Filomat 32:5 (2018), 1875–1885
https://doi.org/10.2298/FIL1805875Q

Published by Faculty of Sciences and Mathematics,
University of Niš, Serbia
Available at: http://www.pmf.ni.ac.rs/filomat

# Mining negative sequential patterns from infrequent positive sequences with 2-level multiple minimum supports

**Ping Qiu[a], Long Zhao[a], Weiyang Chen[a], Tiantian Xu[a], Xiangjun Dong[a]**

[a] *School of Information, Qilu University of Technology (Shandong Academy of Sciences), Jinan, China*

**Abstract.** Negative sequential patterns (NSP) referring to both occurring items (positive items) and non-occurring items (negative items) play a very important role in many real applications. Very few methods have been proposed to mine NSP and most of them only mine NSP from frequent positive sequences, not from infrequent positive sequences (IPS). In fact, many useful NSP can be mined from IPS, just like many useful negative association rules can be obtained from infrequent itemsets. e-NSPFI is a method to mine NSP from IPS, but its constraint is very strict to IPS and many useful NSP would be missed. In addition, e-NSPFI only uses a single minimum support, which implicitly assumes that all items in the database are of the similar frequencies. In order to solve the above problems and optimize NSP mining, a 2-level multiple minimum supports (2-LMMS) constraint to IPS is proposed in this paper. Firstly, we design two minimum supports constraints to mine frequent and infrequent positive sequences. Secondly, we use Select Actionable Pattern (SAP) method to select actionable NSP. Finally, we propose a corresponding algorithm msNSPFI to mine actionable NSP from IPS with 2-LMMS. Experiment results show that msNSPFI is very efficient for mining actionable NSP.

## 1. Introduction

Negative sequential patterns (NSP) considering both occurring items and non-occurring items (also called positive and negative items) play a very important role in many intelligent systems and applications [1][2][3][4][5][6]. For instance, $nsp =< xy\neg zF >$ is a negative sequential patterns, where $x$, $y$ and $z$ stand for drug codes, and $F$ stands for disease status. $nsp$ indicates that patients who usually receive medical services $x$ and $y$ but NOT $z$ are likely to have disease status $F$ rather than the others. Such situation cannot be expressed by the identification of PSP alone. So NSP is a more accurate way to assist doctors in arranging the follow-up treatment of patients.

Very few methods have been proposed to mine NSP and most of them are very inefficient because they calculate the negative sequential candidates' (NSC') supports by re-scanning the database [7][8][9][10][11][12]. Furthermore, most of them mine NSP only from frequent positive sequences (also called positive sequential patterns), and do not consider infrequent positive sequences (IPS). However, many useful NSP can

2010 *Mathematics Subject Classification*. Primary 68T10
*Keywords*. Negative sequential patterns, infrequent positive sequences, multiple minimum supports, actionable
Received:27 October 2017; Accepted: 30 January 2018
Communicated by Hari M. Srivastava
Corresponding authors are Tiantian Xu and Xiangjun Dong
Research supported by National Natural Science Foundation of China (71271125, 61502260), Shandong Natural Science Foundation, China (ZR2018MF011)
*Email addresses:* qiupcl@163.com (Ping Qiu), zxcvbnm9515@163.com (Long Zhao), weiyangchen@yeah.net (Weiyang Chen), xtt-ok@163.com (Tiantian Xu), d-xj@163.com (Xiangjun Dong)

be mined from IPS, just like many useful negative association rules or negative frequent itemsets can be mined from infrequent itemsets (inFIS) [13][14][15]. If no constraint is added, the number of IPS is very large and many of them are meaningless [5]. How to guarantee the number of IPS in a suitable degree is very challengeable. Although several algorithms have been proposed to mine negative association rules from inFIS, no algorithm has been proposed to mine NSP from IPS except e-NSPFI. e-NSPFI[5] proposed a constraint to control the number of IPS and the constraint was defined as follows: the support of any $k$-length $ips$, denoted by $sup(ips)$, is less than minimum support ($ms$) threshold, but the supports of all ($k$-1)-length subsequences of $ips$ is no less than $ms$. For example, a dataset is given as follows:10 :< $abcad$ >; 20 :< $acad$ >; 30 :< $bcd$ >; 40 :< $acb$ >; 50 :< $adcd$ >, we assume $ms$ = 2. According to the existing PSP mining algorithms, $s_1$ =< $abc$ > and $s_2$ =< $abcd$ > are infrequent sequences because of $sup(s_1) = sup(s_2) = 1 < ms$. $s_3$ =< $ab$ >, $s_4$ =< $ac$ > and $s_5$ =< $bc$ > are frequent sequences, because $sup(s_3) = 2$, $sup(s_4) = 4$ and $sup(s_5) = 2$ are not less than $ms$. $s_2$ does not satisfy the infrequent constraint of e-NSPFI, because the infrequent sequence $s_1$ is the 3-length subsequence of 4-sequence $s_2$. However, $s_2$ can also generate NSP, such as< $a\neg bc\neg d$ >. Although e-NSPFI can efficiently mine NSP from such IPS, it has the following problems.

(1) Too strict constraint. The constraint on IPS in e-NSPFI is too strict and many useful NSP would be missed. Take $s_2$ in the above example for instance, a NSC $nsc$ =< $a\neg bc\neg d$ > can be generated and $sup(nsc) = 3$. So nsc is a frequent negative sequence, but it cannot be mined by e-NSPFI.

(2) Single minimum support. e-NSPFI only uses single ms to constrain IPS, which implies that all items in the database are of the same frequencies [6]. This is not the case in many applications.To solve this problem, the concept of multiple minimum supports (MMS) has been proposed in a few methods, such as MS-GSP [16], MS-PS[16], CPNFSP [17] and E-msNSP [6]. They assign different items with different minimum supports.That is, each item has its own minimum item support (MIS). Among them, MS-GSP and MS-PS only mined PSP with MMS. CPNFSP only defined three forms of NSP, i.e., $(\neg A, B)$, $(A, \neg B)$ and $(\neg A, \neg B)$ with MMS. E-msNSP only mined NSP from PSP with MMS, not mined NSP from IPS. We have not found any literatures to mine NSP from IPS with MMS until now.

To solve the above problems, we propose an efficient algorithm, named msNSPFI to mine NSP from IPS with 2-level MMS (2-LMMS). The main contributions of this paper can be summarized as follows:

Firstly, we propose a 2-LMMS constraint to reduce the number of IPS. That is, we assign two $ms$ constraints to each item to mine PSP and IPS. 2-LMMS constraint give a good solution to the above two problems.

Secondly, we use SAP method [2] to select actionable NSP.

Finally, we propose a corresponding algorithm msNSPFI to mine actionable NSP from IPS with 2-LMMS.

The remainder of the paper is organized as follows. Section 1 discusses the related work. Section 2 is preliminary. Section 3 proposes msNSPFI algorithm. Section 4 presents experimental results. Conclusions are described in Section 5.

## 2. Related work

In this section, some related studies about NSP mining, useful information mining from infrequent patterns and frequent patterns mining with MMS are briefly reviewed.

### 2.1. Negative Sequential Patterns Mining

NegGSP is an algorithm to mine NSP [8]. It calculates the NSC's supports by re-scanning the database and generates NSP by using a single minimum support. PNSP is an algorithm to mine both PSP and NSP [7]. It generates NSC by concatenating positive and negative itemsets and compares the supports of NSC with *ms* to generate NSP. GA algorithm obtains a generation by crossover and mutation operations, which avoids NSC generation [11]. NSPM only deals with the last element in the NSP [18]. The method in [19] only defines three forms of NSP, i.e., $(\neg A, B)$, $(A, \neg B)$ and $(\neg A, \neg B)$ and it requires $A \cap B = \emptyset$. This is a general requirement in negative association rule mining, but it is very strict in NSP mining [1].

E-NSP algorithm is the most efficient method to mine NSP [4]. It only uses equations to calculate the NSC's supports, so as to improve the efficiency of the algorithm. SAPNSP and SAP are two improved versions of e-NSP to select actionable PSP and NSP [2][3]. SAPNSP improves Wu's method [20] to analyze the correlation between elements in a sequential pattern. SAP uses the correlation coefficient to mine the actionable PSP and NSP. SAPBN [21] uses Bayesian network (BN) to find the actionable PSP and NSP. In this paper, SAP method is used to find actionable NSP. f-NSP uses bitset structure to effectively improve the time efficiency of e-NSP algorithm[22]. e-RNSP mine repetitive negative sequence patterns[23] and HUNSPM can mine high utility NSP[12].

### 2.2. Useful Information Mining from Infrequent Patterns

The concept of multiple level minimum supports (MLMS) is first proposed in mining frequent itemsets[24]. It assigns different ms for itemsets with different length. Suppose $ms(k)$ is the *ms* of $k$-itemsets ($k = 1, 2, \ldots, n$), $ms(1)ms(2), \ldots, ms(n)ms > 0$, the frequent itemsets (FIS) and inFIS are as follows. For a $k$-itemset $X$, if $sup(X) \geq ms(k)$, then $X$ is a FIS; and if $ms(k) > sup(X) \geq ms$, then $X$ is an inFIS. The positive and negative association rules can be mined from those FIS and inFIS discovered by MLMS model [23]. A 2-Level Supports model is proposed in [25] to discover FIS and inFIS. 2-Level Supports model uses two level supports $ms-FIS$ and $ms-inFIS$ ($ms-FIS \geq ms-inFIS > 0$) to constrain the FIS and inFIS respectively. For any itemset $B$, if $s(B) \geq ms - FIS$, then $B$ is a FIS; and if $ms - FIS > s(B) \geq ms - inFIS$, then $B$ is an inFIS. e-NSPFI is a method to mine NSP from both PSP and IPS [5]. But it requires that any subsequence of IPS should be frequent. This is a very strict requirement in sequential patterns mining.

### 2.3. Frequent patterns mining with MMS

Liu proposes MS-GSP and MS-PS methods to find all PSP with MMS based on breadth-first search and depth-first search [16]. In MS-GSP, a MIS value of each item is assigned by users. It generates PSP by re-scanning the database. MS-PS generates PSP by building projected database. Hu, et al. proposes a preorder linked multiple supports tree to store and compress the sequence database [26]. Huang proposes a model to find fuzzy quantitative PSP with MMS [27]. E-msNSP mines NSP with MMS from IPS, but it does not mine NSP from IPS [6].

## 3. Preliminary

In sequential patterns mining, positive sequences only consist of positive items and negative sequences consist of both positive items and negative items.

### 3.1. Positive Sequential Patterns-PSP

Let $I = \{i_1, i_2, \ldots, i_n\}$ be a set of items. An itemset is a subset with distinct items. A sequence $s =< e_1e_2 \ldots e_l >$ is an ordered list of itemsets (elements), where $e_k = (i_1i_2 \ldots i_h)$ is an element ($e_k \subseteq I(1 \leq j \leq l), i_j \in I(1 \leq j \leq h)$). For simplicity, the bracket is omitted if an element only contains one item.

The size of $s$ is the number of elements in a sequence $s$, which is denoted as $size(s)$. When $size(s) = k$, $s$ is called a $k$-size sequence. The length of $s$ is the number of items in $s$, which is denoted as $length(s)$. When $length(s) = m$, $s$ is called a $m$-length sequence. For instance, a sequence $s =< (ab)(cde)f >$ consists of three elements $(ab)$, $(cde)$ and $f$. Therefore, $s$ is a 3-size and 6-length sequence, i.e., $size(s) = 3$ and $length(s) = 6$.

Sequence $s_\alpha =< \alpha_1\alpha_2 \ldots \alpha_n >$is a sub-sequence of sequence $s_\beta =< \beta_1\beta_2 \ldots \beta_m >$ and $s_\beta$ is a super-sequence of $s_\alpha$, if $1 \leq j_1 < j_2 < \ldots < j_n \leq m$, $\alpha_1 \subseteq \beta_{j_1}, \alpha_2 \subseteq \beta_{j_2}, \ldots, \alpha_n \subseteq \beta_{j_n}$, denoted as $s_\alpha \subseteq s_\beta$. We also say that $s_\beta$ contains $s_\alpha$. For example, $< (ab) >$ and $< acf >$ are the subsequences of $< (ab)(cde)f >$.

A database D is a set of tuples <*sid*, *ds*>, where *sid* is the identity number of *ds* and *ds* is a sequence in D. The number of tuples in D is denoted as |D|. The set of tuples containing s denoted as $\{< s >\}$, is the support of $s$. That is, $sup(s) = |\{<s>\}| = |\{<sid, ds>, <sid, ds> \in D \wedge (s \subseteq ds)\}|$.

### 3.2. Negative Sequential Patterns-NSP

#### 3.2.1. Two constraints

The number of NSC is often large, but many of them are meaningless. In this paper, we use two constraints to limit the number of NSC[1].

Constraint 1 (Format constraint). Continuous negative elements in a NSC are not allowed. For instance, $< \neg(xy)(cde)\neg f >$satisfies Constraint 1, but $< \neg(xy)\neg(cde)f >$does not.

Constraint 2 (Negative element constraint). The smallest negative unit in a NSC is an element. For instance, $< \neg(xy)(cde)f >$ satisfies constraint 2, but $< (\neg xy)(cde)f >$does not.

#### 3.2.2. The definition of negative containment

The definition of negative containment is very important to mine NSP because it affects the speed to calculate the NSC's supports. We use the same definition as e-NSP [1]. Before introducing the definition of negative containment, we first give an important related definition Positive Partner.

Definition 1 (*Positive Partner*). The positive partner of a negative element $\neg a$ is $a$, which is denoted as $p(\neg a)$, i.e., $p(\neg a) = a$. The positive partner of positive element $a$ is a itself, i.e., $p(a) = a$. The positive partner of a negative sequence $ns =< s_1 \ldots s_k >$ can be obtained by converting all negative elements to their positive partners, which is denoted as $p(ns)$, i.e.,$p(ns) = \{< s'_1 \ldots s'_k > | s'_i = p(s_i), s_i \in ns\}$. For instance, $p(< \neg(xy)z\neg x >) =< (xy)zx >$.

Given a positive sequence $ds =< a(bc)d(cde) >$ and a negative sequence $ns =< a\neg bb\neg a(cde) >$, ds contains $ns$ if and only if $ds$ contains $< ab(cde) >$ and $ds$ does not contain $p(< a\neg bb(cde) >) =< abb(cde) >$ and $p(< ab\neg a(cde) >) =< aba(cde) >$ respectively. The sequence $< ab(cde) >$ is the sub-sequence of $ns$ that contains all positive elements with the same order as $ns$, denoted as $MPS(ns)$. The sequence $< a\neg bb(cde) >$ (or $< ab\neg a(cde) >$) is the sub-sequence of $ns$ that contains all positive elements and only one negative element with the same order as $ns$, denoted as $1 - negMS$. The set consisting of all $1 - negMS$ in $ns$ is denoted as $1 - negMSS_{ns}$. For example, $1 - negMSS < a\neg bb\neg a(cde) >= < a\neg bb(cde) >, < ab\neg a(cde) >$.

To sum up, the definition of negative containment is as follows. Definition 2 (*Negative containment*). Given a data sequence $ds$ and a negative sequence $ns$, $ds$ contains $ns$ if and only if the two conditions hold:

(1) $MPS(ns) \subseteq ds$;

(2) $\forall 1 - negMS \in 1 - negMSS_{ns}, p(1 - negMS) \nsubseteq ds$.

## 4. msNSPFI Algorithm

This section consists of three parts. First, the definition of multiple level minimum supports (MLMS), including the scope of infrequent sequential patterns (IPS) and frequent sequential patterns, is proposed. Second, the steps of MLMS-NSP algorithm are given. Final, the pseudo code is given.

*4.1. Frequent and Infrequent Sequential Patterns Generation Strategy*

In this section, We use similar method proposed in [28] to assign MIS values to items, then give a formal definition of 2-level multiple supports that can constrain the number of PSP and IPS, and then the corresponding pseudo code is given.

*4.1.1. Assigning MIS values of items*

This paper uses the actual frequencies of the sequences including single item in a database to assign their MIS values [20]. The formulas can be stated as follows:

$$MIS(i) = \begin{cases} ms(i) & ms(i) > LS, \\ LS & otherwise \end{cases} \tag{1}$$

.

$$ms(i) = \beta * sup(< i >) \tag{2}$$

where $sup(< i >)$ is the actual support of the sequence including single item $i$ in the database. $LS$ is the user-specified lowest MIS value and $\beta$ ($0 \leq \beta \leq 1$) is a parameter that controls how the MIS values for items should be related to the supports of their corresponding single item sequences. In particular, when $\beta = 0$, all MIS are equal to $LS$, which is same as using single minimum support, i.e., $LS = ms$, to mine NSP; when $\beta = 1$ and $sup(< i >) \geq LS$, $sup(< i >) = MIS(i)$.

*4.1.2. 2-level multiple minimum supports-(2-LMMS)*

Definition 3 (*2-level multiple minimum supports*). For a sequence database, suppose it contains $n$ distinct items, denoted as $\{x_1, x_2 \dots x_n\}$.Two supports are set: one is the lower bound of minimum support, denoted as *lms*; the other is the upper bound of minimum support, denoted as *ums*. For *lms* and *ums*, we set $\beta$ and $LS$, $\beta'$ and $LS'$ respectively. *lms* and *ums* are defined as follows.

$$lms = \{MIS(x_1), MIS(x_2), \dots, MIS(x_n)\} \tag{3}$$

$$ums = \{MIS(x'_1), MIS(x'_2), \dots, MIS(x'_n)\} \tag{4}$$

where $MIS(x_1) < MIS(x'_1), MIS(x_2) < MIS(x'_2), \dots, MIS(x_n) < MIS(x'_n)$.

Definition 4 (*Minimum support of an element with MMS*) For a positive element a denoted as $(i_1 i_2 \dots i_m)$, the *ms* of a is the lowest MIS value of item $i_j$ ($MIS(i_j)$) ($1 \leq j \leq m$). For one negative element $\neg a$, its *ms* is defined as follows:

$$ms(\neg a) = 1 - ms(a) \tag{5}$$

For another negative element $\neg(xy)$, its *ms* is defined as follows:

$$ms(\neg(xy)) = 1 - ms(xy) = 1 - min[MIS(x), MIS(y)] \tag{6}$$

Definition 5 (*Minimum support of a negative sequence with MMS*). The *ms* of a negative sequence $s$ with *MMS* is the lowest *ms* value among the elements in the sequence. Suppose $s =< e_1 e_2 \dots e_r >$, its *ms* is defined as follows:

$$ms(s) = min[ms(e_1), ms(e_2), \dots, ms(e_r)] \tag{7}$$

For instance, given a sequence $s =< \neg(xy)(zxy)z >$, its ms is denoted as $ms(s) = min[ms(\neg(xy)), ms((zxy)), ms(z)]$, where $ms(\neg(xy)) = 1 - ms(xy)$.

Definition 6 (Frequent sequential patterns and infrequent sequential patterns) For sequence $s =< e_1 e_2 \dots e_r >$ in the sequence database, $lms(s) = min[MIS(e_1), MIS(e_2), \dots, MIS(e_r)]$ and $ums(s) = min[MIS(e'_1), MIS(e'_2), \dots, MIS(e'_r)]$.

If $s$ is a positive sequence and $sup(s) \geq ums(s)$, then $s$ is a PSP.

If $s$ is a positive sequence and $lms(s) \leq sup(s) < ums(s)$, then $s$ is a IPS.

If $s$ is a negative sequence and $sup(s) \geq ums(s)$, then $s$ is a NSP.

*4.1.3. The Pseudo Code of Generating lms or ums*

Because the process of generating *lms* and *ums* is the same, only the corresponding parameters $\beta$ and *LS* are different. So in the section, we just give the process of generating *lms*.

Algorithm 1.

Input: D: Sequence dataset $D$ and Parameter $\beta$ and $LS$;
Output: *lms*;
(1) For(Sequence Dataset){
(2) For (each item $x_i$){
(3) $ms(x_i) = \beta * sup(< x_i >)$;
(4) If($ms(x_i) > LS$){
(5) $MIS(x_i) = ms(x_i)$;
(6) }else{
(7) $MIS(x_i) = LS$;
(8) }
(9) $lms.add(MIS(x_i))$;
(10) }
(11) }

Form line 1 to line 7, the minimum item support of each item $x_i$ is calculated by equations (1) and (2).

Line 9 adds $MIS(x_i)$ to *lms*.

*4.2. Generating NSC*

In this section, the NSC are generated from PSP and IPS by using generation strategy in e-NSP. Its key process is introduced as follows.

For a $n$-size PSP, its NSC are generated by changing any $m$ non-contiguous elements to their negative elements, $m$ =1, 2, ..., $\lceil n/2 \rceil$, where $m$ =1, 2, ..., $\lceil n/2 \rceil$ and $\lceil n/2 \rceil$ is a minimum integer that is not less than $n/2$. For instance, for $< (xy)(zxy)z >$, its NSC include:

$m$=1, $< \neg(xy)(zxy)z >$, $< (xy)\neg(zxy)z >$, $< (xy)(zxy)\neg z >$;

$m$=2, $< \neg(xy)(zxy)\neg z >$.

*4.3. Calculating the Supports of NSC*

Given an $m$-size and $n$-neg-size negative sequence $ns$, for $\forall 1 - negMS_i \in 1 - negMSS_{ns} (1 \leq i \leq n)$, the support of $ns$ is:

$$sup(ns) = sup(MPS(ns)) - |\cup_{i=1}^{n} p(1 - negMS_i)| \tag{8}$$

where $p(1 - negMS_i)$ is a set of tuples $< sid, ds >$, denoted by $\{< sid, ds >, < sid, ds > \in D \wedge (p(1 - negMSi) \subseteq ds)\}$, and $\cup_{i=1}^{n}\{p(1 - negMS_i)\}$ is the union of all tuples $< sid, ds >$ of $1 - negMSS_{ns}$ and $|\cup_{i=1}^{n} \{p(1 - negMS_i)\}|$ is the number of $\cup_{i=1}^{n}\{p(1 - negMS_i)\}$.

If $ns$ only has one negative element, the support of $ns$ is:

$$sup(ns) = sup(MPS(ns)) - sup(p(ns)) \tag{9}$$

In particular, if $ns =< \neg i >$only contains one item, the support of $ns$ is:

$$sup(< \neg i >) = |D| - sup(< i >) \tag{10}$$

*4.4. SAP Method*

In this section, SAP method is used to select actionable NSP. The correlation coefficient is one of the important parameters of SAP, so the concept of correlation coefficient is given as follows:

Correlation coefficient can measure the relationships between two itemsets $\alpha$ and $\beta$ [2]. The equation of the correlation coefficient between $\alpha$ and $\beta$, denoted as $\rho(\alpha, \beta)$ ($-1 \leq \rho(\alpha, \beta) \leq +1$), is defined in equation 11.

$$\rho(\alpha, \beta) = \frac{sup(\alpha \cup \beta) - sup(\alpha)sup(\beta)}{\sqrt{sup(\alpha)(1 - sup(\alpha))sup(\beta)(1 - sup(\beta))}} \tag{11}$$

where $sup(*) \neq 0, 1$, and $\rho(\alpha, \beta)$ has three possible cases:

if $\rho(\alpha, \beta) < 0$, then $\alpha$ and $\beta$ are positively correlated;

if $\rho(\alpha, \beta) = 0$, then $\alpha$ and $\beta$ are independent;

if $\rho(\alpha, \beta) > 0$, then $\alpha$ and $\beta$ are negatively correlated.

The absolute value of $\rho(\alpha, \beta)$ represents the correlation strength of $\alpha$ and $\beta$. Therefore, we can set a minimum threshold $\rho_{min}$ to prune NSP with small correlation strength.

The main idea of SAP is as follows. We first need to test whether any 2-size subsequence of a NSP is actionable. If an $m$-size ($m > 1$) NSP $nsp =< e_1e_2 \dots e_m >$ is actionable, then we require that$< e_1e_2 >, < e_2e_3 >, \dots, < e_{m?1}e_m >$ are actionable too. Second, we use the equation 12 to test whether any 2-size sequence $< e_{i-1}e_i >$ is actionable.

In the equation 12, if the support of $< e_{i-1}e_i >$ is greater than the upper bound of minimum support ($ums$) and the equation 13 is 1, then $< e_{i-1}e_i >$ is actionable. The equation 14 can test the correlation between $e_{i-1}$ and $e_i$. Below we formally define the actionable NSP.

Definition 7 (*Actionable NSP*). An $m$-size ($m > 1$) NSP $nsp =< e_1e_2 \dots e_m >$is an actionable NSP if $\forall j \in \{2 \dots m\}$,

$$ansp(e_{j-1}, e_j) = sup(< e_{j-1}e_j >) \geq ums(f(e_{j-1}, e_j, ums, \rho_{min}) = 1), \tag{12}$$

then

$$f(e_{j-1}, e_j, ums, \rho_{min}) = \frac{sup(< e_{j-1}e_j >) + (e_{j-1}e_j) - (ums + \rho_{min}) + 1}{|sup(< e_{i-1}e_i >) - ums| + |\rho(e_{i-1}e_i) - \rho_{min}| + 1} \tag{13}$$

$$\rho(e_{j-1}, e_j) = \frac{sup(< e_{j-1}e_j >) - sup(< e_{j-1} >)sup(< e_j >)}{\sqrt{sup(< e_{j-1} > (1 - sup(< e_{j-1} >))sup(< e_j >)(1 - sup(< e_j >))}} \tag{14}$$

Corollary 1. An $m$-size ($m > 1$) NSP $nsp =< e_1e_2...e_m >$ is not an actionable NSP if $\exists j \in \{2...m\}$, $< e_{j-1}e_j >$ is not an actionable sequence pattern.

We can use Corollary 1 to prune meaningless NSP.

### 4.5. The Pseudo Code of msNSPFI Algorithm

The msNSPFI algorithm is presented in Algorithm 2.

Algorithm 2 msNSPFI

Input: D: Sequence Dataset; Parameters $lms$, $ums$ and $\rho_{min}$;
Output: actionable NSP;
(1) PSP and IPS=MS-GSP();
(2) For (each $sp$ in $PSPandIPS$){
(3) $NSC = e - NSP - NSC - Generation(sp)$;
(4) For (each $c$ in $NSC$){
(5) If ($size(c) = 1$){
(6) $sup(c) = |D| - sup(p(c))$;
(7) } else if ($size(c) > 1 and c.neg - size = 1$){
(8) $sup(c) = sup(MPS(c)) - sup(p(c))$;
(9) }
(10) else$sup(c) = sup(MPS(c)) - |\cup_{i=1}^{n} p(1 - negMS_i)|$;}
(11) If ($sup(c) \geq ums$){
(12) NSP.add(c);}}}

(13) For ((each $sp$ in $NSP \cup PSP \cup IPS$) and $size(sp) = 2$){
(14) $PSNP.add(sp)$;}
(15) For (each $nsp =< e_1e_2 \dots e_k > inNSP$){
(16) test $nsp$ with definition 7;
(17) If ($nsp$ is an actionable NSP){
(18) $ASP.add(nsp)$;
(19) }else{
(20) remove $nsp$ and all the patterns contain $nsp$ from $PNSP$;
(21) }}
(22) return $ASP$;

Line 1 finds PSP and IPS with MMS from the sequence database D by using improved MS-GSP methods.
Line 3 generates NSC from those PSP and IPS by e-NSP method.
From line 4 to 10, the support of $nsc$ is calculated by the above equations (5), (6) and (7).
From line 11 to 12, NSP are generated by comparing the support of each $nsc$ with $ums$.
From line 13 to 21, actionable NSP are generated by using the SAP method.
Line22 returns the results.

## 5. Experiments and Results

According to our main research contents, we design experiments to test (1) the number of NSP with different 2-LMMS and corresponding runtime, (2) the number of actionable NSP with different 2-LMMS and corresponding runtime, and (3) the scalability of msNSPFI.

All experiments are implemented in eclipse, running on Windows 10 PC with 32GB memory, Inter Core i7 3.4GHz CPU, all the programs are written in Java. In the experiments, the support of $s$ and the minimum support $ms$ are calculated in terms of the percentage of the frequency $| < s > |$ compared to $|D|$.

In this section, we do not do experiments to compare msNSPFI with other algorithms. Although E-msNSP and e-NSPFI are similar to msNSPFI, the two algorithms are not suitable for comparison because msNSPFI uses two MMS, $lms$ and $ums$, to constrain the number of IPS, and uses $ums$ to select the number of NSP. However, E-msNSP only uses one MMS to constrain PSP and NSP, and e-NSPFI is based on a single $ms$ to constrain NSP. Different models may yield different results. They mine NSP based on these results, so comparing these algorithms is not objective.

### 5.1. Datasets

All datasets are generated by IBM data generator [21]. Table 1 summarizes their characteristics.

$Dataset1(DS1)isC12.T10.S20.I10.DB1k.N100$, which contains 1k (DB) sequences. The number of items is 100 (N), the average number of elements in a sequence is 12 (C), the average number of items in an element is 10 (T), average length of maximal pattern consists of 20 (S) elements and each element is composed of 10 (I) items averagely.

$Dataset2(DS2)isC8.T8.S8.I8.DB100k.N100$.

$Dataset3(DS3)isC10.T8.S10.I12.DB10k.N200$.

Table 1: Summary of datasets

| Dataset | sequence Numbers | distinct item Numbers | file size |
|---|---|---|---|
| DS1 | 1K | 100 | 1.5M |
| DS2 | 100K | 100 | 112.1M |
| DS3 | 10K | 200 | 13.9M |

*5.2. Experimental Results*

In all experiments, we set $\beta = 0.6$ and $\beta' = 0.7$ and use the different values of $LS1$ and $LS2$ to generate PSP and IPS from three datasets, where $\beta$, $\beta'$, $LS1$ and $LS2$ are parameters in equations 1 and 2.

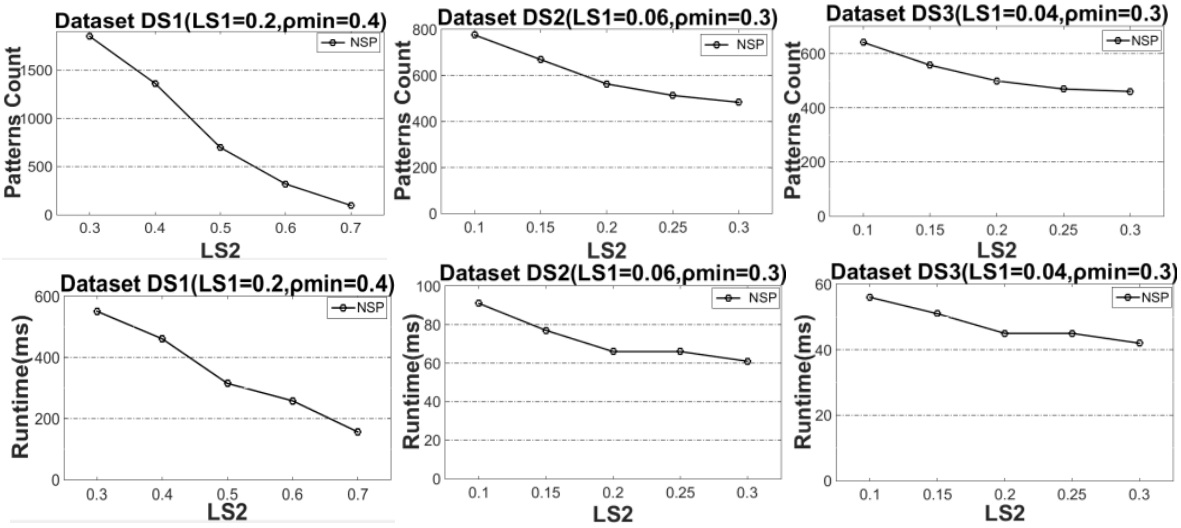


Figure 1: The performance of msNSPFI with 2 Level MMS.

From Fig.1, we can see that with increasing $LS2$, the number of NSP and the runtime decrease. This is because when $LS1$ remains constant, with increasing $LS2$, the number of ISP decreases.

From Fig.2, we can see that with increasing $\rho_{min}$, the number of actionable NSP by all datasets decreases. However, the corresponding runtime almost unchange. Thus using SAP to select actionable NSP almost does not affect the runtime of msNSPFI.

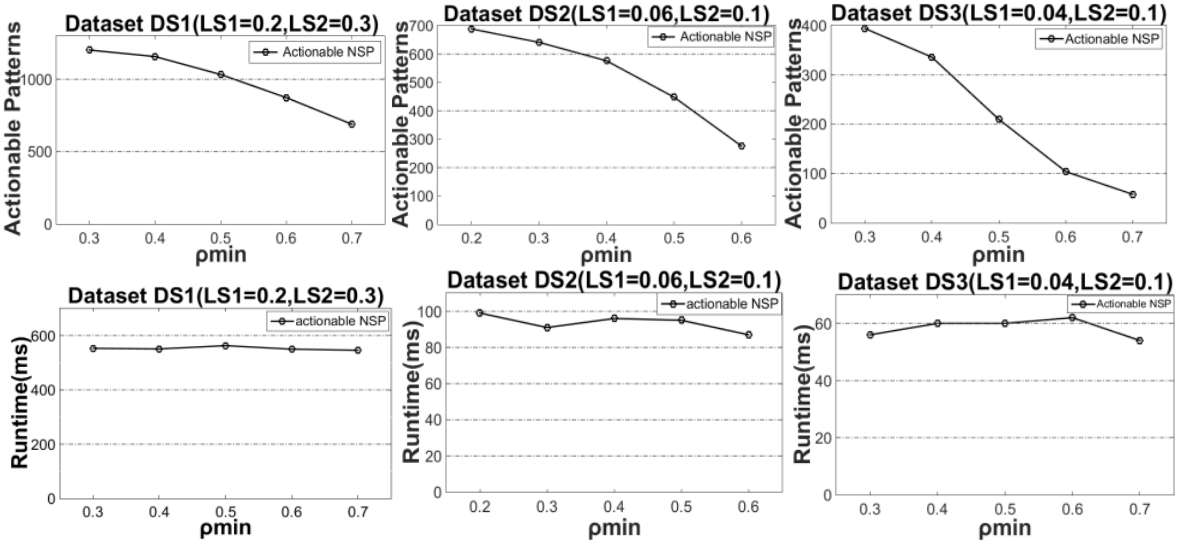


Figure 2: The performance of msNSPFI to select actionable NSP.

From Fig.3, we test the scalability of msNSPFI on datasets DS1 and DS3 to evaluate the msNSPFI performance on large datasets. This is because msNSPFI calculates the supports of NSC based on the *sid* sets of corresponding PSP. Therefore, its performance is sensitive to the size of *sid* sets. If a dataset is large,

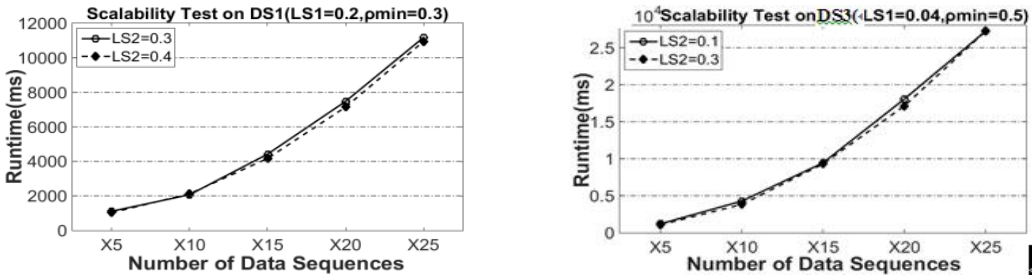


Figure 3: The scalability of msNSPFI.

it produces large sid sets. In terms of different data sizes: from 5 (i.e., 3.5M) to 25 (18.2M) times of DS1, with two *LS*2 0.3 and 0.4, and from 5 (i.e., 36.6M) to 25 (189.3M) times of DS3, with two *LS*2 0.1 and 0.3, the experimental results show that when the sequences of a dataset increase, the runtime increase linearly. Our algorithm is relatively stable.

## 6. Conclusions

NSP mining has increasingly attracted attention in recent years and sometimes play an irreplaceable role in real applications. However, very few methods have been proposed to mine NSP and most of them only mine NSP from PSP. IPS also have useful information. In addition, most of existing methods mine NSP only by using a single *ms*, which is not the case in real applications. To solve these problems, we have proposed an efficient algorithm, named msNSPFI, to mine NSP from IPS with 2-LMMS. Firstly, we have proposed a 2-LMMS to constraint the number of IPS, i.e., assigned two minimum supports for each item to constrain frequent and infrequent sequences. Secondly, In order to ensure that the resulting NSP are actionable, we have introduced the method SAP proposed to select actionable NSP. Experiment results show that msNSPFI can effectively mine actionable NSP from IPS with 2-LMMS. Final, in order to prove the effectiveness of msNSPFI, we have used three ways to do experiments. Experimental results show that msNSPFI is very efficient especially on large datasets.

## References

[1] L.B. Cao, X.J. Dong, Z.G. Zheng, e-NSP: Efficient negative sequential pattern mining, Artificial Intelligence 235 (2016) 156–182.
[2] X.J. Dong, C.L. Liu, T.T. Xu, DK Wang, Select actionable positive or negative sequential patterns,Journal of Intelligent & Fuzzy Systems(fsdm2015), 29(6) (2015) 2759-2767.
[3] C.L. Liu, X.J. Dong, C.Y. Li, L. Li, SAPNSP: Select actionable positive and negative sequential patterns based on a contribution metric, Fuzzy Systems and Knowledge Discovery (FSKD), (2015)811-815.
[4] Y.S. Gong, C.L. Liu, X.J. Dong, Research on typical algorithms in negative sequential pattern mining, Open Automation & Control Systems Journal, 7(1) (2015) 934-941.
[5] Y.S. Gong, T.T. Xu, X.J. Dong, G.H. Lv, e-NSPFI: Efficient mining negative sequential pattern from both frequent and infrequent positive sequential patterns, International Journal of Pattern Recognition and Artificial Intelligence, ,31(2)(2016) 3-14.
[6] T.T. Xu, X.J. Dong, J.L. Xu, Y.S. Gong, E-msNSP: Efficient negative sequential patterns mining based on multiple minimum supports, International Journal of Pattern Recognition and Artificial Intelligence,31(2)(2017).
[7] S.C. Hsueh, M.Y. Lin, C.L. Chen, Mining negative sequential patterns for e-commerce recommendations, in: APSCC08, IEEE (2008) 1213C1218.
[8] Z. Zheng, Y. Zhao, Z. Zuo, L. Cao, Negative-GSP: an efficient method for mining negative sequential patterns, in: Data Mining and Analytics (AusDM09), 101 (2009) 63C67.
[9] W.M. Ouyang, Q.H. Huang, Mining negative sequential patterns in transaction databases, in: ICMLC2007, (2007) 830C834.
[10] N.P. Lin, H.J. Chen, W.H. Hao, Mining negative sequential patterns, in: WSEAS2007 (2007) 654C658.
[11] Z.G. Zheng, Y. Zhao, Z. Zuo, C.L. Cao, An efficient GA-based algorithm for mining negative sequential patterns, in: PAKDD10, in: LNCS, 6118 (2010) 262C273.
[12] T.T. Xu, T.X. Li, X.J. Dong. Efficient High Utility Negative Sequential Patterns Mining in Smart Campus, IEEE Access, 6 (2018) 23839-23847.
[13] X.J. Dong, Z.D. Niu, X.L. Shi, X.D. Zhang, D.H. Zhu, Mining both positive and negative association rules from frequent and infrequent itemsets. Proceedings of the Third International Conference on Advanced Data Mining and Applications ( ADMA 2007), Harbin, China, (2007) 122-133.

[14] J. Ayres, J. Flannick, J. Gehrke and T. Yiu, Sequential pattern mining using a bitmap representation, in KDD'02: Proc. of the 8th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining, New York, NY, USA, (2002) 429C435.
[15] Y.C. Zhao, H.F. Zhang, L.B. Cao, C.Q. Zhang, H. Bohlscheid, Mining both positive and negative impact-oriented sequential rules from transactional data, in: PAKDD09, in: LNCS, 5476 (2009) 656C663.
[16] B. Liu. Web data mining, 2nd Edition. Springer-Verlag Berlin Heidelberg, 2011.
[17] W. Ouyang, Q. Huang. Mining positive and negative sequential patterns with multiple minimum supports in large transaction databases, Second Wri Global Congress on Intelligent Systems. IEEE Computer Society, (2010) 190-193.
[18] N.P. Lin, W.H. Hao, H.J. Chen, C.I. Chang, H.E. Chueh, An algorithm for mining strong negative fuzzy sequential patterns, Int. J. Comput. 3(1) (2007) 167C172.
[19] Y.C. Zhao, H.F. Zhang, L.B. Cao, C.Q. Zhang, H. Bohlscheid, Efficient mining of event-oriented negative sequential rules, in: WI-IAT'2008, 336C342.
[20] X.D. Wu, C.Q. Zhang, S.C. Zhang, Efficient mining of both positive and negative association rules, ACM Trans Inf Syst.22(3)(2004)381-405.
[21] C.L. Liu, G.H. Lv, X.J. Dong, H.N. Yuan, X.J. Dong, Selecting actionable patterns from positive and negative sequential patterns, Journal of Residuals Science & Technology 14(1) (2017) 407-419.
[22] X.J. Dong, Y.S.Gong, L.B.Cao. F-NSP+: A fast negative sequential patterns mining method with self-adaptive data storage, Pattern Recognition, 84 (2018) 13-27.
[23] X.J. Dong; Y.S. Gong; L.B. Cao, e-RNSP: An efficient method for mining repetition negative sequential patterns, IEEE Transactions on Cybernetics, DOI: 10.1109/TCYB.2018.2869907.
[24] X.J. Dong, Z.D. Niu, D.H. Zhu, Z.Y. Zhang, Q.T. Jia, Mining interesting infrequent and frequent itemsets based on MLMS model. The Fourth International Conference on Advanced Data Mining And Applications (ADMA2008), Chengdu, China, Springer-Verlag Berlin Heidelberg, (2008) 444-451.
[25] X.J. Dong, S.J. Wang, H.T. Song. 2-level support based approach for mining positive & negative association rules. Computer Engineering, 31(10) (2005) 16-18.
[26] Y.H. Hu, F. Wu, Y.J. Liao, An efficient tree-based algorithm for mining sequential patterns with multiple minimum supports. Journal of Systems and Software, 86(5)(2013) 1224-1238.
[27] C.K. Tony. Huang. Discovery of fuzzy quantitative sequential patterns with multiple minimum supports and adjustable membership functions. Information Sciences, 222 (2013) 126-146.
[28] R. Agrawal and R. Srikant. Mining sequential patterns. ICDE.1995, 3-14.