IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received April 20, 2018, accepted May 16, 2018, date of publication May 29, 2018, date of current version June 20, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2839751

A Weighted Frequent Itemset Mining Algorithm for Intelligent Decision in Smart Systems

XUEJIAN ZHAO[1], XINHUI ZHANG[2], PAN WANG[1], SONGLE CHEN[1], AND ZHIXIN SUN[1]

[1]Key Laboratory of Broadband Wireless Communication and Sensor Network Technology of Ministry of Education, Nanjing University of Posts and Telecommunications, Nanjing 210003, China

[2]College of the Internet of Things, Nanjing University of Posts and Telecommunications, Nanjing 210003, China

Corresponding author: Zhixin Sun (sunzx@njupt.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61373135, Grant 61300240, and Grant 61672299, in part by the Natural Science Foundation of Jiangsu Province of China under Grant BK20140883, in part by the China Postdoctoral Science Foundation funded project under Grant 2015M581844, in part by the Jiangsu Planned Projects for Postdoctoral Research Funds under Grant 1501125B, and in part by NUPTSF under Grant NY214101.

ABSTRACT Intelligent decision is the key technology of smart systems. Data mining technology has been playing an increasingly important role in decision-making activities. Frequent itemset mining (FIM), as an important step of association rule analysis, is becoming one of the most important research fields in data mining. Weighted FIM in uncertain databases should take both existential probability and importance of items into account in order to find frequent itemsets of great importance to users. However, the introduction of weight makes the weighted frequent itemsets not satisfy the downward closure property any longer. As a result, the search space of frequent itemsets cannot be narrowed according to downward closure property which leads to a poor time efficiency. In this paper, the weight judgment downward closure property for the weighted frequent itemsets and the existence property of weighted frequent subsets are introduced and proved first. Based on these two properties, the Weight judgment downward closure property-based FIM (WD-FIM) algorithm is proposed to narrow the searching space of the weighted frequent itemsets and improve the time efficiency. Moreover, the completeness and time efficiency of WD-FIM algorithm are analyzed theoretically. Finally, the performance of the proposed WD-FIM algorithm is verified on both synthetic and real-life data sets.

INDEX TERMS Frequent itemset mining, weight judgment, downward closure property, intelligent decision, smart system, data mining.

I. INTRODUCTION

Intelligent decision is the key technology of smart systems. Data mining technology has been playing an increasingly important role in decision making activities. FIM (Frequent Itemset Mining), as one of the most hot research topics in data mining, is an important approach to discover association rules in datasets [1], [2], that is widely used in the field of precision marketing, personalized recommendation, network optimization, medical diagnosis and so on. So far, many perfect and mature FIM algorithms have been proposed for binary databases. However, with the rapid development of data acquisition and data processing technologies, various forms of complex data have emerged, like uncertain data.

Uncertain data means the existence of an item in a transaction is described by a likelihood measure or a probability [3]. As is known, if we adopt a binary data model, then each item in a transaction can either be present or absent. However, in the uncertain data model, the existence of an item in a transaction can be indicated by a probability, thus it allows more information to be captured by the dataset which can lead to more accurate analytical results. However, each coin has two sides. Uncertain data model still have its drawbacks. The first disadvantage is that the size of the dataset would be much larger because of the storage of existence probability. Another disadvantage is that the mining algorithms for uncertain databases will be more complicated and time consuming. Thus, developing efficient mining algorithms for uncertain

databases has become a hot research topic in recent years. Many algorithms have been developed to mine frequent itemsets in uncertain databases.

Most existing studies assume that all the items in uncertain databases have the same importance. However, in actual reality, the values and importances of various items are usually different to users. For example, the profits of a costly luxury goods and a cheap living goods cannot be mentioned in the same breath. Consequently, the mining based on only occurrence frequencies or existence probabilities without taking importances or values of items into account is insufficient to identify useful and meaningful patterns. To address this issue, a prominent solution is to let the users assign different weights to items to indicate their relative importances or values. The weight of items can be set by the users according to their professional domain knowledge or specific application requirements to indicate profits, risks, costs and so on. In this context, itemsets with high importances for the users will be discovered. Moreover, the introduction of weights of items can greatly reduce the number of frequent itemsets. However, the downward closure property used for mining frequent itemsets in uncertain databases would not hold any longer because different weights are assigned to items. This means an infrequent itemset may have a frequent superset. As a result, the searching space cannot be narrowed according to the downward closure property any longer which will lead to low time efficiency of FIM algorithms.

In this paper, on the basis of the weight judgment downward closure property, the WD-FIM (Weight judgment Downward closure property based Frequent Itemset Mining) algorithm is proposed to narrow the searching space of weighted frequent itemsets and improve the time efficiency. Consequently, more useful and meaningful weighted frequent itemsets in uncertain databases can be discovered. The main contributions of this paper are listed as following.

1. The weight judgment downward closure property and the existence property of weighted frequent subsets for uncertain databases are introduced and proved. The weight judgment downward closure property can be used to narrow the searching space of weighted frequent itemsets. The existence property of weighted frequent subsets can ensure all the weighted frequent itemsets be discovered.
2. The WD-FIM algorithm is proposed on the basis of weight judgment downward closure property to narrow the searching space of weighted frequent itemsets and improve the time efficiency.
3. A considerable amount of experiments are conducted on both real-life and synthetic datasets to evaluate the performance of the proposed WD-FIM algorithm in terms of runtime, number of patterns and memory consumption.

The remainder of this paper is organized as follows. Related works is reviewed in Section 2. The preliminaries and problem statement are given in Section 3. In section 4, the related properties are introduced and proved theoretically, and the WD-FIM algorithm is described in detail. Additionally, the completeness and time efficiency of WD-FIM algorithm are also analyzed in this section. Experimental results are discussed in Section 5. Finally, a conclusion is drawn and future work are discussed in Section 6.

II. RELATED WORKS

In this section, related works on frequent itemset mining in uncertain databases and weighted frequent itemset mining in binary databases are briefly reviewed.

A. FREQUENT ITEMSET MINING IN UNCERTAIN DATABASES

With the popular use of various data acquisition and communication technologies, a huge amount of data stored in a database may be inaccurate, imprecise, or incomplete in real-life applications, such as wireless sensor network applications or location-based services [4], [5]. To address this issue, developing efficient algorithms to mine patterns in uncertain databases has become a major research topic in recent years and many efficient FIM algorithms for uncertain databases have been proposed. These algorithms can be generally classified into two categories: candidate generate-and-test based uncertain frequent itemset mining and pattern-growth mining.

One way to mine frequent itemsets from uncertain data is to apply the candidate generate-and-test paradigm. For example, Chui *et al.* [6] proposed U-Apriori algorithm which applies the candidate generate-and-test process to mine frequent itemsets from for uncertain data. Similar to Apriori algorithm for mining precise data, U-Apriori algorithm needs to scan the database frequently and generates a large number of candidate frequent itemsets. Chui and Kao [7] applied the decremental pruning technique to further improve the efficiency of U-Apriori. MBP [8] is an approximation method for uncertain frequent pattern mining based on statistical techniques. IMBP [9] was proposed to more improve the mining speed and memory efficiency of MBP at the cost of losing accuracy.

An alternative to candidate generate-and-test based mining is pattern-growth mining, which avoids generating a large number of candidates [10]. Commonly used pattern-growth mining paradigms are mostly based on hyperlinked structures or tree structures. For example, Aggarwal *et al.* [11] proposed a hyperlinked structure based algorithm called UH-mine to mine frequent patterns from uncertain data. Leung *et al.* [12] proposed a tree-based mining algorithm called UF-growth which also constructs a tree structure to store the contents of the uncertain datasets, like its counterpart - the FP-growth algorithm [13] for mining precise data. In order to reduce the tree size, Aggarwal *et al.* [11] proposed the UFP-growth algorithm. To further reduce the tree size, Leung and Tanbeer [14] proposed an uncertain frequent pattern mining algorithm called CUFgrowth, which builds a new tree structure called CUF-tree. Leung and Tanbeer [15] introduced the concept of a prefixed item cap and proposed

PUF-growth algorithm to mine uncertain frequent patterns which runs faster than CUF-growth. TPC-growth [16] is an advanced version of PUF-growth. It employs an upgraded overestimation method that can tighten upper bounds to expected supports more than PUF-growth. CUFP-Mine [17] is a method for mining exact uncertain frequent patterns without employing recursive call-based pattern growth manners. However, the larger the given database is, the worse the mining performance of CUFP-Mine becomes. AT-Mine [18] is another tree-based efficient approach proposed to overcome the fatal problems of CUFP-Mine. It guarantees more efficient mining performance than that of CUFP-Mine, but it still has limitations in runtime and memory performance aspects. U-WFI [19] is a tree-based approach that applies weight factors into uncertain pattern mining. Through weight constraints, the algorithm can find more meaningful uncertain frequent patterns but have limitations in the aforementioned aspects.

State-of-the-art algorithms based on tree structures can cause fatal problems in terms of runtime and memory usage according to the characteristics of uncertain databases and threshold settings because their own tree data structures can become excessively large and complicated in their mining processes. Various approaches have been suggested to overcome such problems. For example, Lee and Yun [20] propose LUNA algorithm which is an exact, efficient algorithm for mining uncertain frequent patterns based on newly proposed list-based data structures and pruning techniques, which can also guarantee a complete set of uncertain frequent patterns to be mined more efficiently without pattern losses.

B. WEIGHTED FREQUENT ITEMSET MINING IN UNCERTAIN DATABASES

Traditional frequent itemset mining methods have a problem that it does not apply importance of each item obtained from the real world into the mining process. In order to discover more useful and interesting patterns, numerous algorithms have been developed for weighted frequent itemset mining. However, most of these algorithms are proposed for precise datasets or data streams, for example, WAR (Weighted Association Rules) algorithm [21], WARM (Weighted Association Rule Mining) algorithm [22], WFIM (Weighted Frequent Itemset Mining) algorithm [23], WSpan algorithm [24], WMFP-SW (Weighted Maximal Frequent Pattern mining over data streams based on Sliding Window model) algorithm [25], MWS (Maximal frequent pattern mining with Weight conditions over data Streams) algorithm [26], WEP (Weighted Erasable Patterns) mining algorithm [27] and so on. Mining weighted frequent itemsets in uncertain databases have only a few researches. To our knowledge, only two algorithms are proposed to discover weighted frequent itemsets in uncertain datasets. Lee *et al.* [19] suggested a new tree-based U-WFI (Uncertain Mining of Weighted Frequent Itemsets) algorithm which can mine uncertain frequent itemsets considering item weights from a given uncertain database. As a result, more meaningful itemsets with high importance and existential probabilities can be discovered effectively. Lin*et al.* proposed HEWI-Uapriori (High Expected Weighted Itemset) algorithm to mine high expected weighted itemsets based on high upper-bound expected weighted downward closure property to early prune the search space and unpromising itemsets [28]. Consequently, further research should be conducted to improve the efficiency of mining frequent itemset in uncertain databases.

III. PRELIMINARIES AND PROBLEM STATEMENT

In this section, preliminary definitions are given first and the problem of mining weight frequent itemsets are stated formally.

A. PRELIMINARIES

Let DS be the uncertain dataset to be analyzed which is composed of a set of transactions, i.e. $DS = \{T_1, T_2, \ldots, T_n\}$, n is the number of transactions in the uncertain dataset D. There are a finite set of m distinct items $I = \{I_1, I_2, \ldots, I_m\}$ in the uncertain dataset DS. Each transaction $T_q \in DS, q \in \{1, 2, \ldots, n\}$ is a subset of I, and q is the unique identifier of transaction T_q, also called its TID (Transaction IDentifier). According to the probabilistic model [11] that commonly used for uncertain frequent pattern mining, the uncertainty can be expressed in terms of the existential probability $p(I_j, T_q)$, which indicates that the item I_j exists in T_q with a probability $p(I_j, T_q)$. The existential probability $p(I_j, T_q)$ ranges from a positive value close to 0 (indicating that I_j has an insignificantly low chance to be present in T_q) to a value of 1 (indicating that I_j is definitely present in T_q). In order to assign different importances to distinct items in DS, a weight table is defined as $wtable = \{w(I_1), w(I_2), , w(I_{\mathrm{m}})\}$, where $w(I_{\mathrm{j}}) \in [0, 1], j \in \{1, 2, \ldots, \mathrm{m}\}$ is the weight of the item I_{j}. If an itemset X contains k distinct items, then X is called a $k-$itemset. If $X \subseteq T_q$, we can say that the itemset X is contained in the transaction T_q. The minimum expected weighted support threshold is $\varepsilon \in (0, 1]$. Table 1 is an example of uncertain databases, which consists of 10 transactions and 6 items. Table 2 is the weight table of the items in Table 1.

TABLE 1. The uncertain dataset *DS*.

TID	Transaction{(Item, Probability)}
1	(*A*, 0.8), (*B*, 0.4), (*D*, 1.0)
2	(*B*, 0.3), (*F*, 0.7)
3	(*B*, 0.7), (*C*, 0.9), (*E*, 1.0), (*F*, 0.7),
4	(*E*, 1.0), (*F*, 0.5)
5	(*A*, 0.6), (*C*, 0.4), (*D*, 1.0)
6	(*A*, 0.8), (*B*, 0.8), (*C*, 1.0), (*F*, 0.3)
7	(*A*, 0.8), (*C*, 0.9), (*D*, 0.5), (*E*, 1.0)
8	(*C*, 0.6), (*E*, 0.4)
9	(*A*, 0.5), (*D*, 0.8), (*F*, 1.0)
10	(*A*, 0.7), (*B*, 1.0), (*C*, 0.9), (*E*, 0.8)

In this section, the example showed in Table 1 are use to explain the preliminary definitions. And the minimum expected weighted support threshold ε is set to 0.1.

Definition 1 (Item Weight): The item weight is a value used to describe the importance of an item, which is set by the users according to their preferences or application scenarios. The weight of item I_j is denoted as $w(I_j)$, where $w(I_j) \in (0, 1]$.

In the given example of Table 2, the weight of item A and item D are $w(A) = 0.1$ and $w(D) = 1.0$ respectively.

TABLE 2. The weight table.

Item	A	B	C	D	E	F
Weight	0.1	0.8	0.3	1.0	0.6	0.1

Definition 2 (Itemset Weight): The weight of itemset X is denoted as $w(X)$, which is the average of the weights of all items contained in itemset X. It can be formally defined as:

$$w(X) = \frac{\sum_{I_j \in X} w(I_j)}{|k|}, \tag{1}$$

where I_j is an item in itemset X, and $|k|$ is the number of items in itemset X.

In the given example of Table 2, the weight of itemset (AD) is $w(AD) = (w(A) + W(D))/2 = 0.55$.

Definition 3 (Itemset Probability in a Transaction): The existential probability of an itemset X in a transaction T_q is denoted as $p(X, T_q)$, which is the product of existential probabilities of the items contained in itemset X. It can be formally defined as:

$$p(X, T_q) = \prod_{I_j \in X} p(I_j, T_q) \tag{2}$$

where $P(I_j, T_q)$ indicates the existential probability of item I_j in T_q.

In the given example of Table 1 and Table 2, the existential probability of itemset (AD) in transaction T_1 is $p(AD, T_1) = p(A, T_1) \times p(D, T_1) = 0.8 \times 1.0 = 0.8$.

Definition 4 (Expected Support of an Itemset in DS): The expected support of itemset X in dataset DS is denoted as $expSup(X)$. It is the summation of existential probabilities of itemset X in all the transactions which contain itemset X. It can be formally defined as:

$$\begin{aligned} \exp Sup(X) &= \sum_{X \subseteq T_q \wedge T_q \in DS} p(X, T_q) \\ &= \sum_{X \subseteq T_q \wedge T_q \in DS} (\prod_{I_j \in X} p(I_j, T_q)). \end{aligned} \tag{3}$$

In the given example of Table 1 and Table 2, the expected support of itemset (AD) in dataset DS is $expSup(AD) = p(AD, T_1) + p(AD, T_5) + p(AD, T_7) + p(AD, T_9) = 0.8 + 0.6 + 0.4 + 0.4 = 2.2$.

Definition 5 (Frequent Itemset in DS): In uncertain dataset DS, if the expected support of itemset X is greater than or equal to the minimum expected support (the minimum expected support is the product of the minimum expected support threshold δ and the number of transactions in the uncertain dataset $|DS|$), i.e. $expSup(X) \geq \delta \times |DS|$, then itemset X is a frequent itemset.

Definition 6 (Expected Weighted Support of an Itemset in DS): The expected weight support of itemset X in dataset DS is denoted as $expwSup(X)$. It is the product of the expected support of itemset X and the weight of itemset X. It can be formally defined as:

$$\begin{aligned} \exp wSup(X) &= w(X) \times \exp Sup(X) \\ &= \frac{\sum_{I_j \in X} w(I_j)}{|k|} \times \sum_{X \subseteq T_q \wedge T_q \in DS} (\prod_{I_j \in X} p(I_j, T_q)) \end{aligned} \tag{4}$$

In the given example of Table 1 and Table 2, the expected weighted support of itemset (AD) in dataset DS is $expwSup(AD) = w(AD) \times expSup(X) = 0.55 \times 2.2 = 1.21$.

Definition 7 (Weighted Frequent Itemset in DS): In uncertain dataset DS, if the expected weighted support of itemset X is greater than or equal to the minimum expected weighted support (the minimum expected weighted support is the product of the minimum expected weighted support threshold ε and the number of transactions in the uncertain dataset $|DS|$), i.e. $expwSup(X) \geq \varepsilon \times |DS|$, then itemset X is a weighted frequent itemset.

In the given example of Table 1 and Table 2, when both of the minimum expected support threshold δ and the minimum expected weighted support threshold ε are set to 0.1, the frequent itemsets and weighted frequent itemsets are listed in Table 3. As is shown in Table 3, the frequent itemsets considering only the existential probability are different from the weighted frequent itemsets taking both the existential probability and weight into account. For example, itemset (AB) is a frequent itemset but not a weighted frequent itemset. We can also see from Table 3 that the downward closure property does not apply to weighted frequent itemsets any more. For example, itemset (A) is not a weighted frequent itemset. However, itemset (AD) which is a superset of itemset (A) is a weighted frequent itemset. Consequently, the downward closure property cannot be applied directly to narrow the searching space of weighted frequent itemsets.

TABLE 3. The frequent itemset and weighted frequent itemset table.

Itemset	Weight	Frequency	expSup	isFI	expWSup	isWFI
(A)	0.10	6	4.20	Y	0.42	N
(B)	0.80	5	3.20	Y	2.56	Y
(C)	0.30	6	4.70	Y	1.41	Y
(D)	1.00	4	3.30	Y	3.30	Y
(E)	0.60	5	4.20	Y	2.52	Y
(F)	0.10	5	3.20	Y	0.32	N
(AB)	0.45	3	1.66	Y	0.747	N
(AC)	0.20	4	2.39	Y	0.478	N
(AD)	0.55	4	2.20	Y	1.21	Y
(AE)	0.35	2	1.36	Y	0.476	N
(BC)	0.55	3	2.33	Y	1.2815	Y
(BE)	0.70	2	1.50	Y	1.05	Y
(CE)	0.45	3	2.76	Y	1.242	Y
(EF)	0.35	1	1.20	Y	0.42	N
(ABC)	0.40	2	1.27	Y	0.508	N
(ACE)	0.333	2	1.225	Y	0.408	N
(BCE)	0.567	2	1.35	Y	0.765	N

B. PROBLEM STATEMENT

Based on the above preliminary definitions, the problem of mining weighted frequent itemsets in uncertain databases can be formulated as following:

The uncertain database to be analyzed is DS, the user-specified weights of the items in DS are defined in *wtable*, and the user-specified minimum expected weighted support threshold is ε. The problem of mining weighted frequent itemsets in the uncertain database DS is to discover the weighted frequent itemsets considering both the weight and the existential probability constraints. An itemset X is a weighted frequent itemset if the expected weighted support of an itemset X is greater than or equal to the minimum expected weighted support, i.e. $expwSup(X) \geq \varepsilon \times |DS|$.

As mentioned above, the downward closure property cannot be applied directly to narrow the searching space of mining weighted frequent itemsets. Consequently, how to improve the time efficiency of mining weighted frequent itemsets is a major issue for urgent solution.

IV. WEIGHT JUDGMENT DOWNWARD CLOSURE PROPERTY AND WD-FIM ALGORITHM

In this section, the weight judgment downward closure property and the existence property of weighted frequent subsets are first proposed and proved. Moreover, the WD-FIM algorithm is proposed and described in detail.

A. WEIGHT JUDGMENT DOWNWARD CLOSURE PROPERTY

Theorem 1 (Downward Closure Property in Uncertain Dataset): In uncertain dataset DS, if an itemset X is a frequent itemset, any subset of X is also a frequent itemset.

Proof: Let δ denotes the minimum expected support threshold. Since itemset X is a frequent itemset, according to definition 4 and 5, it follows that:

$$\begin{aligned}\exp Sup(X) &= \sum_{X\subseteq T_q \wedge T_q \in DS} p(X, T_q) \\ &= \sum_{X\subseteq T_q \wedge T_q \in DS} \left(\prod_{I_j \in X} p(I_j, T_q)\right) \\ &\geq \delta \times |DS|.\end{aligned}$$

Let X' denotes an arbitrary nonempty subset of itemset X, i.e. $X' \subseteq X$ and $X' \neq \emptyset$, then it follows that:

$$\begin{aligned}\exp Sup(X') &= \sum_{X'\subseteq T_q \wedge T_q \in DS} p(X', T_q) \\ &= \sum_{X'\subseteq T_q \wedge T_q \in DS} \left(\prod_{I_j \in X'} p(I_j, T_q)\right) \\ &\geq \sum_{X'\subseteq T_q \wedge T_q \in DS} \left(\prod_{I_j \in X} p(I_j, T_q)\right) \\ &\geq \sum_{X\subseteq T_q \wedge T_q \in DS} \left(\prod_{I_j \in X} p(I_j, T_q)\right) \\ &\geq \delta \times |DS|.\end{aligned}$$

Therefore, it follows that X' is a frequent itemset.

Theorem 1 is the downward closure property in uncertain dataset, which is widely used to narrow the searching space of discovering the expected support frequent itemsets in previous studies [6], [11], [12], [17].

Theorem 2 (Unfitness of Downward Closure Property in Weighted Uncertain Dataset): In weighted uncertain dataset DS, if an itemset X is a weighted frequent itemset, the subset of X may not be a weighted frequent itemset.

Proof: Let $I = \{I_1, I_2, \ldots, I_m\}$ be the items in weighted uncertain dataset DS, $w(I_j), j \in \{1, 2, \ldots, m\}$ is the weight of the item I_j, and ε denotes the minimum expected weighted support threshold. Since itemset X is a weighted frequent itemset, according to definition 6 and 7, it follows that:

$$\begin{aligned}\exp wSup(X) &= w(X) \times \sum_{X\subseteq T_q \wedge T_q \in DS} p(X, T_q) \\ &= \frac{\sum_{I_j \in X} w(I_j)}{|k|} \times \sum_{X\subseteq T_q \wedge T_q \in DS} \left(\prod_{I_j \in X} p(I_j, T_q)\right) \\ &\geq \varepsilon \times |DS|.\end{aligned}$$

Let X' denotes an arbitrary nonempty subset of itemset X, i.e. $X' \subseteq X$ and $X' \neq \emptyset$, then it follows that:

$$\begin{aligned}\sum_{X'\subseteq T_q \wedge T_q \in DS} p(X', T_q) &= \sum_{X'\subseteq T_q \wedge T_q \in DS} \left(\prod_{I_j \in X'} p(I_j, T_q)\right) \\ &\geq \sum_{X\subseteq T_q \wedge T_q \in DS} \left(\prod_{I_j \in X} p(I_j, T_q)\right).\end{aligned}$$

However, the weight of itemset X' can be very small, i.e.:

$$w(X') = \frac{\sum_{I_j \in X'} w(I_j)}{|k'|} \to 0.$$

Thus:

$$\exp wSup(X') = w(X') \times \sum_{X'\subseteq T_q \wedge T_q \in DS} p(X', T_q) \to 0.$$

Therefore, it follows that X' is not a weighted frequent itemset.

Corollary 1: In weighted uncertain dataset DS, if an itemset X is not a weighted frequent itemset, the superset of X can be a weighted frequent itemset.

Corollary 1 can be easily proved according to Theorem 2, the detailed proof is not given in this paper.

Corollary 2: In weighted uncertain dataset DS, if the minimum expected support threshold δ equals the minimum expected weighted support threshold ε, the set containing all weighted frequent itemsets is a subset of the set containing all frequent itemsets.

Proof: Let $I = \{I_1, I_2, \ldots, I_m\}$ be the items in DS, $w(I_j), j \in \{1, 2, \ldots, m\}$ denotes the weight of the item I_j, and itemset X is an arbitrary nonempty element of the set containing all weighted frequent itemsets, then it

follows that:

$$\begin{aligned}\exp wSup(X) &= w(X) \times \sum_{X \subseteq T_q \wedge T_q \in DS} p(X, T_q) \\ &= \frac{\sum_{I_j \in X} w(I_j)}{|k|} \times \sum_{X \subseteq T_q \wedge T_q \in DS} \left(\prod_{I_j \in X} p(I_j, T_q)\right) \\ &\geq \varepsilon \times |DS|.\end{aligned}$$

Since $w(I_j) \in [0, 1]$, thus:

$$\sum_{X \subseteq T_q \wedge T_q \in DS} p(X, T_q) = \sum_{X \subseteq T_q \wedge T_q \in DS} \left(\prod_{I_j \in X} p(I_j, T_q)\right) \geq \delta \times |DS|.$$

Therefore, it follows that itemset X must be a frequent itemset. Consequently, it is proved that the set containing all weighted frequent itemsets is a subset of the set containing all frequent itemsets.

Theorem 2 and its two corollaries indicate that the downward closure property cannot be used directly to narrow the searching space of weighted frequent itemsets.

Theorem 3 (Weight Judgment Downward Closure Property): Let $I = \{I_1, I_2, \ldots, I_m\}$ be the items in uncertain dataset DS, $w(I_j), j \in \{1, 2, \ldots, m\}$ is the weight of item I_j, and itemset X is not a weighted frequent itemset. If the weight of item I_s is no bigger than the weight of itemset X, i.e. $w(I_s) \leq w(X)$, then the superset composed of itemset X and 1-itemset $\{I_s$ cannot be a weighted frequent itemset.

Proof: Let ε be the user specified minimum expected weighted support threshold. Since itemset X is not a weighted frequent itemset, according to definition 6 and 7, it follows that:

$$\begin{aligned}\exp wSup(X) &= w(X) \times \sum_{X \subseteq T_q \wedge T_q \in DS} p(X, T_q) \\ &= \frac{\sum_{I_j \in X} w(I_j)}{|k|} \times \sum_{X \subseteq T_q \wedge T_q \in DS} \left(\prod_{I_j \in X} p(I_j, T_q)\right) \\ &< \varepsilon \times |DS|.\end{aligned}$$

Let X' be the superset composed of itemset X and item I_s. Since $w(I_s) \leq w(X)$, thus:

$$\begin{aligned}\exp wSup(X') &= w(X') \times \sum_{X' \subseteq T_q \wedge T_q \in D} p(X', T_q) \\ &= \frac{w(I_s) + \sum_{I_j \in X} w(I_j)}{|k|+1} \times \sum_{X' \subseteq T_q \wedge T_q \in D} \left(\prod_{I_j \in X'} p(I_j, T_q)\right) \\ &\leq \frac{\sum_{I_j \in X} w(I_j)}{|k|} \times \sum_{X \subseteq T_q \wedge T_q \in D} \left(\prod_{I_j \in X} p(I_j, T_q)\right) \\ &< \varepsilon \times |D|.\end{aligned}$$

Therefore, X' is not a weighted frequent itemset.

Corollary 3: Let $I = \{I_1, I_2, \ldots, I_m\}$ be the items in uncertain dataset DS, and $w(I_j), j \in \{1, 2, \ldots, m\}$ is the weight of item I_j. Itemset X is the union of itemset X_1 and itemset X_2, $X_1 \neq \emptyset, X_2 \neq \emptyset$, and $X_1 \cap X_2 = \emptyset$. If itemset X is a weighted frequent itemset and itemset X_1 is not a weighted frequent itemset, then itemset X_2 must be a weighted frequent itemset.

Proof: Since itemset X is a weighted frequent itemset, then according to definition 6 and 7, it follows that:

$$\begin{aligned}\exp wSup(X) &= w(X) \times \sum_{X \subseteq T_q \wedge T_q \in DS} p(X, T_q) \\ &= \frac{\sum_{I_j \in X} w(I_j)}{|k|} \times \sum_{X \subseteq T_q \wedge T_q \in DS} \left(\prod_{I_j \in X} p(I_j, T_q)\right) \\ &\geq \varepsilon \times |DS|.\end{aligned}$$

Because $X_1 \subseteq X, X_2 \subseteq X$, thus:

$$\sum_{X_1 \subseteq T_q \wedge T_q \in DS} p(X_1, T_q) \geq \sum_{X \subseteq T_q \wedge T_q \in DS} p(X, T_q),$$

$$\sum_{X_2 \subseteq T_q \wedge T_q \in DS} p(X_2, T_q) \geq \sum_{X \subseteq T_q \wedge T_q \in DS} p(X, T_q).$$

Therefore, in order to prove itemset X_2 is a weighted frequent itemset, according to definition 6 and 7, we just need to prove $w(X_2) \geq w(X)$.

Since itemset X_1 is not a weighted frequent itemset, thus $w(X) > w(X_1)$, i.e.:

$$\frac{\sum_{I_j \in X} w(I_j)}{|k|} > \frac{\sum_{I_j \in X_1} w(I_j)}{|k_1|}, \tag{5}$$

where $|k_1|$ is the number of items in X_1, and $|k_2|$ is the number of items in X_2. Then the formula can be converted to:

$$|k_1| \times \left(\sum_{I_j \in X_1} w(I_j) + \sum_{I_j \in X_2} w(I_j)\right) > (|k_1| + |k_2|) \times \sum_{I_j \in X_1} w(I_j),$$

i.e.:

$$|k_1| \times \sum_{I_j \in X_2} w(I_j) > |k_2| \times \sum_{I_j \in X_1} w(I_j)$$

Thus we can get:

$$(|k_1| + |k_2|) \times \sum_{I_j \in X_2} w(I_j) > |k_2| \times \left(\sum_{I_j \in X_1} w(I_j) + \sum_{I_j \in X_2} w(I_j)\right),$$

i.e.:

$$\frac{\sum_{I_j \in X_2} w(I_j)}{|k_2|} > \frac{\sum_{I_j \in X} w(I_j)}{|k|}.$$

Therefore, it follows that $w(X_2) > w(X)$. Thus, itemset X_2 must be a weighted frequent itemset.

Theorem 4: (Existence Property of Weighted Frequent Subsets): Let $I = \{I_1, I_2, \ldots, I_m$ be the items in uncertain dataset DS, and $w\left(I_j\right), j \in \{1, 2, \ldots, m\}$ is the weight of item I_j. If itemset X is a weighted frequent k-itemset, then there are at least one subset of itemset X is weighted frequent $(k-1)$-itemset.

Proof: Theorem 4 will be proved using reduction to absurdity. The frequent k-itemset X can be denoted as $X = \{I_1, I_2, \ldots, I_k\}$. Suppose all $(k-1)$-subsets of itemset X are not weighted frequent itemsets. Suppose item $I_j, j \in \{1, 2, \ldots, k\}$ gets the minimum weight in X and $X = X' \cup \{I_j\}$, then it follows that $w\left(X'\right) > w\{I_j\}$, where X' is an $(k-1)$-subset of itemset X. According to Theorem 3, the superset X composed of itemset X' and 1-itemset $\{I_j\}$ cannot be a weighted frequent itemset. This contradict with the fact that itemset X is a weighted frequent k-itemset. Therefore, at least one subset of itemset X is weighted frequent $(k-1)$-itemset.

Theorem 3 is called weight judgment downward closure property, which can be used to narrow the searching space of weighted frequent itemsets. On the basis of weight judgment downward closure property, the efficient WD-FIM algorithm is proposed. Theorem 4 is called existence property of weighted frequent subsets. It guarantees that all weighted frequent itemsets can be discovered by the WD-FIM algorithm.

B. WD-FIM ALGORITHM

The proposed WD-FIM algorithm mines the weighted frequent itemsets from an uncertain database using the candidate generate-and-test paradigm. The weighted frequent itemsets are discovered by repeated iteration like U-Apriori algorithm. Obviously, there are significant differences between WD-FIM algorithm and U-Apriori algorithm. First, WD-FIM algorithm is proposed for mining weighted frequent itemsets in uncertain datasets. However, U-Apriori can only be used to discover frequent itemsets in uncertain datasets. Second, the basis of the proposed WD-FIM algorithm is the aforementioned weight judgment downward closure property and existence property of weighted frequent subsets, but the downward closure property is used directly to narrow the searching space of frequent itemsets in U-Apriori algorithm. Based on the aforementioned definitions and theorems, the pseudo code of proposed WD-FIM algorithm is given below.

As shown in Fig. 1, the proposed WD-FIM algorithm takes the input as: an uncertain transactional dataset, *DS*; a weight table, *wtable*; a user-specified minimum expected weighted support threshold, ε. First of all, the variables like *WFIS* and $WFIS_k$ are initialized (Line 1). Then it scans the dataset to get the weighted frequent 1-itemsets (Lines 2 to 8). Finally, on the basis of the proposed weight judgment downward closure property and the existence property of weighted frequent subsets, the weighted frequent k-itemsets will be discovered and all weighted frequent itemsets will be returned (Lines 9 to 24). During this process, the calculation of $CWFIS_k$ is extremely critical (Lines 13 to 15). Line 13 is performed to guarantee all the weighted frequent k-itemsets are included in $CWFIS_k$ according to the existence property of weighted frequent subsets. The *C*onnection operation is similar to the Apriori_gen operation in HEWI-Uapriori algorithm. Line 14 is implemented to obtain the itemsets which are definitely not weighted frequent

Algorithm 1: WD-FIM algorithm

Input:
DS, an uncertain transactional dataset;
wtable, a weight table;
ε, a user-specified minimum expected weighted support threshold.
Output:
The set of weighted frequent itemsets *WFIS*.
/* *initialization* */
1. initialize the variables and parameters
/* *scan the dataset and get weighted frequent 1-itemset* */
2. for each item I_j in *DS* do
3. scan *DS* and calculate $expwSup(I_j)$
4. if $expwSup(I_j) \geq |DS| \times \varepsilon$ then
5. $WFIS_1 = WFIS_1 \cup \{I_j\}$
6. end if
7. end for
8. $WFIS = WFIS \cup WFIS_1$
/* *scan the dataset and get weighted frequent k-itemsets* */
9. $CWFIS_1 = I$
10. let $SCWFIS_1$ be sorted $CWFIS_1$ by weight in descending order
11. set k =2
12. while $WFIS_{k-1} \neq null$ do
13. $CWFIS_k = Connection(WFIS_{k-1}, CWFIS_1)$
14. $NCWFIS_k = wConnection((CWFIS_{k-1} - WFIS_{k-1}), SCWFIS_1)$
15. $RCWFIS_k = CWFIS_k - NCWFIS_k$
16. for each candidate k itemset X in $RCWFIS_k$ do
17. scan *DS* and calculate $expwSup(X)$
18. if $expwSup(X) \geq |DS| \times \varepsilon$ then
19. $WFIS_k = WFIS_k \cup \{X\}$
20. end if
21. end for
22. $WFIS = WFIS \cup WFIS_k$
23. end while
24. return *WFIS*

FIGURE 1. Pseudo code of WD-FIM algorithm.

k-itemsets according to weight judgment downward closure property. The *wConnection* operation means that the itemsets in $(CWFIS_{k-1} - WFIS_{k-1})$ are connected with the itemsets with smaller weights in $SCWFIS_1$. Line 15 is performed to narrow the searching space of weighted frequent itemsets by deleting itemsets which are definitely not weighted frequent k-itemsets from $CWFIS_k$.

C. PERFORMANCE ANALYSIS

In this subsection, the performance of WD-FIM algorithm will be analyzed theoretically. The completeness of the weighted frequent itemsets generated by WD-FIM algorithm will be proved firstly as following.

Theorem 5 (Completeness): All weighted frequent itemsets can be discovered by WD-FIM algorithm.

Proof: According to Theorem 4, each weighted frequent k-itemset has at least one weighted frequent $(k-1)$-itemset. Additionally, each weighted frequent $(k-1)$-itemset is connected with $CWFIS_1$ to get $CWFIS_k$ in WD-FIM algorithm, as shown in Fig. 1 (Line 13). Consequently, as long as weighted frequent $(k-1)$-itemsets are complete, weighted frequent k-itemsets are also complete in WD-FIM algorithm. In WD-FIM algorithm, all the items in I are verified to get weighted frequent 1-itemsets, thus it is guaranteed that all the weighted frequent 1-itemsets are discovered, i.e. weighted

frequent 1-itemsets are complete. Then weighted frequent $(k-1)$-itemsets $(k = 2, 3, \ldots)$ are complete according to above analysis. Therefore, it follows that all weighted frequent itemsets can be discovered by WD-FIM algorithm.

Then the time efficiency of the proposed WD-FIM algorithm will be discussed and compared to HEWI-Uapriori algorithm which is the only candidate generate-and-test based weighted frequent itemset mining algorithm proposed for uncertain dataset to our knowledge. As we know, scanning the dataset to test and verify a candidate weighted frequent itemset is the most time-consuming. In this term, we can say that the time efficiency of a weighted frequent itemset mining algorithm mainly depends on the number of candidate weighted frequent itemsets. In WD-FIM algorithm, the number of candidate weighted frequent itemsets is $\sum_{1}^{k} N(CWFIS_k)$, where $N(CWFIS_k)$ is the number of candidate weighted frequent itemsets in $CWFIS_k$. In HEWI-Uapriori algorithm, the number of candidate weighted frequent itemsets is $\sum_{1}^{k} [N(C_k) + N(HUBEWI^k)]$, where $N(C_k)$ is the number of candidate weighted frequent itemset in C_k and $N(HUBEWI^k)$ is the number of high upper-bound expected weighted itemsets in $HUBEWI^k$. We can learn from above analysis that HEWI-Uapriori algorithm have to identify actual high expected weighted itemsets among high upper-bound expected weighted itemsets by performing an additional round of database scan. Thereby, we have reason to believe that WD-FIM algorithm is more time efficient than HEWI-Uapriori algorithm on condition that $N(CWFIS_k)$ is not greater than $N(C_k)$.

D. AN ILLUSTRATED EXAMPLE OF WD-FIM ALGORITHM

In this section, an example is given to illustrate the process of mining weighted frequent itemsets in WD-FIM algorithm. The uncertain dataset is shown in Table 1 and the weight of each item is given in Table 2. There are totally 6 items in the dataset which is composed of 10 transactions. The minimum expected weighted support threshold is set to 0.1. Thus the minimum expected weighted support is 1.0 which is the product of the minimum expected weighted support threshold and the number of transactions in the dataset.

In WD-FIM algorithm, the dataset is scanned for the first time to get weighted frequent 1-itemsets. The weighted frequent 1-itemsets are $\{B\}, \{C\}, \{D\}, \{E\}$. Then the weighted frequent 1-itemsets are connected with candidate weighted frequent 1-itemsets to generate candidate weighted frequent 2-itemsets according to existence property of weighted frequent subsets. The candidate weighted frequent 2-itemsets are $\{AB\}, \{BC\}, \{BD\}, \{BE\}, \{BF\}, \{AC\}, \{CD\}, \{CE\}, \{CF\}, \{AD\}, \{DE\}, \{DF\}, \{AE\}, \{EF\}$. Afterwards, the itemsets in $CWFIS_1$ which are not weighted frequent 1-itemsets are connected with itemsets with smaller weights in $SCWFIS_1$ to obtain the itemsets in $NCWFIS_2$ which are definitely not weighted frequent 2-itemsets according to weight judgment downward closure property. Finally, the $RCWFIS_2$ is finally obtained by deleting itemsets in $NCWFIS_2$ which are definitely not weighted frequent 2-itemsets. In the first iteration, $NCWFIS_2$ is an empty set, so $CWFIS_2$ cannot been effectively reduced.

This process is then repeated to obtain $WFIS_k, k \in \{2, 3, \ldots\}$ until no candidate itemsets can be generated. The complete progress of this example is as Figure 2 shows.

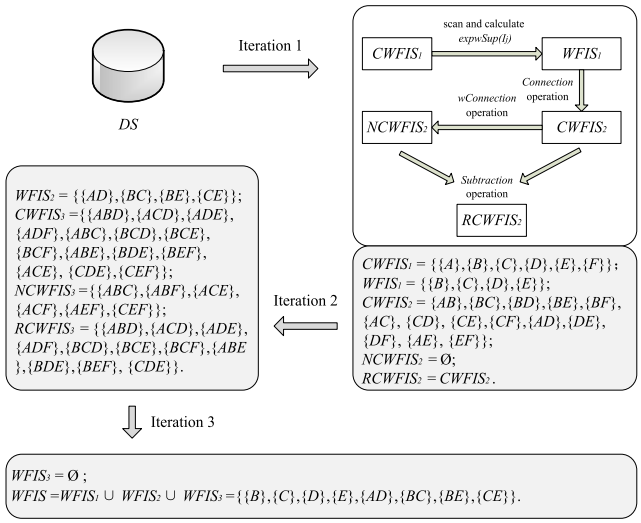

FIGURE 2. Schematic diagram of WD-FIM algorithm.

V. EXPERIMENTAL RESULTS

In this section, the performance of the proposed WD-FIM algorithm are verified and analyzed on both synthetic and real-life datasets as used in [28]. The existing Uapriori algorithm is the most famous FIM algorithm for mining expected support frequent itemsets in uncertain datasets. The existing HEWI-Uapriori is the only candidate generate-and-test based FIM algorithm for mining weighted frequent itemsets in uncertain datasets. Consequently, the existing Uapriori algorithm and HEWI-Uapriori algorithm are used as benchmark algorithms for comparison with the proposed WD-FIM algorithm.

The proposed WD-FIM algorithm and other compared algorithms are implemented in Python. Experiments are performed on a computer with an Intel Core i7-4510U 2.6GHz processor and 8GB RAM (Random Access Memory), running the 64 bit Microsoft Windows 7 operating system. The characteristics of the real-life and synthetic datasets used in the experiments are depicted in Table 4. There are two real-life datasets (mushroom and foodmart) and one synthetic (T10I4D100K) dataset. $|DS|$ is the total number of transactions in a dataset. $|I|$ is the number of distinct items in a dataset. $AvgLen$ is the average number of items in a transaction. In addition, the weights of items in a dataset and the existential probabilities of items in transactions are generated randomly in the (0,1] interval.

In order to test the effectiveness and efficiency of the proposed algorithm, three groups of experiments are carried

TABLE 4. Dataset characteristics.

Dataset	$\|DS\|$	$\|I\|$	$AvgLen$
mushroom	8124	119	23
retail	88162	16470	10.3
T10I4D100K	100000	870	10.1

out to show the performance of the proposed WD-FIM algorithm in terms of runtime, number of patterns and memory consumption.

A. PERFORMANCE OF RUNTIME

In this subsection, the runtime of the proposed WD-FIM algorithm and the compared algorithms are first analyzed. In this group of experiments, the sizes of the datasets are fixed. However, the minimum expected weighted support threshold is changed to analyze the corresponding changes of runtime. Note that the Uapriori algorithm can be seen as a FIM algorithm for weighted uncertain datasets in which the weights of all items are set to 1. Moreover, the runtime contains both the computation time and the time of scanning the dataset.

It can be observed from Fig. 3, Fig. 4 and Fig. 5 that the runtimes of Uapriori algorithm, HEWI-Uapriori algorithm and WD-FIM algorithm all decline gradually with the increase of minimum expected weighted support threshold. The reason is that the number of candidate weighted frequent itemsets to be verified has dropped with the increase of minimum expected weighted support threshold. Thus all the three algorithms do not have to spend a lot of time scanning the dataset any more.

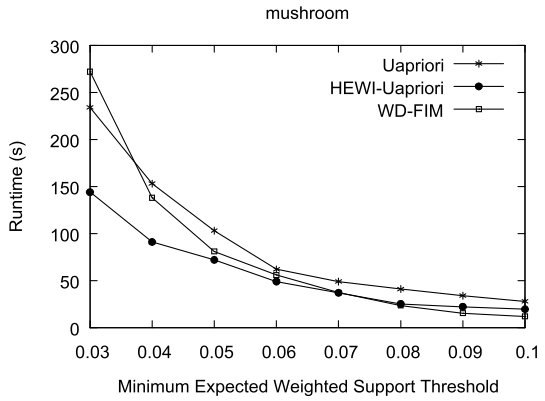

FIGURE 3. Runtime analysis for the mushroom dataset.

From Fig. 3, Fig. 4 and Fig. 5, it also can be seen that the proposed WD-FIM algorithm is faster than Uapriori algorithm and HEWI-Uapriori algorithm when the minimum expected weighted support threshold is relatively large. This is because the number of candidates can be effectively reduced by the proposed weight judgment downward closure property. However, when the minimum expected weighted support threshold becomes smaller, there will be more candidates in $CWFIS_k$. Thus, the *wConnection* operation and the process of deleting itemsets which are

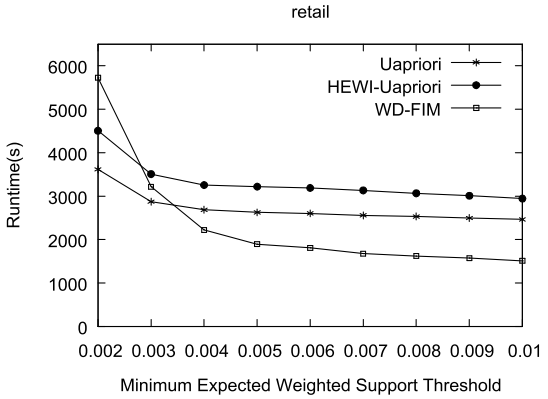

FIGURE 4. Runtime analysis for the retail dataset.

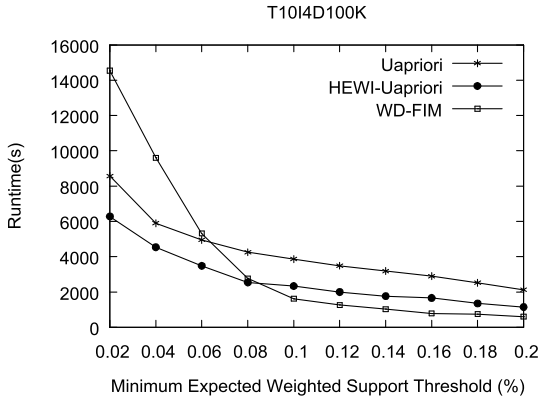

FIGURE 5. Runtime analysis for the T10I4D100K dataset.

definitely not weighted frequent k-itemsets from $CWFIS_k$ can be very time-consuming. Consequently, the runtime of the proposed WD-FIM algorithm increases sharply. For example, Fig. 4 shows that the runtime of the proposed WD-FIM algorithm is much more shorter when the minimum expected weighted support threshold is varied from 0.004 to 0.01. However, when the minimum expected weighted support threshold is set to 0.002, the proposed WD-FIM algorithm runs much slower than Uapriori algorithm and HEWI-Uapriori algorithm.

As for the performance of Uapriori algorithm and HEWI-Uapriori algorithm, it can be seen form Fig. 3 and Fig. 5 that Uapriori algorithm is slower than HEWI-Uapriori algorithm for the mushroom dataset and the T10I4D100K dataset. This is because for Uapriori algorithm all the weights of items in mushroom can be seen as 1. Consequently, when the minimum expected weighted support threshold is fixed, the Uapriori algorithm, which uses the downward closure property to prune unpromising candidates, is more likely to have much more candidates than the HEWI-Uapriori algorithm. However, Fig. 4 shows that the Uapriori algorithm is more faster than HEWI-Uapriori algorithm for the retail dataset. It is also reasonable because there are too many items in the retail dataset. The downward closure property in the Uapriori algorithm plays a more effective role in narrowing the searching space of weighted frequent itemsets for the retail dataset.

B. PATTERNS ANALYSIS

In this subsection, the number of patterns discovered by Uapriori algorithm, HEWI-Uapriori algorithm and the proposed WD-FIM algorithm is evaluated under various minimum expected weighted support threshold values first. It should be noted that the patterns discovered by Uapriori algorithm are expected support frequent itemsets (EFIs), the patterns discovered by HEWI-Uapriori algorithm are high expected weighted itemsets (HEWIs), and the patterns discovered by the proposed WD-FIM algorithm are weighted frequent itemsets (WFIs). The results for the mushroom dataset, the retail dataset and the T10I4D100K dataset are as shown in Fig. 6, Fig. 7 and Fig. 8 respectively.

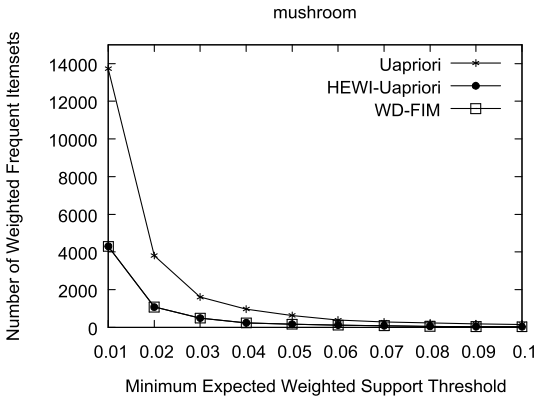

FIGURE 6. Patterns analysis for the mushroom dataset.

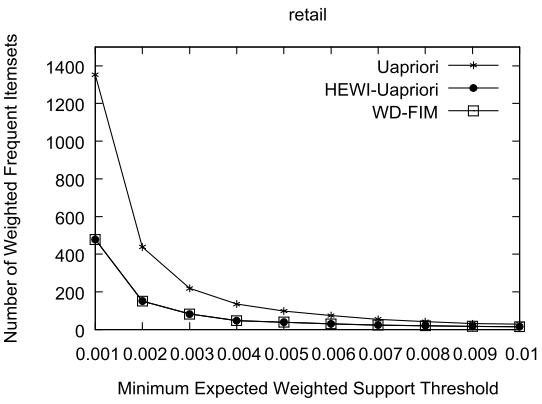

FIGURE 7. Patterns analysis for the retail dataset.

From Fig. 6, Fig. 7 and Fig. 8, it can be seen that the number of patterns discovered by all the three algorithms decline gradually with the increase of minimum expected weighted support threshold. The downtrend is significant at the beginning, but gradually level off in late-stage. When the minimum expected weighted support threshold is fixed, it is obvious that the Uapriori algorithm could always discover more patterns compared with the HEWI-Uapriori algorithm and the proposed WD-FIM algorithm. The reason is that the weight of all items in Uapriori algorithm can be seen as 1 as mentioned above. Thus, an itemset is more likely to be an EFI when the minimum expected weighted support

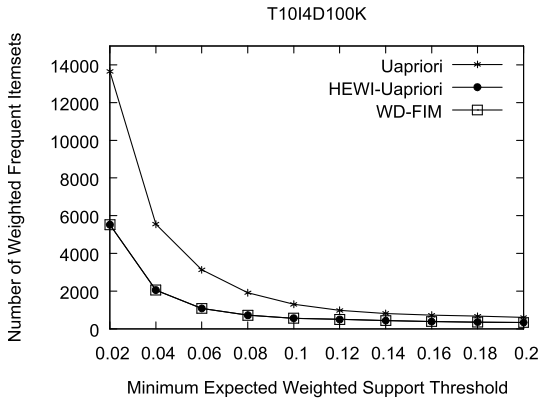

FIGURE 8. Patterns analysis for the T10I4D100K dataset.

threshold is fixed. Moreover, an important fact is that the number of patterns discovered by the HEWI-Uapriori algorithm and the proposed WD-FIM algorithm is identical. This is because both the HEWI-Uapriori algorithm and the proposed WD-FIM algorithm are exact approaches which can discover all possible weighted frequent itemsets in the dataset.

In the following group of experiments, the number of k-itemsets discovered by Uapriori algorithm, HEWI-Uapriori algorithm and the proposed WD-FIM algorithm is also analyzed. Results for different datasets are shown in Fig. 9, Fig. 10 and Fig. 11 respectively.

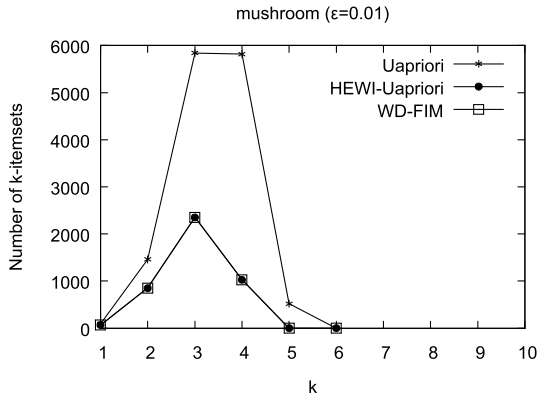

FIGURE 9. Number of k-itemsets analysis for the mushroom dataset.

From Fig. 9, Fig. 10 and Fig. 11, it can be seen that the proposed WD-FIM algorithm can generate the same amount of k-itemsets as the HEWI-Uapriori algorithm on all the three datasets. However, the number of k-itemsets generated by the HEWI-Uapriori algorithm and the proposed WD-FIM algorithm is smaller than the number of k-itemsets discovered by the Uapriori algorithm. This is because both the weight and probability properties are considered in the HEWI-Uapriori algorithm and the proposed WD-FIM algorithm. Thus, fewer and more meaningful k-itemsets are produced compared with the Uapriori algorithm. Additionally, for the same dataset, the k-itemsets produced by the compared three algorithms have the same distribution trend.

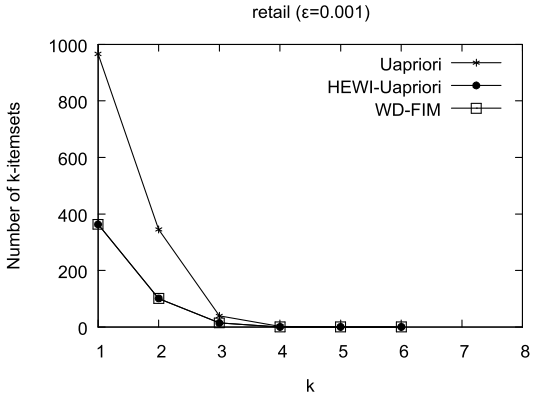

FIGURE 10. Number of k-itemsets analysis for the retail dataset.

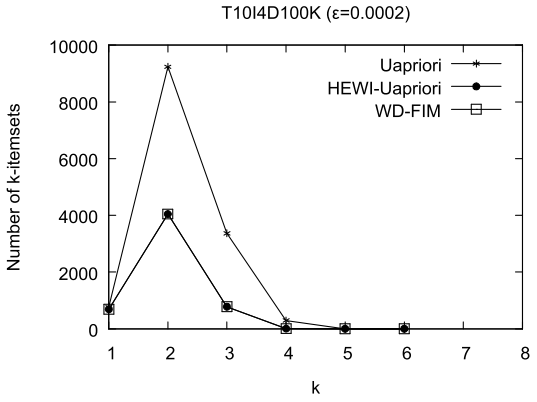

FIGURE 11. Number of k-itemsets analysis for the T10I4D100K dataset.

C. PERFORMANCE OF MEMORY CONSUMPTION

In this subsection, additional experiments were performed to assess the memory consumption of Uapriori algorithm, HEWI-Uapriori algorithm and the proposed WD-FIM algorithm. Memory consumptions for different datasets are shown in Fig. 12, Fig. 13 and Fig. 14 respectively.

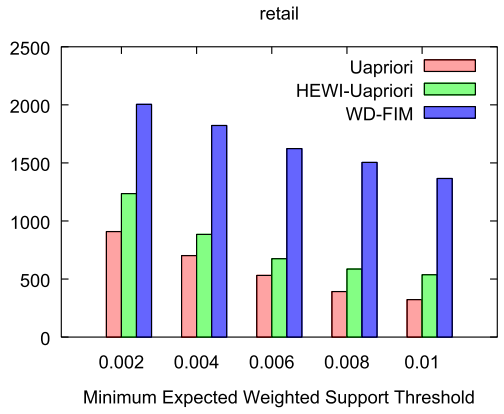

FIGURE 12. Memory consumption analysis for the retail dataset.

From Fig. 12, Fig. 13 and Fig. 14, it can be seen that the Uapriori algorithm always requires less memory than the HEWI-Uapriori algorithm and the proposed WD-FIM algorithm. There are several reasons for this. First, the

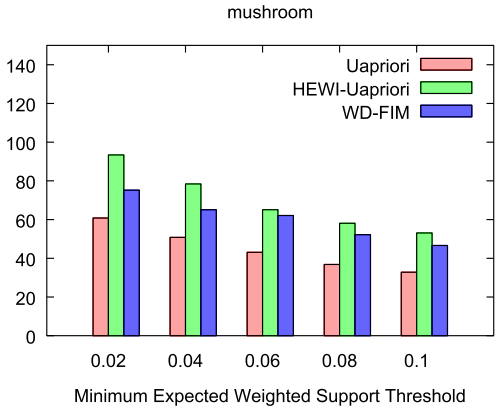

FIGURE 13. Memory consumption analysis for the mushroom dataset.

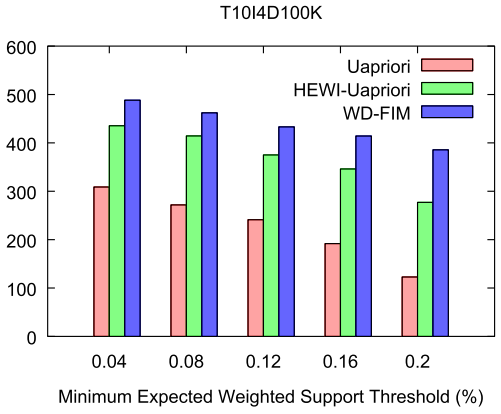

FIGURE 14. Memory consumption analysis for the T10I4D100K dataset.

Uapriori algorithm uses the downward closure property to prune numerous unpromising candidates of EFIs directly. Thus, it generates fewer candidates. Second, the HEWI-Uapriori algorithm applies the proposed HUBEW downward closure property to prune the candidates. After the first dataset scan, the candidates are still kept in the main memory to be further determined. In the proposed WD-FIM algorithm, the weight judgment downward closure property is used to narrow the searching space of weighted frequent itemsets. In this process, the $NCWFIS_k$ should be calculated and kept in the main memory too.

VI. CONCLUSIONS

In order to realize intelligent decision making in smart systems, a weight judgment downward closure property based frequent itemset mining algorithm is proposed in this paper to narrow the searching space of weighted frequent itemsets and improve the time efficiency.

The weight judgment downward closure property for weighted frequent itemsets and the existence property of weighted frequent subsets are introduced and proved first. Based on these two properties, the WD-FIM algorithm is described in detail. Moreover, the completeness and time efficiency of WD-FIM algorithm are analyzed theoretically.

Finally, the performance of the proposed WD-FIM algorithm is verified on both synthetic and real-life datasets.

REFERENCES

[1] R. Ishita and A. Rathod, "Frequent itemset mining in data mining: A survey," *Int. J. Comput. Appl.*, vol. 139, no. 9, pp. 15–18, Apr. 2016.
[2] L. Yue, "Review of algorithm for mining frequent patterns from uncertain data," *Int. J. Comput. Sci. Netw. Secur.*, vol. 15, no. 6, pp. 17–21, Jun. 2015.
[3] T. G. Green and V. Tannen, "Models for incomplete and probabilistic information," in *Current Trends in Database Technology* (Lecture Notes in Computer Science). Berlin, Germany: Springer, Oct. 2006, pp. 278–296.
[4] C. C. Aggarwal and P. S. Yu, "A survey of uncertain data algorithms and applications," *IEEE Trans. Knowl. Data Eng.*, vol. 21, no. 5, pp. 609–623, May 2009.
[5] D. Suciu, "Probabilistic databases," *ACM Sigact News*, vol. 39, no. 2, pp. 111–124, Jun. 2008.
[6] C.-K. Chui, B. Kao, and E. Hung, "Mining frequent itemsets from uncertain data," in *Proc. Pacific-Asia Conf. Knowl. Discovery Data Mining*, 2007, pp. 47–58.
[7] C. K. Chui and B. Kao, "A decremental approach for mining frequent itemsets from uncertain data," in *Proc. PAKDD*, 2008, pp. 64–75.
[8] L. Wang, D. W.-L. Cheung, R. Cheng, S. D. Lee, and X. S. Yang, "Efficient mining of frequent item sets on large uncertain databases," *IEEE Trans. Knowl. Data Eng.*, vol. 24, no. 12, pp. 2170–2183, Dec. 2012.
[9] X. Sun, L. Lim, and S. Wang, "An approximation algorithm of mining frequent itemsets from uncertain dataset," *Int. J. Advancements Comput. Technol.*, vol. 4, no. 3, pp. 42–49, Feb. 2012.
[10] J. Pei, J. Han, and W. Wang, "Constraint-based sequential pattern mining: The pattern-growth methods," *J. Intell. Inf. Syst.*, vol. 28, no. 2, pp. 133–160, Apr. 2007.
[11] C. C. Aggarwal, Y. Li, J. Wang, and J. Wang, "Frequent pattern mining with uncertain data," in *Proc. ACM KDD*, 2009, pp. 29–38.
[12] C. K.-S. Leung, M. A. F. Mateo, and D. A. Brajczuk, "A tree-based approach for frequent pattern mining from uncertain data," in *Proc. PAKDD*, 2008, pp. 653–661.
[13] J. Han, J. Pei, and Y. Yin, "Mining frequent patterns without candidate generation," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 2000, pp. 1–12.
[14] C. K.-S. Leung and S. K. Tanbeer, "Fast tree-based mining of frequent itemsets from uncertain data," in *Proc. Int. Conf. Database Syst. Adv. Appl.*, 2012, pp. 272–287.
[15] C. K.-S. Leung and S. K. Tanbeer, "PUF-Tree: A compact tree structure for frequent pattern mining of uncertain data," in *Proc. Pacific-Asia Conf. Knowl. Discovery Data Mining*, 2013, pp. 13–25.
[16] C. K.-S. Leung, R. K. Mackinnon, and S. K. Tanbeer, "Tightening upper bounds to the expected support for uncertain frequent pattern mining," in *Proc. 18th Int. Conf. Knowl.-Based Intell. Inf. Eng. Syst.*, 2014, pp. 328–337.
[17] C. W. Lin and T. P. Hong, "A new mining approach for uncertain databases using CUFP trees," *Expert Syst. Appl.*, vol. 39, no. 4, pp. 4084–4093, Mar. 2012.
[18] L. Wang, L. Feng, and M. Wu, "AT-Mine: An efficient algorithm of frequent itemset mining on uncertain dataset," *J. Comput.*, vol. 8, no. 6, pp. 1417–1427, Jun. 2013.
[19] G. Lee, U. Yun, and H. Ryang, "An uncertainty-based approach: Frequent itemset mining from uncertain data with different item importance," *Knowl.-Based Syst.*, vol. 90, pp. 239–256, Dec. 2015.
[20] G. Lee and U. Yun, "A new efficient approach for mining uncertain frequent patterns using minimum data structure without false positives," *Future Gener. Comput. Syst.*, vol. 68, pp. 89–110, Mar. 2017.
[21] W. Wang, J. Yang, and P. S. Yu, "Efficient mining of weighted association rules (war)," in *Proc. 6th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2000, pp. 270–274.
[22] F. Tao, F. Murtagh, and M. Farid, "Weighted association rule mining using weighted support and significance framework," in *Proc. 9th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2003, pp. 661–666.
[23] U. Yun and J. Leggett, "WFIM: Weighted frequent itemset mining with a weight range and a minimum weight," in *Proc. SIAM Int. Conf. Data Mining*, 2005, pp. 636–640.
[24] U. Yun and J. Leggett, "WSpan: Weighted sequential pattern mining in large sequence databases," in *Proc. IEEE Int. Conf. Intell. Syst.*, Sep. 2006, pp. 512–517.
[25] G. Lee, U. Yun, and K. H. Ryu, "Sliding window based weighted maximal frequent pattern mining over data streams," *Expert Syst. Appl.*, vol. 41, no. 2, pp. 694–708, Feb. 2014.
[26] U. Yun, G. Lee, and K. H. Ryu, "Mining maximal frequent patterns by considering weight conditions over data streams," *Knowl.-Based Syst.*, vol. 55, no. 55, pp. 49–65, Jan. 2014.
[27] G. Lee, U. Yun, and H. Ryang, "Mining weighted erasable patterns by using underestimated constraint-based pruning technique," *J. Intell. Fuzzy Syst.*, vol. 28, no. 3, pp. 1145–1157, May 2015.
[28] A. C.-W. Lin, W. Gan, P. Fournier-Viger, T.-P. Hong, and V. S. Tseng, "Weighted frequent itemset mining over uncertain databases," *Appl. Intell.*, vol. 44, no. 1, pp. 232–250, Jan. 2016.

XUEJIAN ZHAO received the M.Sc. and Ph.D. degrees in computer application technology from the Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2007 and 2011, respectively. He is currently an Associate Professor with the School of Modern Posts, Nanjing University of Posts and Telecommunications, Nanjing. His current research interests include wireless sensor networks, ad hoc networks, and big data.

XINHUI ZHANG is currently pursuing the master's degree with the College of Internet of Things, Nanjing University of Posts and Telecommunications, Nanjing, China. Her current research interests include wireless sensor networks.

PAN WANG received the Ph.D. degree from the College of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing, China, in 2013. He is currently an Associate Professor with the School of Modern Posts, Nanjing University of Posts and Telecommunications. His current research interests include information security, ubiquitous computing, and wireless communications technologies.

SONGLE CHEN received the Ph.D. degree from the Department of Computer Science and Technology, Nanjing University, Nanjing, China, in 2015. He is currently a Lecturer with the Nanjing University of Posts and Telecommunications, Nanjing. His current research interests include grid computing, social Network, computer vision, geometric modeling, and realistic rendering.

ZHIXIN SUN received the Ph.D. degree from the Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 1998. He held a post-doctoral position at Seoul National University in 2002. He is currently a Professor with the School of Modern Posts, Nanjing University of Posts and Telecommunications. His current research interests include computer network and security, network multimedia communication, and network management and protocol.