Mining Frequent Serial Episodes Over Uncertain Sequence Data

Li Wan *
Computer Science and Technology College
Chongqing University, China
wanli@cqu.edu.cn

Ling Chen, Chengqi Zhang
Centre for Quantum Computation & Intelligent Systems
University of Technology Sydney, Australia
{ling.chen, chengqi.zhang}@uts.edu.au

ABSTRACT

Data uncertainty has posed many unique challenges to nearly all types of data mining tasks, creating a need for uncertain data mining. In this paper, we focus on the particular task of mining *probabilistic frequent serial episodes* (P-FSEs) from uncertain sequence data, which applies to many real applications including sensor readings as well as customer purchase sequences. We first define the notion of P-FSEs, based on the *frequentness probabilities* of serial episodes under possible world semantics. To discover P-FSEs over an uncertain sequence, we propose: 1) an exact approach that computes the accurate frequentness probabilities of episodes; 2) an approximate approach that approximates the frequency of episodes using probability models; 3) an optimized approach that efficiently prunes a candidate episode by estimating an upper bound of its frequentness probability using approximation techniques.

We conduct extensive experiments to evaluate the performance of the developed data mining algorithms. Our experimental results show that: 1) while existing research demonstrates that approximate approaches are orders of magnitudes faster than exact approaches, for P-FSE mining, the efficiency improvement of the approximate approach over the exact approach is marginal; 2) although it has been recognized that the normal distribution based approximation approach is fairly accurate when the data set is large enough, for P-FSE mining, the binomial distribution based approximation achieves higher accuracy when the the number of episode occurrences is limited; 3) the optimized approach clearly outperforms the other two approaches in terms of the runtime, and achieves very high accuracy.

*Dr.Wan is a visiting scholar with UTS/QCIS. Dr Wan is also associated with the State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, China.

EDBT/ICDT '13 March 18 - 22 2013, Genoa, Italy

Categories and Subject Descriptors

H.2.8 [**Database management**]: Data applications—*data mining*; G.3 [**Probability and statistics**]: Distribution functions

General Terms

Algorithms

Keywords

frequent serial episodes, uncertain sequences

1. INTRODUCTION

Frequent serial episode discovery, first introduced in [16], is a popular framework for mining useful and interesting temporal patterns from sequential (and often symbolic) data. Different from frequent sequential patterns [4] which refer to frequent subsequences discovered from a set of sequences, where each sequence consists of a list of elements and each element consists of a set of item symbolics, frequent serial episodes are frequent subsequences mined from a single long sequence of events (represented by symbolic items). Frequent serial episodes have been demonstrated to be an effective tool to unearth temporal correlations in data, being successfully used in many application domains, such as analysis of alarm sequences in telecommunication networks [16], root cause diagnostics from faults log data in manufacturing [23], and user-behavior prediction from web interaction logs [14] etc.

Uncertainty is inherent in data from many different domains, including sensor network monitoring and moving object tracking [22]. The data uncertainty has posed many unique challenges to nearly all types of data mining tasks, creating a need for uncertain data mining. Recently, a number of techniques and algorithms have been devised to take into account data uncertainty during the data mining process, including clustering uncertain data [11][18], classifying uncertain data [6][20][19], and mining frequent patterns over uncertain data [8][2]. A survey on uncertain data mining and management can be found in [3]. However, to our knowledge, this is the first work that studies the problem of mining *frequent serial episodes* over uncertain sequence data.

In this paper, we focus on the problem of mining frequent serial episodes from an uncertain sequence, which consists of an ordered list of uncertain events. Due to the fact that the frequency of a serial episode becomes a discrete random

variable in an uncertain sequence, we define the measure *frequentness probability* to evaluate the probability that the frequency of a serial episode is no less than some minimum frequency threshold. Informally, a serial episode is a *probabilistic frequent serial episode* (P-FSE) if its frequentness probability is no less than some minimum probability.

To mine P-FSEs over an uncertain sequence, we develop three data mining algorithms. First, we develop an exact approach that discovers P-FSEs by computing the accurate frequentness probabilities of episodes using the dynamic programming based scheme. Secondly, we propose an approximate approach that approximates the frequentness probabilities of episodes using probability models. *Normal distribution* has been used in existing probabilistic frequent pattern mining to approximate the frequency distribution and has achieved high accuracy. We theoretically show that the error caused by the normal approximation may be large when the number of episode occurrences is low. In this case, we propose to use the *Binomial distribution* based approximation, which demonstrates high accuracy in our experiments. Thirdly, motivated by the observation that the major computation cost of the exact approach and the approximate approach come from the scanning of the uncertain sequence to recognize all probabilistic occurrences of an episode, we devise an optimized approach that estimates an upper bound of the frequentness probability for an episode without recognizing all of its occurrences. The optimized approach prunes an episode immediately if the upper bound of its frequentness probability does not satisfy the minimum probability threshold.

We carry out extensive experiments on both synthetic and real-world data to evaluate the performance of the proposed P-FSEs mining algorithms. Our experimental results demonstrate the superiority of the optimized approach which achieves the highest efficiency, and very high accuracy at the same time. We also have some findings which highlight the difference between P-FSE mining and other probabilistic frequent pattern mining.

1. While approximate approaches can be orders of magnitudes faster than exact approaches for probabilistic frequent itemset mining [24], for P-FSE mining, the efficiency improvement of the approximate approach is marginal. The reason is that the major cost of P-FSE mining is the occurrence recognition, rather than the frequency checking.

2. Although the Normal distribution based approximation usually achieves high accuracy for probabilistic frequent itemset mining [7], for P-FSE mining, when the number of episode occurrences is low, the Binomial distribution based approximation is more accurate.

The remainder of this paper is structured as follows. We review related research work in Section 2. The concept of probabilistic frequent serial episodes and the problem of probabilistic frequent serial episode mining are stated in Section 3. Section 4 describes the algorithm proposed for P-FSE mining. We evaluate the performance of the proposed data mining algorithms in Section 5. Section 6 closes this paper with some conclusive remarks.

2. RELATED WORK

In this section, we review related research in the following two sub-areas: traditional frequent serial episode mining and frequent pattern mining over uncertain data.

2.1 Frequent Serial Episode Mining

The problem of frequent serial episode mining was first introduced in [16]. Similar to frequent sequential pattern mining, frequent serial episode mining is an important tool to discover useful and interesting temporal patterns from sequential data. While frequent sequential pattern mining discovers patterns from a set of sequences, frequent serial episode mining focuses on discovering episodes from a single long sequence of events. Various frequency definitions of episodes have been proposed, which has given rise to different types of frequent episodes. Recently, Achar et al. [1] reviewed 7 different frequency definitions in the literature. Three of them, window-based frequency [16], head frequency [9], and total frequency [9], consider the number of windows containing at least one occurrence of an episode, where each window has the same specified width. The remainder definitions, minimal occurrence-based frequency [16], non-overlapped frequency [13], non-interleaved frequency [12] and distinct frequency [10], directly take into account the different occurrences of an episode in the sequence. Most of the existing algorithms for frequent serial episode mining are level-wise apriori-based discovery methods, which involve two main steps: candidate generation and frequency counting. Candidate generation is usually handled based on the anti-monotonicity property or some more restricted anti-monotonic properties introduced by different frequency definitions. For frequency counting, most algorithms [16][13] use the finite state automata as the basic building blocks for recognizing occurrences of episodes in data sequence. In this paper, we study the problem of frequent serial episode mining in the context of uncertain data. In particular, we consider frequent serial episodes defined on the non-overlapped frequency and the distinct frequency, since the two frequency definitions similarly incur independent occurrences of an episode.

2.2 Pattern Mining over Uncertain Data

Due to the wide applications of uncertain data, mining frequent patterns over uncertain databases has attracted much attention recently. Frequent itemset mining and frequent sequential pattern mining are two of the most important pattern mining problems studied under the uncertain environment.

Existing work on mining frequent itemsets from uncertain databases falls into two categories based on the definition of a frequent itemset: *expected support-based frequent itemset* [8][2] and *probabilistic frequent itemset* [5][21]. Both definitions consider the frequency (support) of an itemset as a discrete random variable. The former employs the expectation of the support as the measurement. That is, an itemset if frequent only if the expected support of the itemset is no less than a specified minimum expected support. The latter uses the *frequentness probability* as the measurement, which is the probability that an itemset appears no less than a specified minimum support times. Then, an itemset is frequent only if its frequentness probability is no less than a specified minimum probability threshold. For mining expected support-based frequent itemsets, there are three representative algorithms: UApriori [8], UFP-growth [15], and UH-Mine [2]. UApriori is the uncertain version of the well-known Apriori algorithm. Both UFP-growth and UH-Mine are based on the divide-and-conquer framework which uses the depth-first strategy to search frequent itemsets. For

mining probabilistic frequent itemsets, two representative algorithms are DP − dynamic programming-based Apriori algorithm [5], and DC − divide-and-conquer-based Apriori algorithm [21]. Approximate probabilistic frequent itemsets mining algorithms based on Poisson distribution and Normal distribution have been proposed as well in [24] and [7], respectively. An empirical comparison of eight existing representative algorithms for frequent itemset mining over uncertain data has been reported in [22] with uniform measures.

Some initial research has been undertaken to mine sequential patterns from uncertain sequence data. For example, the expected support-based frequent sequential pattern mining has been studied in [17]. In contrast, Zhao et al. [25] propose to mine probabilistic frequent sequential patterns according to the frequentness probability. To our knowledge, this is the first work of mining frequent serial episodes from uncertain sequence data.

3. PROBLEM DEFINITION

In this section, we first review relevant concepts in deterministic data. Next, we introduce the definitions in uncertain sequence data. Then, the problem of probabilistic frequent serial episode mining is formulated.

3.1 Preliminaries in deterministic data

Definition 1. Given a set of event types E, an event is a pair (e, t) where $e \in E$ is an event type and t is the occurrence time of the event. Then, an **event sequence** is an ordered sequence of events, denoted by $\mathbf{S} = \langle (e_1, t_1)(e_2, t_2) \dots (e_n, t_n) \rangle$, such that $e_i \in E$ ($\forall\ i \in [1, n]$), and $t_i \leq t_{i+1}$ ($\forall\ i \in [1, n-1]$).

For example, the following is an event sequence containing seven events:

$$\langle (A,1), (B,3), (A,4), (C,6), (A,14), (B,15), (C,17) \rangle \quad (1)$$

Definition 2. A **serial episode** α is a triple $(V, \leq, g)$ where V is a set of nodes, $\leq$ is a total order on V, and $g : V \rightarrow E$ is a mapping associating each node in α with an event type in E. The interpretation of a serial episode is that the events in $g(V)$ have to occur in the order described by $\leq$. A serial episode α containing $|V|$ nodes is a $|V|$-node episode.

Consider a 3-node episode $\alpha = (V_\alpha, \leq_\alpha, g_\alpha)$, $V_\alpha = \{v_1, v_2, v_3\}$, $g_\alpha(v_1) = A$, $g_\alpha(v_2) = B$, $g_\alpha(v_3) = C$, with $v_1 \leq_\alpha v_2 \leq_\alpha v_3$. We denote the episode as $(A \rightarrow B \rightarrow C)$. For brevity, we will refer to a serial episode as an episode hereafter.

Definition 3. An episode $\beta = (V', \leq', g')$ is a **sub-episode** of $\alpha = (V, \leq, g)$, denoted by $\beta \preceq \alpha$ if there exists a mapping $f : V' \rightarrow V$ such that $g'(v) = g(f(v))$ for all $v \in V'$, and for all $v, w \in V'$ with $v \leq' w$, $f(v) \leq f(w)$ also. An episode α is a **super-episode** of β if and only if $\beta \preceq \alpha$.

For example, let α be a 3-node episode $(A \rightarrow B \rightarrow C)$, and β be a 2-node episode $(A \rightarrow B)$. Then, $\beta \preceq \alpha$.

Definition 4. An episode $\alpha = (V, \leq, g)$ **occurs** in an event sequence $\mathbf{S} = < (e_1, t_1)(e_2, t_2) \dots (e_n, t_n) >$ if there exists a mapping $h : V \rightarrow \{1, \dots, n\}$ from nodes of α to events of $\mathbf{S}$, such that $g(v_i) = E_{h(v_i)}$, $\forall\ v_i \in V$; and $\forall\ v_i, v_j \in V$ with $i \neq j$ and $v_i \leq v_j$, we have $t_{h(v_i)} < t_{h(v_j)}$.

The 3-node episode $\alpha = (A \rightarrow B \rightarrow C)$ occurs in the example sequence (1). For example, the events $\langle (A,1)(B,3)(C,6) \rangle$ constitute an occurrence of α.

As reviewed in Section 2, a number of different frequency definitions have been proposed to capture how often an episode occurs in an event sequence. We observe that existing frequency definitions can be grouped into two categories: definitions incurring dependent occurrences (e.g. two occurrences of an episode may share common events) and definitions incurring independent occurrences. Due to space constraint, in this paper, we focus on only the type of frequency definitions incurring independent occurrences, which contains two frequency definitions, the non-overlapped frequency [13] and the distinct frequency [10]. We review the definitions of the two frequency measure as follows.

Definition 5. Two occurrences h_1 and h_2 of an N-node episode α are said to be *non-overlapped* if either $t_{h_1(v_N)} < t_{h_2(v_1)}$ or $t_{h_2(v_N)} < t_{h_1(v_1)}$. A set of occurrences is said to be non-overlapped if every pair of occurrences in the set is non-overlapped. A set H, of non-overlapped occurrences of α in $\mathbf{S}$ is maximal if $|H| \geq |H'|$, where H' is any other set of non-overlapped occurrences of α in $\mathbf{S}$. The **non-overlapped frequency** of α in $\mathbf{S}$, denoted as $freq_{no}(\alpha)$, is defined as the cardinality of a maximal non-overlapped set of occurrences of α in $\mathbf{S}$.

Consider the occurrences of $\alpha = (A \rightarrow B \rightarrow C)$ in the example sequence (1). The two occurrences $\langle (A,1)(B,3)(C,6) \rangle$ and $\langle (A,14)(B,15)(C,17) \rangle$ are non-overlapped, while $\langle (A,1)(B,3)(C,6) \rangle$ and $\langle (A,4)(B,15)(C,17) \rangle$ are overlapped because $(A,4)$ occurs before $(C,6)$. Hence, the non-overlapped frequency considers the first set of occurrences, $freq_{no}(\alpha) = 2$.

Definition 6. Two occurrences h_1 and h_2 of an N-node episode α are said to be *distinct* if $h_1(v_i) \neq h_2(v_j)$, $\forall\ i, j \in [1, N]$. A set of occurrences is distinct if every pair of occurrences in it is distinct. A set H of distinct occurrences of α in $\mathbf{S}$ is maximal if $H \geq H'$, where H' is any other set of distinct occurrences of α in $\mathbf{S}$. The **distinct occurrence frequency** of α in $\mathbf{S}$, denoted as $freq_d(\alpha)$, is the cardinality of a maximal set of distinct occurrences of α in $\mathbf{S}$.

According to distinct occurrence frequency, both of the two sets of occurrences of $\alpha = (A \rightarrow B \rightarrow C)$ in sequence (1), $\{\langle (A,1)\ (B,3)(C,6) \rangle\ \langle (A,14)(B,15)(C,17) \rangle\}$ and $\{\langle (A,1)\ (B,3)(C,6) \rangle\ \langle (A,5)(B,15)(C,17) \rangle\}$, are valid because the occurrences in each set are distinct from each other [1]. Thus, $freq_d(\alpha) = 2$.

Then, an episode is frequent if its frequency is no less than some user specified minimum frequency threshold. Given an event sequence, the problem of frequent serial episode mining is to discover the complete set of episodes satisfying the minimum frequency threshold.

3.2 Concepts in uncertain data

After reviewing relevant definitions of frequent serial episode mining in deterministic data, we now introduce the concepts in the context of uncertain data.

[1]There are definitions in the literature, such as *earliest transition* and *minimal occurrence windo*w, to further restrict the particular set of occurrences to be considered.

Table 1: An example of uncertain sequence.

Uncertain Sequence
$\langle(A, 0.6, 1)(A, 0.7, 2)(B, 0.8, 2)(B, 1.0, 3)(C, 1.0, 4)\rangle$

Table 2: Possible worlds of the example sequence.

ID	Sequence	Prob.
pw_1	$\langle(A,1)(A,2)(B,3)(C,4)\rangle$	0.084
pw_2	$\langle(A,1)(B,2)(B,3)(C,4)\rangle$	0.144
pw_3	$\langle(A,1)(A,2)(B,2)(B,3)(C,4)\rangle$	0.336
pw_4	$\langle(A,1)(\emptyset,2)(B,3)(C,4)\rangle$	0.036
pw_5	$\langle(\emptyset,1)(A,2)(B,3)(C,4)\rangle$	0.056
pw_6	$\langle(\emptyset,1)(B,2)(B,3)(C,4)\rangle$	0.096
pw_7	$\langle(\emptyset,1)(A,2)(B,2)(B,3)(C,4)\rangle$	0.224
pw_8	$\langle(\emptyset,1)(\emptyset,2)(B,3)(C,4)\rangle$	0.024

Definition 7. Let E be a set of all event types. An *uncertain event* is a triple (e, p, t) where $e \in E$ is an event type, $p \in [0, 1]$ is the existential probability of the event, and t is the occurrence time of the event. Then, **an uncertain sequence** is an ordered list of uncertain events, denoted by $\mathbb{S} = \langle(e_1, p_1, t_1)(e_2, p_2, t_2)\dots(e_n, p_n, t_n)\rangle$, where $e_i \in E$, $\forall\ i \in [1, n]$; $t_i \leq t_{i+1}$, $\forall\ i \in [1, n-1]$.

For example, Table 1 shows an example uncertain sequence with 5 uncertain events, where each event is associated with an existential probability and an occurrence time.

Possible world semantics are commonly used to explain uncertain data. Given an uncertain event sequence $\mathbb{S}$, a set of possible worlds $\mathcal{PW} = \{pw_1, pw_2, \dots, pw_w\}$ can be derived, where each possible world pw_i, associated with an existential probability, contains deterministic events. For example, Table 2 shows the set of 8 possible worlds that can be derived from the uncertain sequence in Table 1. The probability of a possible world can be computed based on the probabilities of corresponding events. For example, the probability of pw_1 is equal to $0.6 \times 0.7 \times (1 - 0.8) \times 1.0 \times 1.0$=0.084.

Given an episode α, we can compute its frequency in a possible world, $freq(\alpha, pw_i)$, using the frequency definitions (e.g. $freq_{no}$ and $freq_d$) introduced in the deterministic environment. Since each possible world pw_i is associated with an existential probability $\Pr(pw_i)$, the frequency of an episode α in a possible world pw_i is also associated with the probability $\Pr(pw_i)$. Therefore, the frequency of an episode in an uncertain sequence is a probability distribution.

Definition 8. Given an uncertain event sequence $\mathbb{S}$, let $\mathcal{PW} = \{pw_1, pw_2, \dots, pw_w\}$ be the set of possible worlds derived from $\mathbb{S}$, the **frequency probability** of an episode α, denoted as $\Pr(freq(\alpha) = c)$, is defined as follows,

$$\Pr(freq(\alpha) = c) = \sum_{pw_i \in \mathcal{PW}, freq(\alpha, pw_i) = c} \Pr(pw_i) \qquad (2)$$

For instance, let α be a 3-node episode $(A \rightarrow B \rightarrow B)$. For either non-overlapped frequency or distinct frequency, α occurs once in the possible worlds pw_2 and pw_3, the probability that α has the frequency value 1, $\Pr(freq(\alpha) = 1) = \Pr(pw_2) + \Pr(pw_3) = 0.144 + 0.336 = 0.48$.

Then, following existing work on mining probabilistic frequent patterns over uncertain data, we define the concept of *frequentness probability* of an episode to measure the probability that an episode occurs no less than some specified minimum occurrence times.

Definition 9. Given an uncertain event sequence $\mathbb{S}$, and a minimum frequency threshold τ_{freq}, the **frequentness probability** of an episode α with respect to τ_{freq}, denoted as $\Pr(freq(\alpha) \geq \tau_{freq})$, is defined as,

$$\Pr(freq(\alpha) \geq \tau_{freq}) = \sum_{c=\tau_{freq}}^{|\mathbb{S}|} \Pr(freq(\alpha) = c) \qquad (3)$$

where $|\mathbb{S}|$ is the number of uncertain events in $\mathbb{S}$, which is consequently the maximum number of times that α may occur.

Frequentness probability reflects how likely an episode is frequent in an uncertain sequence. Then, we can define *probabilistic frequent serial episodes* based on frequentness probability.

Definition 10. Given an uncertain event sequence $\mathbb{S}$, a minimum frequency threshold τ_{freq}, and a minimum probability threshold τ_{prob}, an episode α is a **probabilistic frequent serial episode** (P-FSE) iff $\Pr(freq(\alpha) \geq \tau_{freq}) \geq \tau_{prob}$.

3.3 Problem statement

Formally, given an uncertain sequence $\mathbb{S}$, a frequency threshold τ_{freq} and a probability threshold τ_{prob}, the problem of **probabilistic frequent serial episode (P-FSE) mining** is to find all serial episodes where for each serial episode α, $\Pr(freq(\alpha) \geq \tau_{freq}) \geq \tau_{prob}$.

4. P-FSE MINING

In this section, we present the data mining algorithms devised for P-FSE mining. We first describe the algorithm that discovers P-FSEs by computing the exact frequentness probabilities of episodes. Then, we introduce an approximate algorithm that discovers P-FSEs using probability models. Finally, we propose an optimized approach that early prunes episodes which are not P-FSEs by estimating the upper bounds of their frequentness probabilities.

4.1 Exact frequency based approach

We note that the frequentness probability, defined based on either the non-overlapped frequency or the distinct frequency, satisfy the anti-monotonic property.

Property 1. Given an uncertain sequence $\mathbb{S}$, a frequency threshold τ_{freq} and a probability threshold τ_{prob}, if an episode α is not a P-FSE, then any episode β s.t. $\alpha \preceq \beta$ is not a P-FSE.

The property follows trivially from the fact that in every possible world, the non-overlapped frequency and the distinct frequency satisfy the anti-monotonic property. According to Property 1, we devise the exact frequency based approach with the main idea illustrated in Algorithm 1.

Basically, we discover first the set of P-FSEs of individual events (lines 1-7). Then, for each discovered P-FSE of length l, we grow it by one event from the list of individual P-FSEs to examine whether the new episode of length $(l+1)$ is a P-FSE (lines 8-13). That is, the depth-first search based

Algorithm 1 Exact Frequency based P-FSE Mining

input $\mathbb{S}$ - uncertain sequence on events E, τ_{freq}, τ_{prob}
output $\mathcal{P}$ - complete set of P-FSEs

1: $\mathcal{I} = \{\}$, $\mathcal{P} = \{\}$
2: **for** each individual event $X \in E$ **do**
3: $\alpha = \emptyset$, $\mathcal{O}|_\alpha = \{\}$
4: $\mathcal{O}|_{\alpha \sqcup (X)} = Occurrence\_Recognition(\mathcal{O}|_\alpha, (X))$
5: **if** $Frequency\_Check(\mathcal{O}|_{\alpha \sqcup (X)})$ **then**
6: $\mathcal{I} = \mathcal{I} \cup \{(X)\}$
7: $\mathcal{P} = \mathcal{I}$
8: **for** each individual episode α in $\mathcal{I}$ **do**
9: **for** each individual episode β in $\mathcal{I}$ **do**
10: $\mathcal{O}|_{\alpha \sqcup \beta} = Occurrence\_Recognition(\mathcal{O}|_\alpha, \beta)$
11: **if** $Frequency\_Check(\mathcal{O}|_{\alpha \sqcup \beta})$ **then**
12: $\mathcal{P} = \mathcal{P} \cup \{\alpha \sqcup \beta\}$,
13: $\alpha = \alpha \sqcup \beta$, go to line 9
14: Return $\mathcal{P}$

Algorithm 2 Occurrence Recognition

input
$\mathcal{O}|_\alpha = \langle O_1, O_2, \ldots, O_m \rangle$, X - the event to grow α by
output
$\mathcal{O}|_{\alpha \sqcup X}$ - the recognized occurrences of episode $\alpha \sqcup X$

1: **for** each $O_i \in \mathcal{O}|_\alpha$ **do**
2: Scan $\mathbb{S}$ from O_i^e to the end of $\mathbb{S}$
3: **if** X is found at time t_k **then**
4: **if** $freq_{no}$ is used **then**
5: **for** each $O_j \in \mathcal{O}|_\alpha$ s.t. $i < j \leq m$ & $t_k \geq O_j^s$ **do**
6: remove O_j from $\mathcal{O}|_\alpha$
7: update $O_i^e = t_k$
8: update $\Pr(O_i) = \Pr(O_i) \times \Pr(X, t_k)$
9: **else if** $freq_d$ is used **then**
10: **for** each $O_j \in \mathcal{O}|_\alpha$ s.t. $i < j \leq m$ **do**
11: **if** $E(O_j, t_k) == X$ **then**
12: remove O_j from $\mathcal{O}|_\alpha$
13: **for** each $O_j \in \mathcal{O}|_{\alpha \sqcup X}$ **do**
14: **if** $E(O_j, t_k) == X$ **then**
15: break
16: update $O_i = O_i \cup (X, t_k)$
17: update $\Pr(O_i) = \Pr(O_i) \times \Pr(X, t_k)$
18: $\mathcal{O}|_{\alpha \sqcup X} = \mathcal{O}|_{\alpha \sqcup X} \cup O_i$
19: Return $\mathcal{O}|_{\alpha \sqcup X}$

strategy is used. We stop growing an episode if it is discovered to be not a P-FSE, according to Property 1. There are two main functions engaged in Algorithm 1: *occurrence recognition* and *frequency check*, which are explained respectively in details as follows.

The aim of occurrence recognition is to discover all occurrences of an episode from the input sequence for frequency check. Different from other frequent pattern mining, occurrence recognition in frequent serial episode mining should make sure each occurrence is valid according to the frequency definition. In this paper, we consider the non-overlapped frequency ($freq_{no}$) and the distinct frequency ($freq_d$). For the former, we should check whether any two occurrences are overlapped. To this end, we need to record the starting time and ending time of each occurrence of an episode. For example, let O_i be the i-th occurrence of an episode α. We denote $O_i = (O_i^s, O_i^e)$ which is a pair of time points respectively representing the occurrence times of the first event and the last event of α in the i-th occurrence. For the latter, we should examine whether any two occurrences are distinct. For this purpose, we record each event (i.e. the event type and its occurrence time) that constitutes an occurrence of an episode. That is, the i-th occurrence of an l-node episode is denoted as $O_i = (O_i^1, O_i^2, \ldots, O_i^l)$, where $O_i^j = (E_j, t_j)$ records both the event type and the occurrence time of the event constituting the occurrence of the j-th node in the episode. Then, Algorithm 2 shows the details of the function for occurrence recognition.

The input of the function contains the list of chronologically ordered occurrences of the episode α examined in the last round, and the single node episode (X) that will be appended to α. For each occurrence of α, we need to examine the subsequent events in the sequence to discover the occurrence of the new episode $(\alpha \sqcup X)$ (lines 1-3). If the non-overlapped frequency ($freq_{no}$) is considered, we should check whether the newly found occurrence overlaps with any potential subsequent occurrences. If it happens, we remove the potential subsequent occurrences from consideration[2] (lines 5-6). Otherwise, we update the ending time of current occurrence (line 7), which will be inserted into the list of occurrences of $\alpha \sqcup X$ (line 18). If the distinct frequency is adopted, similarly, we first remove potential subsequent occurrences from consideration if it shares an event with the newly found occurrence (lines 10-12). The function $E(O_j, t_k)$ returns the event type of the event occurring at time t_k in the occurrence O_j. We should further check whether the current occurrence shares an event with existing occurrences of $(\alpha \sqcup X)$. If it happens, we should skip considering the current occurrence (lines 13-15). Otherwise, we update the current occurrence by appending the new event (line 16), and insert the current occurrence into $\mathcal{O}|_{(\alpha \sqcup X)}$ (line 18).

Note that, since the input sequence is uncertain, the discovered occurrences are probabilistic. That is, each occurrence is associated with an existential probability, which represents the likelihood the episode appears in the occurrence. Therefore, for each probabilistic occurrence O_i, we record a probability $\Pr(O_i)$, which can be computed by multiplying the probabilities of the uncertain events in the occurrence. In Algorithm 2, lines 8 and 17 respectively update the probability of an occurrence under the two different frequency definitions. $\Pr(X, t_k)$ returns the existential probability of the event $(X, t_k) \in \mathbb{S}$.

The function of frequency check computes the frequentness probability of an episode, given the list of recognized probabilistic occurrences. Since the probabilistic occurrences recognized based on the non-overlapped frequency or the distinct frequency are independent, a dynamic programming based scheme can be used to compute the frequentness probability of an episode, by splitting the problem into smaller problems. In particular, given the list of probabilistic occurrences of an episode α, $\mathcal{O}|_\alpha = \langle O_1, O_2, \ldots, O_m \rangle$, let $\mathcal{O}_j|_\alpha$ be the first j occurrences in $\mathcal{O}|_\alpha$, $\mathcal{O}_j|_\alpha \subseteq \mathcal{O}|_\alpha$. Then, the probability that α occurs at least i times in the first j possible occurrences of $\mathcal{O}|_\alpha$, denoted as $P_{\geq ij}(\alpha)$, can be considered as follows. If α occurs in the j-th probabilistic occurrence O_j, then the probability $P_{\geq ij}(\alpha)$ equals to the probability that at least $(i-1)$ occurrences of $\mathcal{O}_j|_\alpha \setminus \{O_j\}$ contain α. Otherwise, $P_{\geq ij}(\alpha)$ is equal to the probability that at least i

[2]In this paper, we consider occurrences that appear first. It can be revised straightforwardly to consider occurrences in minimal occurrence windows.

probabilistic occurrences of $\mathcal{O}_j|_\alpha \setminus \{O_j\}$ contain α. Thus, the frequentness probability $\Pr(freq(\alpha) \geq \tau_{freq}) = P_{\geq \tau_{freq},m}$ can be computed recursively based on the following strategy:

$$P_{\geq ij}(\alpha) = P_{\geq i-1,j-1}(\alpha) \cdot \Pr(O_j) + P_{\geq i,j-1}(\alpha) \cdot (1 - \Pr(O_j))$$

where the boundary case is:

$$\begin{cases} P_{\geq 0,j}(\alpha) = 1 & \text{if } 0 \leq j \leq m \\ P_{\geq i,j}(\alpha) = 0 & \text{if } i > j \end{cases}$$

The dynamic programming based frequentness probability computation was introduced in [5] for probabilistic frequent itemset mining. The time complexity of the dynamic programming computation for each episode is $O(m^2 \times \tau_{freq})$, where m is the number of probabilistic occurrences.

4.2 Approximate frequency based approach

Besides exact approaches, approximate approaches based on probability models, such as Poisson distribution and Normal distribution, have been proposed for probabilistic frequent pattern mining over uncertain data. We now show that approximation techniques can be employed for P-FSE mining as well. Different from probabilistic frequent itemset mining which is usually performed on a large set of transaction data, frequent serial episode mining is performed on a single long sequence, which may contain only limited number of occurrences of an episode. In this case, we show that the Normal distribution model is not appropriately applicable, and propose to approximate the frequentness probability of an episode using the Binomial distribution model instead. In the following, we describe the approximations of frequentness probability using the Normal distribution and the Binomial distribution respectively.

Approximating the frequency distribution of a pattern with a Normal distribution using a weak version of the Central Limit Theorem is introduced in [7]. Let $Y_1, Y_2, \ldots$ be an infinite sequence of stochastic variables, and let s_N^2 denote $\sum_{k=1}^{N} \sigma_k^2$ for all numbers N, where σ_k^2 denotes the variance of variable Y_k. A weak form of the central limit theorem, known as the *Lyapunov's Central limit Theorem*, states that if for some $\delta > 0$ the following two conditions hold:

1. $E[|Y_k|^{2+\delta}]$ is finite for all k, and
2. $\lim_{N\to\infty} \frac{1}{s_N^{2+\delta}} \Sigma_{i=1}^{N} E[|Y_i - \mu_i|^{2+\delta}] = 0$

then the Central Limit Theorem still holds, i.e.

$$Z_N = \frac{\Sigma_{i=1}^{N}(Y_i - \mu_i)}{s_N}$$

converges in distribution to a standard normal random variable as N goes to infinity.

For P-FSE mining, Y_k is a stochastic variable denoting if an episode α appears in the k-th probabilistic occurrence O_k. Y_k follows a Bernoulli distribution and $P(Y_k = 1) = 1 - P(Y_k = 0) = \Pr(O_k)$ is the existential probability associated with O_k. The frequency of the episode can then be expressed by the probabilistic variable $freq(\alpha) = \sum_{i=1}^{m} Y_i$, and the expected frequency is $E[Y]$, where $Y = (Y_1, Y_2, \ldots, Y_m)$.

As shown in [7], the two conditions hold for $\delta = 1$. Therefore, Z_N converges to a Normal distribution for increasing N. When N equals to the number of probabilistic occurrences, we get $\frac{\Sigma_{i=1}^{N}(Y_i - \mu_i)}{s_N} = \frac{freq(\alpha) - E[Y]}{\sqrt{s_N^2}}$. For sufficiently large number of probabilistic occurrences, $\frac{freq(\alpha) - E[Y]}{\sqrt{s_N^2}}$ converges in probability to the standard normal distribution. Thus:

$$\Pr(freq(\alpha) \geq \tau_{freq}) \approx \Phi(\frac{E[Y] + 0.5 - \tau_{freq}}{\sqrt{s_N^2}}) \qquad (4)$$

where Φ is the cdf of the standard normal distribution.

The quality of the approximation can be measured using the Berry-Esseen theorem, which gives an upper bound on the error. The theorem states that, there exists some positive constant C less than 0.7164, such that if $Y_1, Y_2, .., Y_m$ are i.i.d. random variables with $E[Y] = 0$, $E[Y^2] = \sigma^2 > 0$, and $E[Y^3] = \rho < \infty$, then for all Y and m,

$$|\Pr(freq(\alpha) > \tau_{freq}) - \Phi(\frac{E[Y] + 0.5 - \tau_{freq}}{\sqrt{s_N^2}})| \leq \frac{C\rho}{\sigma^3\sqrt{m}} \qquad (5)$$

It can be observed that, if m is not sufficiently large, the frequentness probability approximated using the Normal distribution may be quite inaccurate, which is verified by our experimental results. In this case, we propose to use the Binomial distribution to approximate the frequentness probability. If the existential probabilities of occurrences are all identical, e.g. $\Pr(O_i) = p$, the frequency of an episode α, $freq(\alpha) = \sum_{i=1}^{m} Y_i$ follows a Binomial distribution $B(m, p)$, where we set $p = m^{-1} \sum_{i=1}^{m} \Pr(O_i)$. Thus,

$$\Pr(freq(\alpha) \geq \tau_{freq}) \approx 1 - (m-k) C_m^k \int_0^{1-p} t^{m-k-1}(1-t)^k dt \qquad (6)$$

where k is τ_{freq}. Clearly, this remains approximately true only if the values of $\Pr(O_i)$ are approximately equivalent. A direct setup of Stein identity for estimating binomial approximation error is given in Ehm (1991). Let $p = m^{-1} \sum_{i=1}^{m} \Pr(O_i)$, and $q = 1 - p$, the correct order for the distance between $\Pr(freq(\alpha))$ and $B(m, p)$ is

$$min(1, mpq^{-1}) \sum_{i=1}^{m} (\Pr(O_i) - p)^2 \qquad (7)$$

The framework of the approximate frequency based P-FSE mining algorithm is similar to the exact one illustrated in Algorithm 1. After recognizing all probabilistic occurrences of an episode, we compute some statistics on the

Algorithm 3 Approximate Frequency Check

input statistics of $\mathcal{O}|_\alpha$, N - the occurrence number threshold, γ_N - the error threshold for Normal approximation, γ_B - the error threshold for Binomial approximation
output approximation frequentness probability of α

```
if |O|_α| > N then
    calculate the upper bound of error for normal approximation, e_N, according to Eq. (5)
    if e_N ≤ γ_N then
        return Pr(freq(α) ≥ τ_freq) according to Eq. (4)
else
    calculate upper bound of error for binomial approximation, e_B, according to Eq. (7)
    if e_B ≤ γ_B then
        return Pr(freq(α) ≥ τ_freq) according to Eq. (6)
return -1
```

sequence of corresponding existential probabilities, such as $\sigma_i^2 = \Pr(O_i)(1 - \Pr(O_i))$ and $s_m^2 = \sum_{i=1}^{m} \sigma_i^2$, which will be used in the Normal approximation, and the expected frequency $\lambda = \sum_{i=1}^{m} \Pr(O_i)$ which will be used in the Binomial approximation. Then, instead of calling the dynamic programming based frequency check method, we call the function of *Approximate_Frequency_Check*, which is illustrated in Algorithm 3. If the number of probabilistic occurrences is greater than some threshold N, the Normal approximation will be used if the approximation error is less than some predefined error threshold γ_N. If the number of occurrences is small, the Binomial approximation is adopted if the approximate error does not exceed the error threshold γ_B. Otherwise, the function returns -1 as an indication to fall back to the exact approach to compute the frequentness probability for the current episode.

4.3 Optimized approach

We observe that the major cost of the exact approach and the approximate approach comes from the step of scanning the whole sequence to recognize all valid probabilistic occurrences of an episode. Besides, both approaches have to recognize all occurrences of an episode before computing its frequentness probability. Therefore, we are motivated to decide early if an episode is not a P-FSE, before recognizing all probabilistic occurrences. We note that this can be achieved by estimating an upper bound of an episode's frequentness probability using part of its sub-episode's probabilistic occurrences.

Lemma 1. Let $\mathcal{O}|_\alpha = \langle O_1, O_2, \ldots, O_m \rangle$ be the list of m probabilistic occurrences of an episode α. Let $\mathcal{Q}|_{\alpha \sqcup X} = \langle Q_1, Q_2, \ldots, Q_n \rangle$ be the list of n probabilistic occurrences of episode $(\alpha \sqcup X)$, where $n \leq m$. $\forall 1 \leq i \leq n$, $\Pr(freq(\alpha \sqcup X))$ is no greater than the frequentness probability computed on the sequence of $\langle \Pr(Q_1), \ldots, \Pr(Q_i), \Pr(O_{i+1}), \ldots, \Pr(O_m) \rangle$.

Lemma 1 can be proved straightforwardly based on the following two facts. First, the length of the mixed sequence $\langle \Pr(Q_1), \ldots, \Pr(Q_i), \Pr(O_{i+1}), \ldots, \Pr(O_m) \rangle$ is no less than $|\mathcal{Q}|_{\alpha \sqcup X}|$. Secondly, each probability value in the mixed sequence is no less than the corresponding probability at the same position in $\mathcal{Q}|_{\alpha \sqcup X}$, because α is a sub-episode of $(\alpha \sqcup X)$. According to the definition of frequentness probability, if an episode has more probabilistic occurrences, and each occurrence is associated with higher existential probability, the episode is probabilistically more frequent. That is, its frequentness probability is higher. Therefore, the frequentness probability of $(\alpha \sqcup X)$ should be no greater than the one computed on the mixed sequence.

Lemma 1 defines an upper bound of the frequentness probability of an episode $(\alpha \sqcup X)$, which can be computed using part of the probabilistic occurrences of its base episode α. If the upper bound is less than τ_{prob}, we can prune the episode $(\alpha \sqcup X)$ without recognizing the remainder occurrences from Q_{i+1} to Q_n. While the dynamic programming based scheme can be used to compute the upper bound based on the mixed sequence, it is computationally expensive. We thus use the approximation technique to approximate the upper bound.

The framework of the optimized approach is similar to the exact approach. The difference is that the optimized approach calls a function of *Early_Prune*, before the end of the *Occurrence_Recognition* (e.g. between lines 18 and 19 in Algorithm 2), when there are τ_{freq} occurrences have been recognized. This is because if the number of occurrences is less than τ_{freq}, the episode cannot be probabilistically frequent. Algorithm 4 presents the main idea of *Early_Prune*. It first builds the mixed sequence of occurrences. Then, the *Approximate_Frequency_Check* is called to compute the upper bound $\widetilde{\Pr}(freq(\beta))$. If $\widetilde{\Pr}(freq(\beta))$ is less than τ_{prob}, then the episode $(\alpha \sqcup X)$ will be pruned immediately. Note that, since we use approximation technique to compute the upper bound, the optimized approach is also an approximate method. We will experimentally evaluate the accuracy of the approach.

Algorithm 4 Early Prune

input $\mathcal{Q}_{\tau_{freq}}|_{(\alpha \sqcup X)}$ - first τ_{freq} occurrences of $\alpha \sqcup X$, $\mathcal{O}|_\alpha$ - occurrences of α, N- the occurrence number threshold, γ_N and γ_B - the error thresholds
output $+1$ - prune episode $\alpha \sqcup X$, -1 - no prune

```
1: Let Q|_β be ⟨Q_1, ..., Q_{τ_freq}, O_{τ_freq+1}, ..., O_m⟩
2: Pr~(freq(β)) =
   Approximate_Frequency_Check(Q|_β, N, γ_N, γ_B)
3: if Pr~(freq(β)) ≠ −1 then
4:    if Pr~(freq(β)) < τ_prob then
5:       return +1
6: return -1
```

5. EXPERIMENTS

In this section, we evaluate the performance of the proposed algorithms for P-FSE mining using both synthetic and real datasets. Specifically, we test the performance of the three P-FSE mining approaches under the distinct frequency definition and the non-overlapped frequency definition in Sections 5.1 and 5.2 respectively. Then, we apply our P-FSE mining approaches on a set of real-world uncertain sequence data and report the results in Section 5.3.

All the experiments were run on a computer with Intel(R) Core(TM) i6 CPU and 12GB memory. The algorithms were implemented in C++, and run on Linux.

5.1 Distinct P-FSE Mining on Synthetic Datasets

Synthetic Data Generation. We implement a data generator to create uncertain sequences based on parameters (ℓ, m, d), where ℓ is the length of the sequence, m is the maximum number of events at a time point, and d is the total number of distinct event types. For most of the experiments, the sequence of existential probabilities of an event type, $\langle \Pr(O_1), \ldots, \Pr(O_m) \rangle$, is drawn from an Uniform distribution. We also carry out experiments on sequence data with probabilities drawn from Gaussian and Zipf distributions with different parameter values.

Experimental Setting. We compare the performance of the three approaches proposed in this paper, which are respectively referred to as the **Exact**, the **Approximate** and the **Optimized**. We first evaluate the execution time of the three approaches with respect to the variations of different parameters. For the *Approximate* and the *Optimized*, we focus on examining their accuracies. Specifically, we use the standard recall and precision measures. Let $\mathcal{P}_{Exact}$ be the set of P-FSEs generated by the *Exact* approach, $\mathcal{P}_{App}$ be the set of P-FSEs produced by the approximate algorithms (i.e. the *Approximate* and the *Optimized*). The recall and

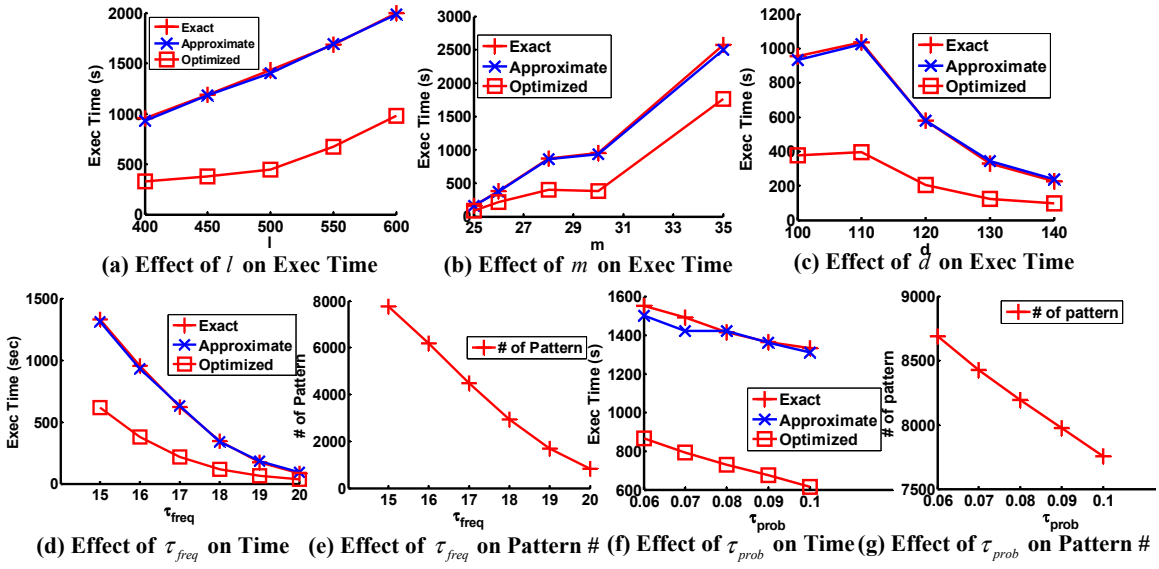


Figure 1: Scalability Results for Discovering Distinct P-FSEs

the precision are computed as follows:

$$recall = \frac{|\mathcal{P}_{Exact} \bigcap \mathcal{P}_{App}|}{|\mathcal{P}_{Exact}|} \tag{8}$$

$$precision = \frac{|\mathcal{P}_{Exact} \bigcap \mathcal{P}_{App}|}{|\mathcal{P}_{App}|} \tag{9}$$

Higher precision and recall values reflect a better accuracy.

Effect of ℓ, m and d on Execution Time. We first evaluate the effect of synthetic data generation parameters on execution time. We generate five sequences for each configuration of (ℓ, m, d), and report the run time averaged over the five corresponding runs. For the *Approximate* and the *Optimized*, the reported run times correspond to the result patterns achieving an accuracy with recall = 1 and precision > 0.95. The experimental results are summarized as follows:

- Figure 1 (a) shows the execution time of the three approaches when ℓ varies from 400 to 600, where we fix $m = 30$, $d = 100$, $\tau_{freq} = 16$ and $\tau_{prob} = 0.1$.
- Figure 1 (b) shows the execution time of the three approaches when m varies from 25 to 35, where we fix $\ell = 400$, $d = 100$, $\tau_{freq} = 16$ and $\tau_{prob} = 0.1$.
- Figure 1 (c) shows the execution time of the three approaches when d varies from 100 to 140, where we fix $\ell = 400$, $m = 30$, $\tau_{freq} = 16$ and $\tau_{prob} = 0.1$.

From these results, we observe the following trends:

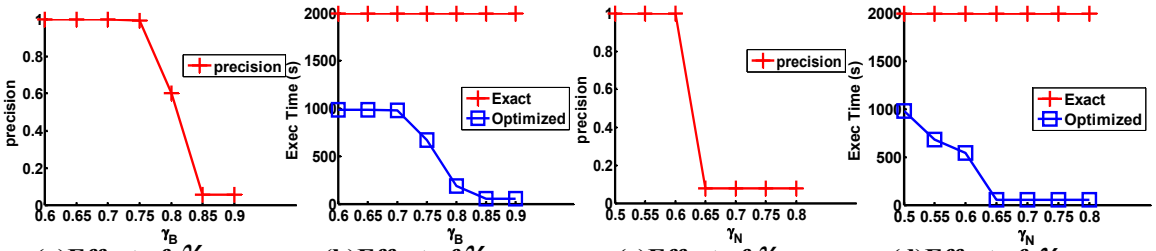
- The efficiency of the *Approximate* is similar to that of the *Exact*. The reason is that, while the *Approximate* is more efficient in frequency checking, the main cost of P-FSE mining is with the occurrence recognition.
- In all of the experiments, the *Exact* is around 2-3 times slower than the *Optimized*, which verifies the effectiveness of the pruning strategy used by the *Optimized*.
- The run time of all approaches increase with the increment of parameters ℓ and m. This trend is intuitive since larger ℓ and m imply larger data size. In particular, the run time of the *Optimized* increases almost linearly with respect to the increment of the sequence length ℓ.
- The run time of all approaches decrease with the increment of d. This is mainly because, when the length of the sequence and the mean value of events at each time point are fixed, a larger pool of event types indicates that the lengths of the P-FSEs tend to be smaller, which further means that the approaches do not have to recurse to deep levels.

Effect of τ_{freq} and τ_{prob} on Execution Time and Number of P-FSEs. We now evaluate the effect of the frequency threshold and the probability threshold on execution time as well as the number of P-FSEs. The experimental results are summarized as follows:

- Figure 1 (d) shows the run time of the three approaches when the frequency threshold τ_{freq} varies from 15 to 19, where we fix $\ell = 400$, $m = 30$, $d = 100$ and $\tau_{prob} = 0.1$. Figure 1 (e) plots the number of P-FSEs discovered by the *Exact* under the same parameter setting.
- Figure 1 (f) shows the run time of the three approaches when the probability threshold τ_{prob} varies from 0.06 to 0.1, where we fix $\ell = 400$, $m = 30$, $d = 100$ and $\tau_{freq} = 16$. Figure 1 (g) plots the number of P-FSEs discovered by the *Exact* under the same parameter setting.

From these results, we observe that,

- Both the runtime of the approaches and the size of the discovered P-FSEs decrease with the increment of τ_{freq} and τ_{prob}. This trend is intuitive since larger threshold values indicate fewer valid P-FSEs.



(a)Effect of γ_B on Precision (b)Effect of γ_B on Exec Time (c)Effect of γ_N on Precision (d)Effect of γ_N on Exec Time

Figure 2: Effect of γ_B and γ_N on precision and execution time on Discovering Distinct P-FSEs

Table 3: Accuracy w.r.t. τ_{freq}

	τ_{freq}	16	17	18	19	20
Optimized	recall	1	1	1	1	1
	precision	1	1	1	1	0.99
Approximate	recall	1	1	1	1	1
	precision	0.93	0.95	1	1	0.92

Table 4: Accuracy w.r.t. τ_{prob}

	τ_{prob}	0.06	0.07	0.08	0.09	0.1
Optimized	recall	1	1	1	1	1
	precision	1	1	1	1	1
Approximate	recall	1	1	1	1	1
	precision	0.94	1	0.96	0.98	1

- The *Optimized* is around 2-3 times faster than the *Exact*, which demonstrates again the superiority the *Optimized*.

Accuracy of Approximate Approaches. Since both the *Approximate* and the *Optimized* are not exact approaches, we conduct experiments to assess their accuracy. Table 3 shows the recall and precision of the two approaches when τ_{freq} varies from 16 to 20, where we fix $\ell = 600$, $m = 30$, $d = 100$, $\tau_{prob} = 0.1$, $\gamma_B = 0.7$, $\gamma_N = 0.5$ and $N = 50$. As we can see, both approaches can achieve 100% recall and very high precision. The precision of the *Optimized* is higher than the *Approximate*, because the approximation technique is used by the *Optimized* only to prune episodes.

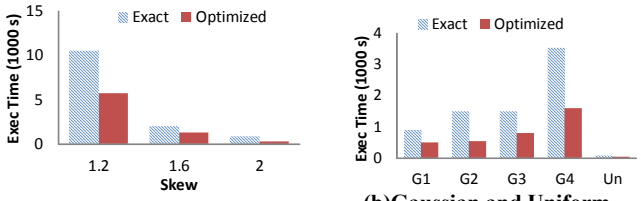
Table 4 shows the recall and precision of the two approaches when τ_{prob} varies from 0.06 to 0.1, where we fix ℓ = 600, $m = 30$, $d = 100$, $\tau_{freq} = 15$, $\gamma_B = 0.7$, $\gamma_N = 0.5$ and $N = 50$. The similar results can be observed.

Since the *Approximate* approach does not clearly outperform the *Exact* in term of the computation efficiency, and the *Approximate* is less accurate than the *Optimized*, we focus on the *Exact* and the *Optimized* in the following experiments.

Effect of Uncertainty Distributions We also examine the effect of using different distributions to generate each event's existential probability. Specifically, we experiment with the Uniform distribution, the Gaussian distribution, and the Zipf distribution with different parameters. Table 5 summarizes the mean and the standard deviation settings of the Gaussian and the Uniform distribution. For the Zipf distribution, we experiment with different skew values varying from 1.2 to 2. The run times of the *Exact* and the *Optimized*

Table 5: Existential Probability Distribution

distribution	mean	standard deviation
Gaussian G_1	0.5	0.5
Gaussian G_2	0.5	0.125
Gaussian G_3	0.5	0.289
Gaussian G_4	0.7	0.125
Uniform Un	0.5	0.289



(a)Zipf Distribution (b)Gaussian and Uniform Distribution

Figure 4: Effect of Existential Probability Distribution on Discovering Distinct Episode

are shown in Figure 4, from which we observe,

- The *Optimized* outperforms the *Exact* over all different distributions, where we make sure both the precision and recall of the *Optimized* are greater than 0.99.
- Both algorithms run relatively slower on G_4. This is because G_4 has a higher mean (0.7) and a lower standard deviation (0.125), so that it generates higher existential probability values. As a result, more P-FSEs exist in the data, which require more time to discover them all.
- For the Zipf distribution, the runtime of both approaches decrease with the increment of the skew value. This is because the Zipf distribution with a larger skew value generates fewer high existential probabilities. Consequently, fewer P-FSEs exist in the data.

Effect of γ_B and γ_N We further evaluate the effect of the error bounds on the performance of the *Optimized*. Figures 2 (a) and (b) show the precision and the execution time of the *Optimized* w.r.t. γ_B. Figures 2 (c) and (d) show the variation of the two variables w.r.t. γ_N. We omit recall here because it is always 100% for the *Optimized* approach.

We observe that, if γ_B (γ_N) is too large, the precision and the execution time of the *Optimized* will decrease simultaneously. However, if the value of γ_B (γ_N) is too small,

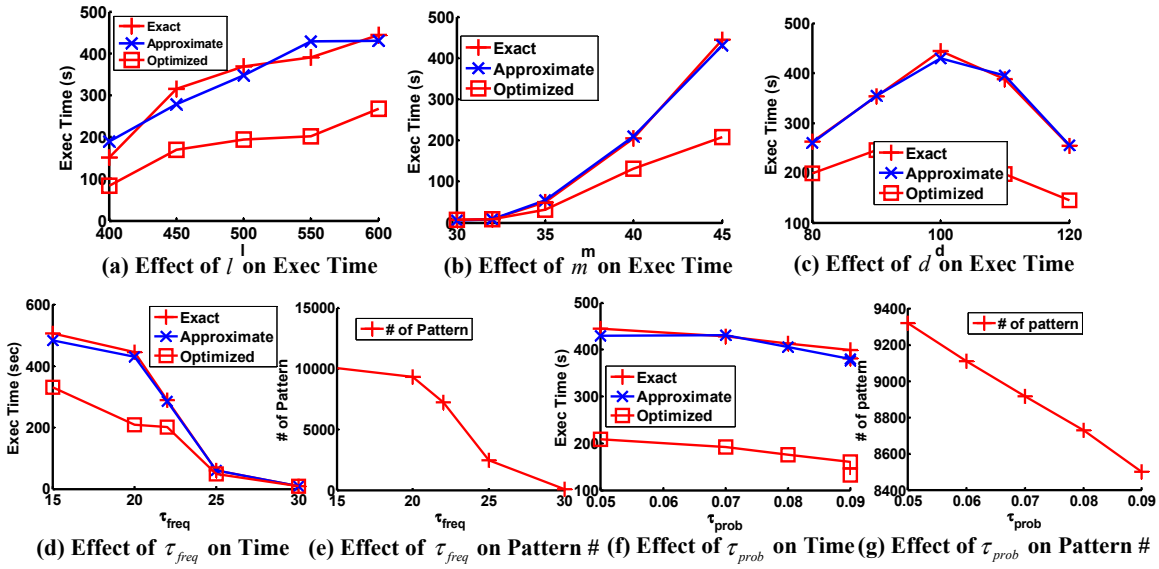


Figure 3: Scalability Results for Discovering Non-Overlapped P-FSEs

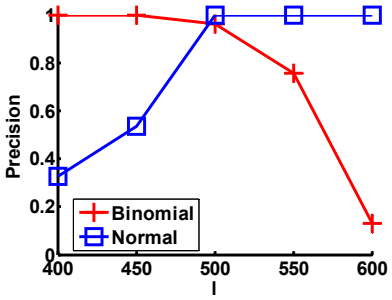


Figure 5: Binomial Approximation vs Normal Approximation

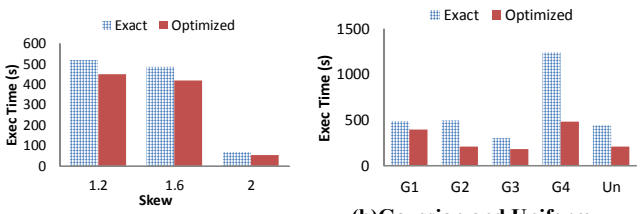


Figure 7: Effect of Existential Probability Distribution on Discovering Non-Overlapping Episode

the precision will remain as 1 while the execution time may increase. Empirically, a tradeoff may be found at where $\gamma_B = 0.7$ and $\gamma_N = 0.5$.

Effect of ℓ on probability model. As analyzed in Section 4.2, the Normal distribution based approximation may not perform well when the number of occurrences is not sufficiently high. Since we cannot directly control the number of occurrences, we experiment by varying the length of the sequence.

Figure 5 shows the precision of the Binomial approximation and that of the Normal approximation when ℓ varies from 400 to 600, where we fix $m = 30$, $d = 100$, $\tau_{freq} = 16$ and $\tau_{prob} = 0.1$. To test the precision of each approximation separately, we use only one approximation once at a time. The recall of all the results are always 1. Figure 5 reveals that the precision of the Binomial approximation decreases with the increase of the length of sequence. In contrast, the precision of the Normal approximation increases with the increase of the sequence length. This is because, when the length of sequence becomes longer, there are more occurrences of episodes. According to the analysis in Section 4.2, the Normal approximation works well when there are enough occurrences of an episode. However, when the number of occurrences of an episode is small, frequentness probability is better to be approximated by the Binomial approximation.

5.2 Non-Overlapped P-FSE Mining on Synthetic Datasets

We also conduct experiments to mine non-overlapped P-FSEs using the three proposed approaches. Similarly, we evaluate the execution time and the number of P-FSEs by varying different parameters. For each configuration of data generation parameters, we generate five datasets and report the results averaged over the fine runs. In particular,

- Figure 3 (a) shows the execution time of the three approaches when ℓ varies from 400 to 600, where we fix $m = 45$, $d = 100$, $\tau_{freq} = 20$ and $\tau_{prob} = 0.05$.
- Figure 3 (b) shows the execution time of the three approaches when m varies from 30 to 45, where we fix $\ell = 600$, $d = 100$, $\tau_{freq} = 20$ and $\tau_{prob} = 0.05$.
- Figure 3 (c) shows the execution time of the three approaches when d varies from 80 to 120, where we fix $\ell = 600$, $m = 45$, $\tau_{freq} = 20$ and $\tau_{prob} = 0.05$.
- Figure 3 (d) shows the run time of the three approaches when the frequency threshold τ_{freq} varies from 15 to 30, where we fix $\ell = 600$, $m = 45$, $d = 100$ and

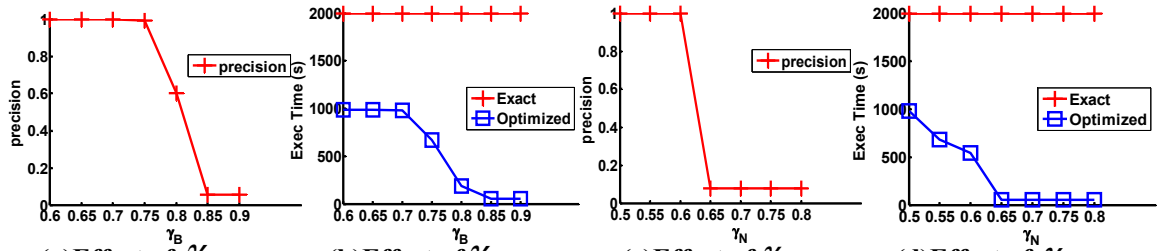


Figure 6: Effect of γ_B and γ_N on precision and execution time for Discovering Non-Overlapping P-FSEs

$\tau_{prob} = 0.05$. Figure 3 (e) shows the number of P-FSEs under the same parameter setting.

- Figure 3 (f) shows the run time of the three approaches when the probability threshold τ_{prob} varies from 0.05 to 0.09, where we fix $\ell = 600$, $m = 45$, $d = 100$ and $\tau_{freq} = 20$. Figure 3 (g) displays the number of P-FSEs under the same parameter setting.

The trends demonstrated by these results are similar to those in the distinct P-FSE mining. Therefore, we omit summarizing the observations and analyzing the reasons.

We also study the effect of the existential probability distributions, the error bounds (γ_B and γ_N) on discovering non-overlapped P-FSEs. The results of the effect of different existential probability distributions are illustrated in Figure 7. Figure 6 shows the results by varying the γ_B and γ_N. Again, the trends displayed in these results are similar to those in the distinct P-FSE mining.

5.3 P-FSE Mining on Customer Behavior Sequences

In this subsection, we apply the *Exact* approach and the *Optimized* approach on a real-world data set. Our real data is collected from an equipment hire company, which supplies more than 130 different models of equipments to customers. We model each customer's hiring behaviors as a sequence, where each event corresponds to an equipment hired by the customer. We consider the sequences containing more than 100 events in the period from Jan 1, 2010 to Dec 31, 2011. Each event in a sequence is assigned an existential probability, which is obtained according to the customers' hiring histories. For instance, the equipment 'Mini-Excavator' belongs to a model 'excavator', which may contain other equipments such as 'Mid-Excavator'. If a customer has hired an equipment from the model 'excavator' 30 times since year 2010, out of which the customer has hired the equipment 'Mini-Excavator' for 15 times, we assign the probability 50% to the equipment to reflect the uncertainty. For example, a customer who prefers to hire the 'Mini-Excavator' may hire the 'Mid-Excavator' when the former is temporarily not available.

Overall, we have collected 6301 sequences, where the maximum length of a sequence is 320 and the average length is 121. The average number of events at a time point is 12, and the average number of equipment models in a sequence is 30. On average, around 20% of the events in a sequence are uncertain.

We run the *Exact* and the *Optimized* on the set of real data. Figures 8 and 9 respectively report the performance of the two approaches in mining distinct P-FSEs and non-overlapped P-FSEs on the longest sequence with 320 events, by varying the frequency threshold and the probability threshold. For the *Optimized*, the performance corresponds to the result patterns of the best accuracy that can be achieved. The trends shown in the figures comply with those on synthetic data.

Besides showing P-FSEs can be mined efficiently from real data, we also discover some interesting episodes. For example, the company is interested in an episode $\alpha =$("heater" $\rightarrow$ "excavator"), with the frequentness probability $\Pr(freq(\alpha) \geq 20) = 70\%$, since "heater" is a low profit equipment while the profit of "excavator" is high. Based on this episode, customized promoting strategies might be designed for the customer. We expect that our P-FSE mining algorithms would also be useful in many other real world applications involving uncertain data.

6. CONCLUSION

Frequent serial episode mining is an important tool to discover interesting and useful temporal correlations from sequential data. The inherent data uncertainty issue in many domains call for the need to discover frequent episodes over uncertain sequence data. In this paper, we focus on the problem of probabilistic frequent serial episode mining. To address the data uncertainty, we define the measure of frequentness probability of each episode under the possible world semantics. We develop two P-FSE mining algorithms, which respectively compute the exact frequentness probability and approximate the frequentness probability. We further devise an optimized approach that prunes candidate episodes early by estimating the upper bounds of their frequentness probabilities. We carry out extensive experiments to evaluate the performance of the proposed approaches. Our experimental results reveal the superiority of the optimized approach. We also find that the Binomial distribution is better than the Normal distribution when there are not enough occurrences of episodes in an uncertain sequence.

7. ACKNOWLEDGMENTS

This work was supported, in part, by the Australian Research Council (ARC) Linkage Project under Grant No. LP120100566, and UTS ECRG.

8. REFERENCES

[1] A. Achar, S. Laxman, and P. S. Sastry. A unified view of the apriori-based algorithms for frequent episode discovery. *Knowl. Inf. Syst.*, 31(2):223–250, 2012.

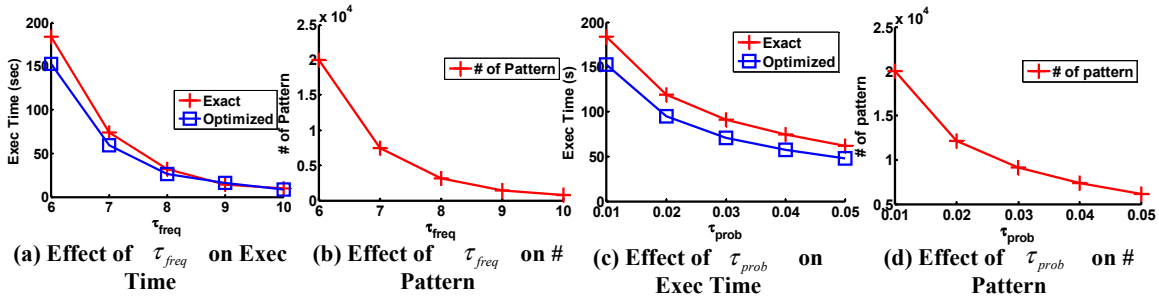


Figure 8: Scalability Results for Distinct P-FSE Mining on Customer Hire Sequence

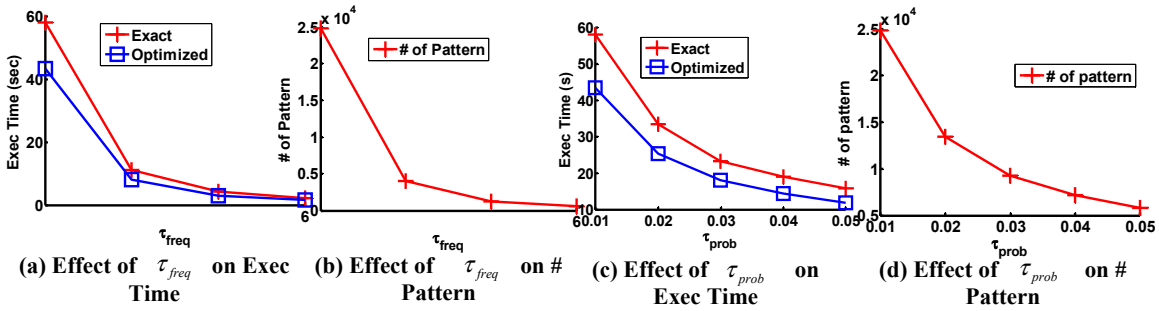


Figure 9: Scalability Results for Non-overlapped P-FSE Mining on Customer Hire Sequence

[2] C. C. Aggarwal, Y. Li, J. Wang, and J. Wang. Frequent pattern mining with uncertain data. In *KDD*, pages 29–38, 2009.

[3] C. C. Aggarwal and P. S. Yu. A survey of uncertain data algorithms and applications. *IEEE Trans. Knowl. Data Eng.*, 21(5):609–623, 2009.

[4] R. Agrawal and R. Srikant. Mining sequential patterns. In *ICDE*, pages 3–14, 1995.

[5] T. Bernecker, H.-P. Kriegel, M. Renz, F. Verhein, and A. Züfle. Probabilistic frequent itemset mining in uncertain databases. In *KDD*, pages 119–128, 2009.

[6] J. Bi and T. Zhang. Support vector classification with input data uncertainty. In *NIPS*, 2004.

[7] T. Calders, C. Garboni, and B. Goethals. Approximation of frequentness probability of itemsets in uncertain data. In *ICDM*, pages 749–754, 2010.

[8] C. K. Chui, B. Kao, and E. Hung. Mining frequent itemsets from uncertain data. In *PAKDD*, pages 47–58, 2007.

[9] K. Iwanuma, Y. Takano, and H. Nabeshima. A unified view of the apriori-based algorithms for frequent episode discovery. In *IEEE Conference on Cybernetics and Intelligent Systems*, pages 213–217, 2004.

[10] G. Karypis, M. V. Joshi, and V. Kumar. *A Universal formulation of sequential patterns*. Technical Report TR99-021, Department of Computer Science, University of Minnesota, Minneapolis, 1999.

[11] H.-P. Kriegel and M. Pfeifle. Density-based clustering of uncertain data. In *KDD*, pages 672–677, 2005.

[12] S. Laxman. *Discovering frequent episodes: fast algorithms,connections with HMMs and generalizations*. PhD thesis, Banalore, India, 2006.

[13] S. Laxman, P. S. Sastry, and K. P. Unnikrishnan. Discovering frequent episodes and learning hidden markov models: A formal connection. *IEEE Trans. Knowl. Data Eng.*, 17(11):1505–1517, 2005.

[14] S. Laxman, V. Tankasali, and R. W. White. Stream prediction using a generative model based on frequent episodes in event sequences. In *KDD*, pages 453–461, 2008.

[15] C. K.-S. Leung, M. A. F. Mateo, and D. A. Brajczuk. A tree-based approach for frequent pattern mining from uncertain data. In *PAKDD*, pages 653–661, 2008.

[16] H. Mannila, H. Toivonen, and A. I. Verkamo. Discovery of frequent episodes in event sequences. *Data Min. Knowl. Discov.*, 1(3):259–289, 1997.

[17] M. Muzammal and R. Raman. Mining sequential patterns from probabilistic databases. In *PAKDD (2)*, pages 210–221, 2011.

[18] W. K. Ngai, B. Kao, C. K. Chui, R. Cheng, M. Chau, and K. Y. Yip. Efficient clustering of uncertain data. In *ICDM*, pages 436–445, 2006.

[19] B. Qin, Y. Xia, S. Prabhakar, and Y.-C. Tu. A rule-based classification algorithm for uncertain data. In *ICDE*, pages 1633–1640, 2009.

[20] J. Ren, S. D. Lee, X. Chen, B. Kao, R. Cheng, and D. W.-L. Cheung. Naive bayes classification of uncertain data. In *ICDM*, pages 944–949, 2009.

[21] L. Sun, R. Cheng, D. W. Cheung, and J. Cheng. Mining uncertain data with probabilistic guarantees. In *KDD*, pages 273–282, 2010.

[22] Y. Tong, L. Chen, Y. Cheng, and P. S. Yu. Mining frequent itemsets over uncertain databases. *PVLDB*, 5(11):1650–1661, 2012.

[23] K. P. Unnikrishnan, B. Q. Shadid, P. S. Sastry, and S. Laxman. *Root cause diagnostics using temporal data mining*. Patent Number(s) US 7509234, 2009.

[24] L. Wang, R. Cheng, S. D. Lee, and D. W.-L. Cheung. Accelerating probabilistic frequent itemset mining: a model-based approach. In *CIKM*, pages 429–438, 2010.

[25] Z. Zhao, D. Yan, and W. Ng. Mining probabilistically frequent sequential patterns in uncertain databases. In *EDBT*, pages 74–85, 2012.