

Vertical Mining of Frequent Patterns from Uncertain Data

Laila A. Abd-Elmegid

Faculty of Computers and Information, Helwan University

E-mail: eng.lole@yahoo.com

Mohamed E. El-Sharkawi

Faculty of Computers and Information, Cairo University

E-mail: mel_sharkawi@gmail.com

Laila M. El-Fangary & Yehia K. Helmy

Faculty of Computers and Information, Helwan University

E-mail: {lailaelfangery@gmail.com, yhelemy@yahoo.com}

Abstract

Efficient algorithms have been developed for mining frequent patterns in traditional data where the content of each transaction is definitely known. There are many applications that deal with real data sets where the contents of the transactions are uncertain. Limited research work has been dedicated for mining frequent patterns from uncertain data. This is done by extending the state of art horizontal algorithms proposed for mining precise data to be suitable with the uncertainty environment. Vertical mining is a promising approach that is experimentally proved to be more efficient than the horizontal mining. In this paper we extend the state-of-art vertical mining algorithm Eclat for mining frequent patterns from uncertain data producing the proposed UEclat algorithm. In addition, we compared the proposed UEclat algorithm with the UF-growth algorithm. Our experimental results show that the proposed algorithm outperforms the UF-growth algorithm by at least one order of magnitude.

Keywords: Frequent patterns, Uncertain data, Vertical mining, Tidset, Diffset, Association rules, Data mining

1. Introduction

Frequent pattern mining has been a focused theme in data mining research for over a decade. It is a core technique used in many mining tasks like sequential pattern mining, structured pattern mining, correlation mining, associative classification, and frequent pattern-based clustering (C. Zhu, X. Zhang, J. Sun, and B. Huang, 2009), as well as their broad applications (H. Kriegel, P. Kroger and A. Zimek, 2009) (Y. Koh, N. Rountree, R. O'Keefe, 2008) (A. Ceglar and J. Roddick, 2006). So, a great effort has been dedicated to this research and tremendous progress has been made to develop efficient and scalable algorithms for frequent pattern mining (A. Ceglar and J. Roddick, 2006) (Z. Zheng, R. Kohavi, and L. Mason, 2001) (J. Han, H. Cheng, D. Xin, and X. Yan, 2007). All these algorithms deal with precise data sets (J. Han, J. Pei, Y. Yin and R. Mao, 2004) (M. Zaki, 2000) (P. Shenoy, J. Haritsa, S. Sudarshan, G. Bhalotia, M. Bawa and D. Shah, 2000) (M. Zaki and K. Gouda, 2003) (W. Consue, and W. Kurutach, 2003) (B. Goethals, 2004) (M. Song, S. Rajasekaran, 2006). Such data is characterized by known and definite existence of the items or events in the transactions. However, there are datasets where the exact existence of items in the transactions cannot be gained. These datasets are called uncertain data. The existence of an item in a transaction is best captured by a likelihood measure or a probability (Chui, C.-K., Kao, B., Hung, E.). As an example, a medical dataset may contain a table of patients' records, each of which contains a set of diseases that a patient suffers. In such case the physician may highly suspect (but cannot guarantee) that a patient suffers from a specific disease. So he expresses his suspicion by a probability of the existence of such disease (H., Li, H., Yang, Q. 2007). Another example of uncertain dataset is pattern recognition applications. Given a satellite picture, image processing techniques can be applied to extract features that indicate the presence or absence of certain target objects (such as bunkers). Due to noises and limited resolution, the presence of a feature in a spatial area is often uncertain and expressed as a probability (Dai, X., Yiu, M.L., et al. 2005). Figure 1 shows an example of precise and uncertain data sets. Few algorithms have been dedicated for mining frequent patterns from uncertain data. All these algorithms follow the horizontal data

representation. Although vertical data representation is a promising approach no published research work has been studied this issue.

In this paper we study the problem of mining frequent patterns from uncertain data using the vertical data representation Tidset. We extend the state-of-art vertical mining algorithm Eclat to be suitable with the uncertain environment. During such extension we propose the Utidset structure for vertical representation of uncertain data. A comparative study between the proposed UEclat algorithm and the well known UF-growth algorithm is conducted and showed that the proposed algorithm outperforms the UF-growth by at least one order of magnitude.

The rest of the paper is organized as follows: In Section two we introduce the preliminaries of mining frequent itemsets. Whereas, in Section three we list and discuss the related work. Section four explains in details the proposed UEclat algorithm. A performance study is given in Section five. Finally, a conclusion is given in Section six.

2. Background

The problem of mining frequent itemsets can be formulated as follows. Let I be a set of items and T a database of transactions, where each transaction has a unique transaction identifier (Tid) and contains a set of items. A set $X \subseteq I$ is called an *itemset*, and a set $Y \subseteq T$ is called a *tidset*. An itemset that contains k items is called a *k-itemset*. The *support* of an itemset X , denoted $\sigma(X)$, is the number of transactions in which X occurs. An itemset is *frequent* if its support is greater than or equal to a user-specified *minimum support* (min_sup) value (M. Zaki and K. Gouda, 2003). Figure 2 shows the frequent itemsets for different values of min_sup on a given transactional database.

A key difference between precise and uncertain data is that each transaction of the latter contains items and their *existential probabilities*. The existential probability $P(x, ti)$ of an item x in a transaction ti indicates the likelihood of x being present in ti . Using the “*possible world*” interpretation of uncertain data (Leung, C.K.-S., Carmichael, C.L., Hao, B. 2007)(C. Aggarwal, 2009), there are two possible worlds for an item x and a transaction ti : (i) $W1$ where $x \in ti$ and (ii) $W2$ where $x \notin ti$. Although it is uncertain which of these two worlds be the true world, the probability of $W1$ be the true world is $P(x, ti)$ and that of $W2$ is $1 - P(x, ti)$. Figure 3 shows all possible worlds for a data set contains only two transactions and two items. To a further extent, there are many items in each of many transactions in a transaction database TDB . Hence, the *expected support* of a pattern (or a set of items) X in TDB can be computed by summing the support of X in possible world Wj (while taking in account the probability of Wj to be the true world) over all possible worlds (Leung, C.K.-S., Carmichael, C.L., Hao, B. 2007).

The following formula in rule 1 is used to calculate the expected support of any itemset X . a summarized form of rule 1 exists in rule 2. With this setting, a pattern X is considered *frequent* if its expected support equals or exceeds the user-specified support threshold min_sup .

$$\begin{aligned}
 expSup(X) &= \sum_j \left[sup(X) \text{ in } W_j \times \prod_{i=1}^{|TDB|} \left(\prod_{x \in t_i \text{ in } W_j} P(x, t_i) \times \prod_{y \notin t_i \text{ in } W_j} (1 - P(y, t_i)) \right) \right] \quad (1) \\
 &= \sum_{i=1}^{|TDB|} \left(\prod_{x \in X} P(x, t_i) \right). \quad (2)
 \end{aligned}$$

There are two types of data representation; the horizontal and vertical representation as in Figure 4. In the horizontal representation approach, the data is organized as a set of rows. Each row has a key identifier that is the transaction identifier (TID) and a set of IIDs (Item Identifier). While in the vertical representation approach, the data is organized as a set of columns; each column has a key identifier, which is the item identifier (IID) and a set of TIDs (M. Zaki and K. Gouda, 2003). There are many variations of vertical and horizontal representations presented in (P. Shenoy, J. Haritsa, S. Sudarshan, G. Bhalotia, M. Bawa and D. Shah, 2000).

Most of the previous work on mining frequent patterns is based on the horizontal representation. However, recently a number of vertical mining algorithms have been proposed for mining frequent itemsets. Mining algorithms using the vertical representation have shown to be effective and usually outperform horizontal approaches (M. Song, S. Rajasekaran, 2006). This advantage stems from the fact that frequent patterns can be counted via tidset intersections, instead of using complex internal data structures like the hash/search trees that the horizontal algorithms require (M. Zaki and K. Gouda, 2003).

Also in the vertical mining, the candidate generation and counting phases are done in a single step. This is done because vertical mining offers natural pruning of irrelevant transactions as a result of an intersection. Another

feature of vertical mining is the utilization of the independence of classes, where each frequent item is a class that contains a set of frequent k -itemsets (where $k > 1$) (M. Zaki, 2000).

3. Related Work

Limited research work has been dedicated for mining frequent patterns from uncertain data. Several studies show that broad classes of algorithms can be extended to the uncertain data setting. To the best of our knowledge no research work has been done to study the feasibility of extending vertical mining algorithms for mining uncertain data. The following paragraphs describe the horizontal algorithms proposed for mining frequent patterns from uncertain data.

Chui et al. proposed the U-Apriori algorithm, which is a modification of the Apriori algorithm. Specifically, instead of incrementing the support counts of candidate patterns by their *actual* support, U-Apriori increments the support counts of candidate patterns by their *expected* support (using Equation (2)). However, U-Apriori suffers from the following problems: (i) Inherited from the Apriori algorithm, U-Apriori does not scale well when handling large amounts because it also follows a level-wise generate-and-test framework. (ii) If the existential probabilities of most items within a pattern X are small, increments for each transaction can be insignificantly small. Consequently, many candidates would not be recognized as infrequent until most (if not all) transactions were processed.

Leung et al. proposed a UF-tree which is a variant of the FP-tree. Each node in the UF-tree stores (i) an item, (ii) its expected support, and (iii) the number of occurrence of such expected support for such an item. The proposed UF-growth algorithm constructs the UF-tree as follows. It scans the database once and accumulates the expected support of each item. Hence, it finds all frequent items (i.e. items having expected support $\geq \text{minsup}$). It sorts these frequent items in descending order of accumulated expected support. The algorithm then scans the database the second time and inserts each transaction into the UF-tree in a similar fashion as in the construction of an FP-tree except that the new transaction is merged with a child (or descendant) node of the root of the UF-tree (at the highest support level) only if the same item and the same expected support exist in both the transaction and the child (or descendant) nodes.

Recently, Aggarwal (C. Aggarwal, 2009) extended several existing classical frequent itemset mining algorithms for deterministic datasets, and compared their relative performance in terms of efficiency and memory usage. The study focused on candidate generate-and-test algorithms, hyper-structure algorithms and pattern growth based algorithms. According to the experiments in the study, the hyper-structure and the candidate generate-and-test algorithms are proved to perform much better than tree-based algorithms.

4. Vertical Mining of Frequent Patterns from Uncertain Data

In this section we propose the UEclat algorithm for vertical mining of frequent patterns from uncertain data. First we introduce the proposed Utidset structure that is used in the mining process. Second, we explain in details the UEclat algorithm provided by an illustrative example.

4.1 Construction of the Utidset mining structure

According to the special nature of uncertain data, a key challenge in its mining is how to represent and store this data. In tidset vertical representation of precise data, each item is associated with a set of transactions identifiers (Tids) where this item appears. The case is different in uncertain data as the item's appearance in the transaction is represented by an existential probability ranging from a positive value close to 0 (indicating that the item has insignificantly low chance to be present in the transaction) to value of 1 (indicating that the item is definitely present).

To effectively represent uncertain data in vertical representation, we propose the Utidset structure which is a variant of the tidset. In the Utidset structure, each node stores (i) an item, (ii) its existential probability in every transaction. It scans the database once and accumulates the expected support of each item. Hence, it finds all frequent items. The following example illustrates the construction process of the Utidset structure.

Example 1. Consider the following uncertain transactional data set in Table 1 and construct the Utidset for all items respecting to minimum support equals 1.

Here, each transaction contains items and their corresponding existential probability. For example the existential probability of item A is 0.9 in all transactions (T1, T2, and T5). However, the case is different for item E, for example, where there are different existential probabilities in different transactions.

The Utidset can be constructed as follows. First, the UEclat algorithm scans the data only once and accumulates the expected support of each item. The expected support is calculating by summing the probabilities of the

current processed item in all its transactions. Table 2 shows all the items with their corresponding Utidset and expected support.

A pruning step is done for removing all items with expected support less than the minimum expected support. At such step, both items E and F are removed. Table 3 shows the Utidset vertical representation of frequent items.

4.2 Calculating the support of k-itemsets where (k>1)

After representing all the frequent items using the Utidset structure, we need to move for mining other frequent k-itemsets where k>1. The main issue here is how to calculate the support of k-itemsets. In the main Eclat algorithm that is used for mining precise data, the support of any given k-itemset is calculated simply by counting the number of transactions result from intersecting the subsets of the k-itemset. So, for any two subsets Y, Z ⊆ X, such that Y ∩ Z = X the support of X is calculated by intersecting both tidsets of Y and Z. However, the case is different in uncertain data because the item Y may have high existential probability in a specific transaction and at the same transaction item Z may have low existential probability which will affect the real probability of item X. So, in vertical mining of uncertain data it is not enough to count the common transactions between any two subsets to calculate the support of their superset, we also need to consider the existential probabilities of both subsets in each common transaction. According to rule 2 described in Section two, the expected support of any k-itemset is the multiplication of the existential probabilities of its subsets in all transactions. For simplification we can conclude that: For any k-itemset X with subset itemsets Y and Z

$$\text{Exp_sup}(X) = \sum_{i=1}^n ((p(Y,t_i) * p(Z,t_i)))$$

For example the expected support of itemset BC is calculated by ((0.6*0.8) + (0.9*0.7)). The first bracket (0.6*0.8) is the support of BC in transaction T1. Whereas the second bracket (0.9*0.7) is the support of BC in transaction T3. Only transactions T1 and T3 are considered because they are the common transactions between the two subsets B and C. The total expected support of itemset BC is calculated by summing all the transactional supports that will result in value of 1.11.

4.3 Mining frequent k-itemsets

Once the Utidsets of all frequent items are constructed, the proposed UEclat algorithm recursively mines frequent itemsets from this Utidset structure. At the first step each frequent item is added to the output set. After that, for every such frequent itemset i, the i-projected database Di is created. This is done by first finding every item j that frequently occurs together with i. the support of this set {i, j} is computed using the previous rule. If {i, j} is frequent, then j is inserted into Di. The algorithm is called recursively to find all frequent itemsets in the new database Di. Figure 4 shows the pseudo code of the UEclat algorithm.

The following example illustrates how the UEclat algorithm mines all frequent k-itemsets from the Utidset structure.

Example 2. Once the Utidset structure is constructed as in Table 2, the proposed UEclat algorithm recursively mines frequent itemsets from the structure with minimum expected support equals to 1 as follows. At the beginning, the UEclat algorithm starts to mine 2-itemsets. For each 2-itemset the expected support is calculated according to the proposed rule. A pruning process is done for all itemsets with expected support less than 1. In this example, the itemsets AB and BD are removed. Table 4 shows result of this step. For clarification purpose, we associate with each transaction the values of its two existential probabilities. Note that the columns of the infrequent itemsets AB and BD are highlighted by a grey color.

Based on the frequent 2-itemsets in Table 4, the same recursive process is done for mining frequent 3-itemsets. Table 5 shows the Utidset of frequent 3-itemsets. Here there is only frequent 3-itemset which is ACD with expected support 1.13 and as a result there is no further processing.

5. Performance Study

In this section we measure the performance of the proposed UEclat algorithm and also compare its performance of with the most recent algorithm UF-growth. Datasets used in the experiments are downloaded from <http://kdd09.crowdvine.com/talks/show/4894>. Two data sets are used in the experiments, T40I10D100K and T25I15D320k, are generated using the IBM synthetic data set generator. These data sets contain 100k records with an average transaction length of 10 items and a domain of 1,000 items. These data sets are used in the performance study of the UF-growth algorithm. All programs are implemented and compiled with Microsoft Visual C# Net 2005. All experiments are performed on an Intel processor 2GHz Core 2 Due with 2G of memory,

running Windows Vista. The accumulated time is measured from the beginning of reading the data set and converting it to its structure to the end of the frequent pattern mining process.

In Figure 6 and Figure 7, a comparison between the UEclat algorithm and the UF-growth algorithm is conducted for the T40I10D100K data set for varying minimal support thresholds. Whereas the performance regarding the T25I15D320k data set is shown in Figure 8 and Figure 9. All experimental results confirmed that, when *minsup* increased, fewer patterns had expected support $\geq \text{min_sup}$, and thus shorter runtimes were required. One can see that the execution time of UEclat algorithm is better than UF-growth in general. However, when the minimum support is lower, the performance of our method gets better than UF-growth. The utilization of the simple data representation used in the UTidset structure and fast counting mechanism accelerate the process of mining large number of frequent patterns and thus result in less processing time rather than the one required by the UF-growth algorithm. These experiments show that vertical mining of uncertain data is a promising approach that can achieve efficient performance regarding its features as been proved in traditional precise data.

6. Conclusion

Most existing algorithms mine frequent patterns from traditional transaction databases that contain precise data. In these databases, users definitely know whether an item (or an event) is present in, or is absent from, a transaction in the databases. However, there are many real-life situations in which one needs to deal with uncertain data. In such data users are uncertain about the presence or absence of some items or events. For example, a physician may highly suspect (but cannot guarantee) that a patient suffers from a specific disease. The uncertainty of such suspicion can be expressed in terms of *existential probability*. Since there are many real-life situations in which data are uncertain, efficient algorithms for mining uncertain data are in demand. Two algorithms have been proposed for mining frequent patterns from uncertain data. The previous two algorithms follow the horizontal data representation. In this paper we studied the problem of mining frequent itemsets from existential uncertain data using the Tidset vertical data representation. We introduced the U-Eclat algorithm, which is a modified version of the Eclat algorithm, to work on such datasets. A performance study is conducted to highlight the efficiency of the proposed algorithm also a comparative study between the proposed algorithm and the well known algorithm UF-growth is conducted and showed that the proposed algorithm outperforms the UF-growth.

References

- A.Ceglar and J. Roddick. (2006). "Association Mining", In ACM Computing Surveys, Vol.38, No.2, Article no. 5, July 2006.
- B. Goethals. (2004). "Memory Issues in Frequent Itemset Mining", In Proceedings of the ACM Symposium on Applied Computing (SAC), pp.530 - 534, March 2004.
- C. Aggarwal. (2009). "Managing and Mining Uncertain Data", Springer, 2009.
- C. Zhu, X. Zhang, J. Sun, and B. Huang. (2009). "Algorithm for Mining Sequential Pattern in Time Series Data", International Conference on Communications and Mobile Computing, pp. 258-262, January 2009.
- Chui, C.-K., Kao, B., Hung, E. Mining frequent itemsets from uncertain data. In: Zhou, Z.-Dai, X., Yiu, M.L., et al. (2005). Probabilistic spatial queries on existentially uncertain data. In: Bauzer Medeiros, C., Egenhofer, M.J., Bertino, E. (eds.) SSTD 2005. LNCS, vol. 3633, pp. 400–417. Springer, Heidelberg (2005)
- H. Kriegel, P. Kroger and A. Zimek, (2009). "Clustering high-dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering", ACM Transactions on Knowledge Discovery from Data (TKDD), Vol.3, No.1, march 2009.
- H., Li, H., Yang, Q. (2007). (eds.) PAKDD 2007. LNCS (LNAI), vol. 4426, pp. 47–58. Springer, Heidelberg (2007)
<http://kdd09.crowdvine.com/talks/show/4894>
- J. Han, H. Cheng, D. Xin, and X. Yan. (2007). "Frequent Pattern Mining: Current Status and Future Directions", Data Mining and Knowledge Discovery, Vol.15, No.1, pp. 55-86, 2007.
- J. Han, J. Pei, Y. Yin and R. Mao. (2004). "Mining Frequent Patterns without Candidate Generation: A Frequent-Pattern Tree Approach", Data Mining and Knowledge Discovery, Vol.8, No.1, pp. 53-87, 2004.
- Leung, C.K.-S., Carmichael, C.L., Hao, B. (2007). Efficient mining of frequent patterns from uncertain data. In: Proc. IEEE ICDM Workshops, pp. 489–494 (2007)
- M. Song, S. Rajasekaran. (2006). "A Transaction Mapping Algorithm for Frequent Itemsets Mining", IEEE Transactions on Knowledge and Data Engineering, Vol.18, No.4, pp. 472-481, April 2006.

M. Zaki and K. Gouda. (2003). "Fast Vertical Mining Using Diffsets", In Knowledge Discovery and Data Mining (KDD), pp. 326-335, 2003.

M. Zaki. (2000). "Scalable Algorithms for Association Mining", IEEE Transactions on Knowledge and Data Engineering , Vol.12, No.3, pp. 372-390, May-June 2000.

P. Shenoy, J. Haritsa, S. Sudarshan, G. Bhalotia, M. Bawa and D. Shah. (2000). "Turbo-Charging Vertical Mining of Large Databases", In ACM Special Interest Group on Management of Data (SIGMOD), Vol.29, No.2, June 2000.

W. Consue, and W. Kurutach. (2003). "Novel Vertical Mining on Diffsets Structure", In Proceedings of the IEEE/WIC International Conference on Intelligent Agent Technology (IAT), pp. 343-349, October 2003.

Y. Koh, N. Rountree, R. (2008). O'Keefe, "Mining Interesting Imperfectly Sporadic Rules", Knowledge and Information Systems ,Vol. 14 , No. 2, pp: 179-196, January 2008.

Z. Zheng, R. Kohavi, and L. Mason. (2001). "Real world performance of association rule algorithms", In ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD), pp: 401-406, 2001.

Table 1. Uncertain transactional data

TID	Items
T1	A (0.9), B(0.8), C(0.7), D(0.6), F(0.8)
T2	A(0.9), C(0.7), D(0.6), F(0.1)
T3	B(0.9), C(O.5), E(0.4)
T4	B(0.9), E(0.2)
T5	A(0.9), C(0.7), D(0.6), E(0.3)

Table 2. Utidset vertical representation of the all items

ITEM	A	B	C	D	E	F
TIDSET	T1(0.9)	T1(0.8)	T1(0.7)	T1(0.6)	T3(0.4)	T1(0.8)
	T2(0.9)	T3(0.9)	T2(0.7)	T2(0.6)	T4(0.2)	T2(0.1)
	T5(0.9)	T4(0.9)	T3(0.5)	T5(0.6)	T5(0.2)	
			T5(0.7)			
Exp_sup	2.7	2.6	2.6	1.8	0.9	0.9

Table 3. Tidset vertical representation of frequent items

ITEM	A	B	C	D
TIDSET	T1(0.9)	T1(0.8)	T1(0.7)	T1(0.6)
	T2(0.9)	T3(0.9)	T2(0.7)	T2(0.6)
	T5(0.9)	T4(0.9)	T3(0.5)	T5(0.6)
			T5(0.7)	
Exp_sup	2.7	2.6	2.6	1.8

Table 4. Utidset vertical representation of 2-itemsets

ITEM	ACD
TIDSET	T1(0.9 * 0.7 * 0.6)
	T2(0.9 * 0.7 * 0.6)
	T5(0.9 * 0.7 * 0.6)
Exp_sup	1.13

Table 5. Tidset vertical representation of frequent 3-itemsets

ITEM	AB	AC	AD	BC	BD	CD
TIDSET	T1(0.9 * 0.8)	T1(0.9 * 0.7)	T1(0.9 * 0.6)	T1(0.6 * 0.8)	T1 (0.8 * 0.6)	T1(0.7 * 0.6)
		T2(0.9 * 0.7)	T2(0.9 * 0.6)	T3(0.9 * 0.7)		T2 (0.7 * 0.6)
		T5(0.9 * 0.7)	T5(0.9 * 0.6)			T5 (0.7 * 0.6)
Exp_sup	0.72	1.89	1.62	1.11	0.48	1.26

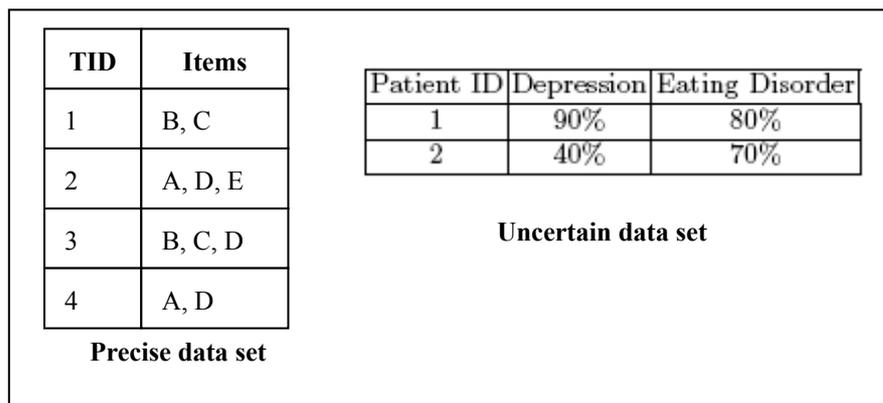


Figure 1. Example of precise and uncertain data sets

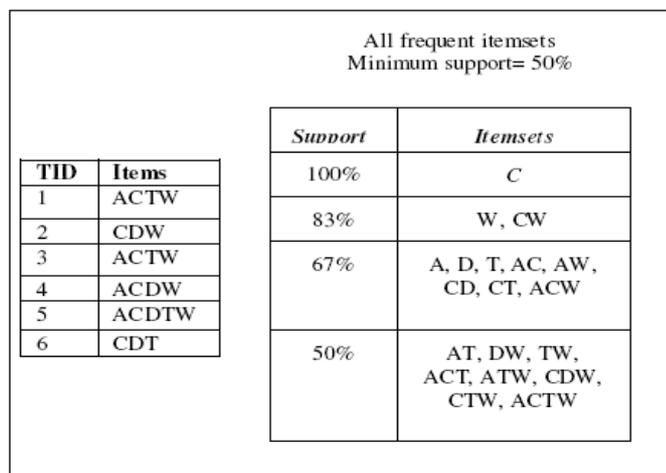


Figure 2. Illustrative example for mining vfrequent itemsets

	W_1	W_2	W_3	W_4	W_5	W_6	W_7	W_8
	A B	A B	A B	A B	A B	A B	A B	A B
t_1	✓✓	✓✓	✓✓	✓✗	✗✓	✓✓	✗✗	✓✗
t_2	✓✓	✓✗	✗✓	✓✓	✓✓	✗✗	✓✓	✓✗
	W_9	W_{10}	W_{11}	W_{12}	W_{13}	W_{14}	W_{15}	W_{16}
	A B	A B	A B	A B	A B	A B	A B	A B
t_1	✗✓	✗✓	✓✗	✗✗	✗✗	✗✓	✓✗	✗✗
t_2	✗✓	✓✗	✗✓	✓✗	✗✓	✗✗	✗✗	✗✗

Figure 3. Possible worlds from dataset with two transactions and two items

HORIZONTAL ITEMSET					VERTICAL TIDSET				
1	A	C	T	W	A	C	D	T	W
2	C	D	W		1	2	4	3	1
3	A	C	T	W	3	5	6	5	2
4	A	C	D	W	4	4	6	6	3
5	A	C	D	T	W	5			4
6	C	D	T		6				5

Figure 4. Horizontal and vertical representation of data

```

UEclat([P]):
For all  $X_i \in [P]$  do
  For all  $X_j \in [P]$ , with  $j > i$ 
     $R = X_i \cup X_j$ 
     $t(R) = t(X_i) \cap t(X_j)$ 
     $\sigma(R) = \sum (\prod p(R, t_i))$ 
    if  $\sigma(R) \geq \text{min sup}$  then
       $D_i = D_i \cup \{R\}$  //  $D_i$  initially empty
    Next j
  if  $D_i \neq \emptyset$ ; then UEclat( $D_i$ )
Next i
    
```

Figure 5. UEclat algorithm

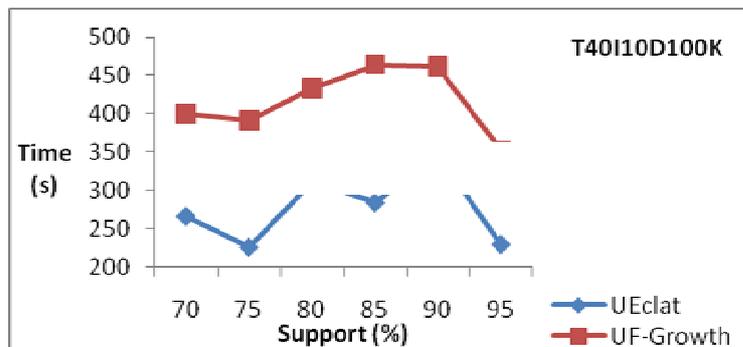


Figure 6. Run time for T40I10D100K
Data set for support from 70% to 95%

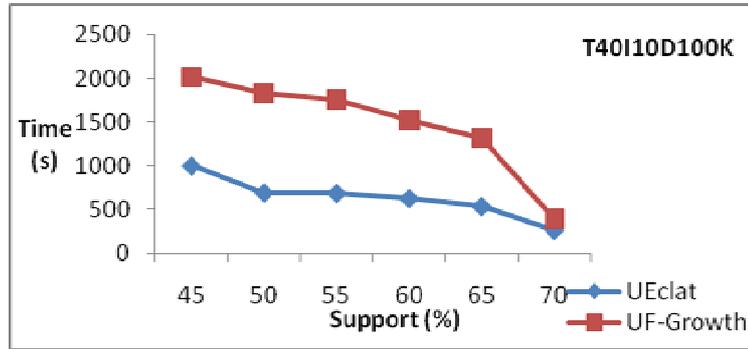


Figure 7. Run time for T40I10D100K
Data set for support from 45% to 70%

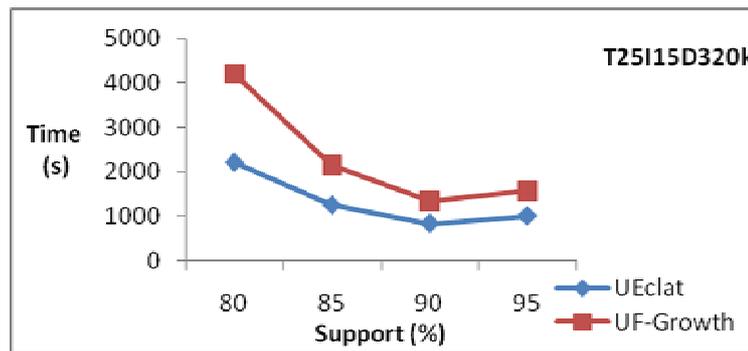


Figure 8. Run time for T25I15D320k
Data set for support from 80% to 95%

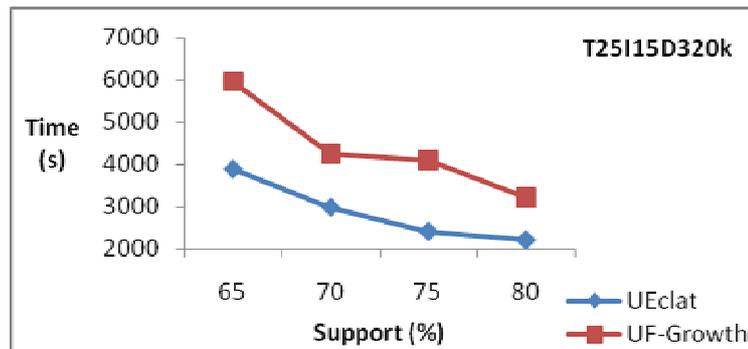


Figure 9. Run time for T25I15D320k
Data set for support from 65% to 80%