002.

[10] X. Yan, J. Han, and R. Afshar, CloSpan: Mining Closed Sequential Patterns in Large Datasets, *Proceedings of 2003 SIAM International Conference Data Mining (SDM'03)*, 2003, pp. 166-177.

[11] M. Zaki, SPADE: An Efficient Algorithm for Mining Frequent sequences, *Machine Learning*, vol. 40, 2001, pp. 31-60.

[12] M. Zaki, Efficient Enumeration of Frequent Sequences, *Proceedings of the Seventh International Conference Information and Knowledge Management (CIKM'98),* 1998, pp. 68-75.

[13] T. Hong, K. Lin and S. Wang, Mining fuzzy sequential patterns from multiple-items transactions, *Proceedings of the Joint ninth IFSA World Congress and twentieth NAFIPS International Conference,* 2001, pp. 1317-1321.

[14] R.-S. Chen, G.-H. Tzeng, C.-C. Chen, and Y.-C. Hu, Discovery of fuzzy sequential patterns for fuzzy partitions in quantitative attributes, *ACS/IEEE International Conference on Computer Systems and Applications (AICCSA),* 2001, pp. 144-150.

[15] Y.-C. Hu, R.-S. Chen, G.-H. Tzeng, and J.-H. Shieh, A fuzzy data mining algorithm for finding sequential patterns, *International Journal of Uncertainty Fuzziness Knowledge-Based Systems,* 2003, vol. 11, no.2, pp. 173-193.