

Determining Weighted, Utility-Based Time Variant Association Rules using Frequent Pattern Tree

Determinación de las reglas de asociación de variables de tiempo ponderadas basadas en utilidades mediante la aplicación de un árbol de patrones frecuentes

Pankaj Gupta¹✉, Bharat Bhushan Sagar²

¹ Birla Institute of Technology, Mesra, Ranchi, India



² Birla Institute of Technology, Mesra, Ranchi, India



✉ BIT, off-campus, A-7, Sector-1, Noida, Mesra, Ranchi, India. Email: pgupta@bitmesra.ac.in

Received on: October 9th, 2017

Accepted on: February 16th, 2018

Available online: May 1st, 2018

How to cite this article: P. Gupta and B. Bhushan-Sagar, "Determining Weighted, Utility-Based Time Variant Association Rules using Frequent Pattern Tree", *Ing. Sol.*, vol. 14, no. 25 (Special issue), pp. 11 , May 2018. doi: <https://doi.org/10.16925/.v14i0.2228>

Abstract

Introduction: The present research was conducted at Birla Institute of Technology, off Campus in Noida, India, in 2017.

Methods: To assess the efficiency of the proposed approach for information mining a method and an algorithm were proposed for mining time-variant weighted, utility-based association rules using FP-tree.

Results: A method is suggested to find association rules on time-oriented frequency-weighted, utility-based data, employing a hierarchy for pulling-out item-sets and establish their association.

Conclusions: The dimensions adopted while developing the approach compressed a large time-variant dataset to a smaller data structure at the same time FP-tree was kept away from the repetitive dataset, which finally gave us a noteworthy advantage in articulations of time and memory use.

Originality: In the current period, high utility recurrent-pattern pulling-out is one of the mainly noteworthy study areas in time-variant information mining due to its capability to account for the frequency rate of item-sets and assorted utility rates of every item-set. This research contributes to maintain it at a corresponding level, which ensures to avoid generating a big amount of candidate-sets, which ensures further development of less execution time and search spaces.

Limitations: The research results demonstrated that the projected approach was efficient on tested datasets with pre-defined weight and utility calculations.

Keywords: association rule, frequent pattern tree, information mining, time variant, weighted transactions.

Determinación de las reglas de asociación de variables de tiempo ponderadas basadas en utilidades mediante la aplicación de un árbol de patrones frecuentes

Resumen

Introducción: la presente investigación se realizó en el Birla Institute of Technology, fuera del campus en Noida, India, en 2017.

Métodos: para evaluar la eficacia del enfoque propuesto para la minería de información, se propusieron un método y un algoritmo para minar las reglas de asociación basadas en la utilidad ponderada en el tiempo usando un árbol de patrones frecuentes (FP).

Resultados: se sugiere un método para encontrar reglas de asociación en datos basados en la utilidad ponderada en frecuencia orientada al tiempo, que emplea una jerarquía para extraer conjuntos de elementos y establecer su asociación.

Conclusiones: las dimensiones adoptadas al desarrollar el enfoque comprimieron un gran conjunto de datos de variante de tiempo hasta alcanzar una estructura de datos más pequeña. A su vez, el árbol FP se mantuvo alejado del conjunto de datos repetitivos, lo que finalmente generó una ventaja considerable en tiempo y uso de memoria.

Originalidad: en la actualidad, la extracción de patrones recurrentes de alta utilidad es una de las áreas de estudio más desarrollada en la minería de información con respecto a la variable temporal debido a su capacidad de dar cuenta de la frecuencia de los conjuntos de elementos y las tasas de servicios varios de cada conjunto de elementos. Esta investigación contribuye a mantener el estudio sobre el tema a un buen nivel, lo que permite evitar generar una gran cantidad de conjuntos posibles, y por ende garantiza mayor desarrollo en menores tiempos de ejecución y espacios de búsqueda.

Limitaciones: Los resultados de la investigación demostraron que la aproximación fue eficiente en conjuntos de datos probados con cálculos predefinidos de peso y utilidad.

Palabras clave: regla de asociación, árbol de patrones frecuentes, extracción de información, variante de tiempo, transacciones ponderadas.

Determinação das regras de associação de variáveis de tempo ponderadas baseadas em utilidades mediante a aplicação de uma árvore de padrões frequentes

Resumo

Introdução: esta pesquisa foi realizada no Instituto Birla de Tecnologia e Ciência, fora do campus, em Noida, na Índia, em 2017.

Métodos: para avaliar a eficácia do enfoque proposto para mineração de informação, foram propostos um método e um algoritmo para minar as regras de associação baseadas na utilidade ponderada no tempo usando uma árvore de padrões frequentes (FP).

Resultados: é recomendado um método para encontrar regras de associação nos dados baseados na utilidade ponderada em frequência orientada ao tempo, que emprega uma hierarquia para extrair conjuntos de elementos e estabelecer a associação entre eles.

Conclusões: as dimensões utilizadas ao desenvolver o enfoque comprimiram um grande conjunto de dados de variante de tempo até alcançar uma estrutura de dados menor, enquanto isso, a árvore FP se manteve distante do conjunto de dados repetitivos, o que finalmente gerou uma vantagem considerável em tempo e uso de memória.

Originalidade: na atualidade, a extração de padrões recorrentes de alta utilidade é uma das áreas de estudo mais desenvolvidas na mineração de informação com respeito à variável temporal, devido a sua capacidade de dar conta da frequência dos conjuntos de elementos e das taxas de serviços vários de cada conjunto de elementos. Esta pesquisa ajuda a manter o estudo desse tema em um nível avançado, o que garante evitar gerar uma grande quantidade de conjuntos possíveis e, dessa forma, um maior desenvolvimento em um menor tempo de execução e espaço de busca.

Limitações: os resultados da pesquisa demonstraram que a aproximação foi eficiente em conjuntos de dados provados com cálculos predefinidos de peso e utilidade.

Palavras-chave: árvore de padrões frequentes, extração de informação, regra de associação, transações ponderadas, variante de tempo.

1. Introduction

Time variant information mining may be considered like a step-by-step algorithmic procedure that acquires time-variant information as input and gives patterns, recurrent item-sets, association rules, or summaries as output. Mining recurrent patterns in large time-variant transactional datasets is currently a focused study-area in data-mining. Several presented time-variant recurrent item-sets realizing algorithms go through various problems in respect to computational-time, input-output cost, and memory-space requirements when we are dealing with a large amount of time-variant transactional data [1]. Recurrent item-sets mining in transactional dataset is based only on their occurrence without considering their utility-factor. However, for a lot of actual submissions, utility of an item-set based on their price, earnings, revenue or some other factor is significant. Utility-based frequency-weighted mining problems are used to locate frequent item-sets which have higher utility than a user specified minimum-one. Unlike a priori-algorithm [2], [3], which is affirmed on support and confidence framework, mining of frequency-weighted recurrent item-sets on their utility-base from time-variant transactional dataset poses a greater challenge [4], [5].

Mining high utility based item-sets from a time-variant transaction dataset is done to discover item-sets which have higher utility than a user precise threshold. Utility pulling out is an addition to recurrent item-sets mining that occurs frequently in time-variant datasets [6], [7].

Recurrent pattern mining methods take care of all items in the dataset uniformly by simply considering the existence of an item-set inside a time-variant transaction. Though, the utility of mined frequent item-sets may vary with user's perspective. Elevated utility based on pulling out approaches has been projected to defeat this difficulty. Thus, it is important to consider the study subject in data mining and finding facts with a utility factor, which reduces needless computations while the dataset is modified, or else while the minimum threshold is altered. The majority of recurrent pattern mining algorithms, together with a priori [2], [3], frequent pattern-growth [8], [9], H-mine find out all frequent item-sets. These algorithms have good presentation in case support-value benchmark is elevated, but while support-value target goes

down, the amount of recurrent item-sets increases considerably, and the efficiency of these algorithms declines rapidly due to the production of a large amount of item-sets.

In this study a well-organized tree hierarchical based organization for pulling out the elevated utility and frequency-weighted item-sets competently was planned. A new utility-based frequency-weighted FP-tree, which stores the information about utility based in recurrent item sets in time-variant transactional datasets was projected. The competence of this approach can be understood by the thought that this approach compressed a large time variant dataset to a lesser data structure, also the utility based frequency-weighted Frequent Pattern tree structure was kept away from repetitive dataset scans which avoid generating a large number of candidate sets and thus save time and search memory space. The presentation assessment of the projected approach is competent in mining time-variant elevated utility frequency-weighted item-sets while compared with the typical Efficient Tree Structures for High Utility Pattern Mining in Incremental Databases (IHUP-TWU algorithm) [10], [11] and Frequent Pattern Growth algorithms [12].

2. Objective

The objective of this research is to study the association rules between time variant items on the basis of their weight and utility in the transactional data and subsequently use the FP-tree structure to generate item-sets with effective relationship.

3. Literature Review

In the current scenario, the mounting move towards utility-based weighted recurrent pattern pulling-out has put on massive significance in actual-life applications. A short analysis of a few of the current noteworthy studies is presented next. Hu and Mojsilovic [13] proposed an algorithm of recurrent item-set pulling-out, which recognizes elevated utility item arrangements. In disparity with conventional association-rule and recurrent item pulling-out methods, the purpose of the algorithm is to find out the part of information that assures certain situations as a collection and maximizes a pre-defined purpose. Yeh et al. [14] projected the isolated items discarding approach

that can be useful to any presented stage-wise utility-mining method to decrease candidate-sets, generate and thus develop better results and performance.

Yao and Hamilton [15] projected a utility-based item set pulling-out approach that allows the client to enumerate their inclinations using the worth of an item set distinguished like a utility restriction. Thus, an item set may be appealing to the client simply if it suits a specified utility-restriction.

Ahmed et al. [16] projected a tree-support candidate-set trimming mechanism i.e. elevated utility candidates trim to competently mine high-utility recurrent patterns without stage-wise candidate-set production. Trial outcomes demonstrate that the method was extremely competent for elevated utility pattern pulling-out and it functioned well with the existing procedures.

Presently numerous procedures exist for pulling-out elevated utility item-sets by trimming candidate-sets supported on expected utility-worth and on transaction-weighted consumption worth. These procedures intend to decrease time and search storage space.

A time variant characteristic of association-ruling was anticipated by Tseng et al. [17], Taha [18] projected a group of item patterns employing a tree-hierarchical arrangement identical to FP-tree. Li [19] stated that time-variant a priori is much faster than straightforward a priori algorithm.

In our methodology, the items' time validity, which includes the initial run through emergence and ending time when the item appears in time-variant transactional databases, were considered. We computed the frequency based weighted time variant support [20] of an item set exemplified by its utility value, and after that, we built frequent pattern trees to extract recurrent patterns and their affiliation. Contribution of frequency-based weight can competently suggest time variants highly useful item sets with fewer candidate item-sets in such way that the concluding time can be reduced efficiently.

4. Background

A few assumptions and explanations of current problems of frequency-weighted utility-based mining of time variant association-rules were described.

4.1. Preliminary

Consider a finite set of items $It = \{it-1, it-2 \dots it-m\}$ where each item has utility yield 'iy'. A time-variant transactional dataset $TD = \{t1, t2 \dots, tn\}$ contains a set of transactions, with each transaction a valid start and stop time. Every transaction $tp (1 \leq p \leq n)$ has a unique identifier p , described as transaction-id. Every transaction 'tp' is related with the number of item sets 'it' ($1 \leq It \leq m$).

Definition 1: Frequency-Weight of an item 'it-x' in a transactional dataset TD is denoted as $\sum it-x (TD)$ within valid time legacy of transactions.

Definition 2: Utility of an item set 'it-x' in tp is denoted as $U(it-x)$.

Definition 3: An item-set is described as high utility item-set if its utility-yield is not less than a user-specific minimum utility threshold, which is denoted as $min_req_utility$; otherwise, it is called a low utility item-set.

4.2. Time-variant Association Ruling

Episode of an item set 'it' in a session of time span 'T' is the sum of the figure of transactions in which it arises separated by entire figures of transaction in that time-span [21]. Whereas confidence of an item set with other item sets is the transaction of these items collectively over the session, separated by the initial item set of that session.

$$\text{Time variant_Support (X)} = \frac{\text{Frequency of occurrence of X in time interval [Ts, Te]}}{\text{Total no., of tuples in time interval [Ts, Te]}} \quad (1)$$

$$\text{Time variant_Confidence (X>>Y [Ts, Te])} = \frac{\text{Support_count (XUY) over interval}}{\text{Occurrence of X in time interval}} \quad (2)$$

Where 'Ts' is the valid beginning time and 'Te' is the valid ending time of the time-variant data.

4.3. Frequency-Weighted Time variant Association Rule Mining

It consists of a projected measurement of weighted support that does not require pre-assigned weights. Our anticipated way to figure out weighted support

based on this approach, is to compute time variant-weighted-support and time variant weighted confidence of items [4] as follows:

$$\text{Frequency-Weighted time variant-support, FWTVS(it)} = \frac{t[\text{time_initiate, time_stop}] \sum_{it} (q(it-t))}{\sum_n [\text{time_initiate, time_stop}]} \quad (3)$$

Where, 'it' = 1;2;3;...m and t = 1;2;3;...n and 'q(it-t)' stand for number of items it ∈ It, in t-th transaction within its valid time-period and Σn is the sum of all transactions within its valid time-period of all items.

$$\text{Frequency-weighted time variant confidence (X⇒Y [time_initiate, time_stop])} = \frac{\text{Time variant-weighted support (XUY) over time gap}}{\text{Occurrence of X in time gap}} \quad (4)$$

The primary objective is to keep the closure property in case of frequency-weighted items too and if weights of each entity item are similar to weighted-support this should also be similar [22].

4.4. Frequency-Weighted Time variant Utility

The significance of an item in item set and of a transaction in transactional datasets gives their usefulness feature [23], [24]. In the construction of the Frequent Pattern tree the arrangements of the transaction are significant. The arrangements depend on the frequency based weighted time variant utility-base of an item set (FWTVU). The item sets are arranged in decreasing-array on behalf of every transaction based on the frequency based weighted time variant utility rate and the item-sets that have less than the required minimum time variant utility rate are detached from transactions. i.e. $\text{FWTVU(it)} \geq \text{min_req_utility_support}$.

Thus, Frequency-Weighted time variant utility, 'FWTVU' of an item 'it', denoted by FWTVU(it) , is computed as follows:

$$\text{FWTVU(IT)} = \text{FWTVS(IT)} * U(IT) / \sum U \quad (5)$$

Where, U(it) is the external utility associated with an item set and ΣU is the addition of utility of all item sets that are present within its valid time-period.

4.5. Problem statement

Consider a given time-variant transactional dataset "TD" and a user-specific minimum required utility threshold support i.e. 'min_req_utility_support'. The problem of mining time-variant frequency-weighted high utility item sets from "TD" is to find the complete set of high frequency item sets whose utilities are larger than or equal to min_req_utility_support. After this, to insert transactions into the FP-tree and obtain frequent item sets and their association.

5. Methods

To assess the efficiency of the proposed approach for information mining a method and algorithm was proposed for mining time-variant weighted, utility-based association rules using FP-tree. The major objective of the anticipated algorithm is to keep utility based frequency-weighted time variant recurrent item sets employing Frequent Pattern tree and then extract the imprints as per utilizer threshold usefulness feature. Steps are as follows:

In-put: Time-varying transactional-dataset, minimum required frequency-weighted utility support.

Out-put: High utility frequency-weighted time variant association-rules.

Step 1. Input time-varying transactional dataset.

Step 2. Convert time-varying transactional-dataset into frequency-weighted evaluation table to compute frequency-weight for each item.

Step 3. Compute frequency-weighted time variant support (FWTVS) as specified in sub-section '3.3' for every item set within its valid-time-period.

Step 4. Compute frequency-weighted time variant utility support (FWTVU) as specified in sub-section '3.4' for every item set within its valid time-period.

Step 5. Prune the item sets which do not possess the required minimum time variant utility i.e. $\text{FWTVU(it)} \geq \text{min_req_utility_support}$ and arrange the transaction in descending order.

Step 6. Create a Frequent Pattern tree structure and mine the recurrent patterns from them as per user threshold utility.

Step 7. Obtain association rules.

Example: In a transactional dataset as shown in Table 1, a set of time-variant dataset shows the transaction records of two quarters, whereas Table 2 shows the usefulness i.e., utility of each item set.

Step 1: Input time-varying transactional dataset as suggested in Table 1.

Table 1. Example of a transactional dataset

Valid timelegacy	Transaction id	Transaction items
Quarter-1	t1	B, C
	t2	B, C, D
	t3	B, C
	t4	A, D
Quarter-2	t5	B, C, E
	t6	D, E
	t7	A, B, C
	t8	C, D, E

Reference: the authors

Table 2. Transactional item's utility

Items	Item per unit yield
A	3
B	2
C	1
D	4
E	5

Reference: the authors

Step 2: Convert transactional dataset into frequency-weighted evaluation table to calculate the weight of each item within its valid time period as shown in Table 3.

Table 3. Weight assessment of transactional items

Valid time legacy	Items	T-id	Frequency weight evaluation (FW)
Quarter-1	A	t4	1
	B	t1, t2, t3	3
	C	t2, t3	2
	D	t1, t2, t4	3
Quarter-2	A	t7	1
	B	t5, t7	2
	C	t5, t7, t8	3
	D	t6, t8	2
	E	t5, t6, t8	3

Reference: the authors

Step 3: Computation frequency-weighted time variant support (FWTVS) for items within their valid time period by using equation 3.

For Quarter-1

$$FWTVS(A) = FW(A) / \sum U = 1/9 = 0.11$$

$$FWTVS(B) = FW(B) / \sum U = 3/9 = 0.33$$

$$FWTVS(C) = FW(C) / \sum U = 2/9 = 0.22$$

$$FWTVS(D) = FW(D) / \sum U = 3/9 = 0.33$$

For Quarter-2

$$FWTVS(A) = FW(A) / \sum U = 1/11 = 0.09$$

$$FWTVS(B) = FW(B) / \sum U = 2/11 = 0.18$$

$$FWTVS(C) = FW(C) / \sum U = 3/11 = 0.27$$

$$FWTVS(D) = FW(D) / \sum U = 2/11 = 0.18$$

$$FWTVS(E) = FW(E) / \sum U = 3/11 = 0.27$$

Step 4: Computation frequency-weighted time variant utility support (FWTVU) for items within their valid time period using equation 5.

For Quarter-1

$$FWTVU(A) = FWTVS(A) * U(A) / \sum U = 0.11 * 3 / 15 = 0.022$$

$$FWTVU(B) = FWTVS(B) * U(B) / \sum U = 0.33 * 2 / 15 = 0.044$$

$$FWTVU(C) = FWTVS(C) * U(C) / \sum U = 0.22 * 1 / 15 = 0.014$$

$$FWTVU(D) = FWTVS(D) * U(D) / \sum U = 0.33 * 4 / 15 = 0.088$$

For Quarter-2

$$FWTVU(A) = FWTVS(A) * U(A) / \sum U = 0.09 * 3 / 15 = 0.018$$

$$FWTVU(B) = FWTVS(B) * U(B) / \sum U = 0.18 * 2 / 15 = 0.024$$

$$FWTVU(C) = FWTVS(C) * U(C) / \sum U = 0.27 * 1 / 15 = 0.018$$

$$FWTVU(D) = FWTVS(D) * U(D) / \sum U = 0.18 * 4 / 15 = 0.048$$

$$FWTVU(E) = FWTVS(E) * U(E) / \sum U = 0.27 * 5 / 15 = 0.09$$

Step 5: Prune the item-sets as per, $FWTVU(i) \geq \min_req_utility_support$ and arrange the transaction in descending order. Say *required min_req_utility_support = 0.022*, as shown in Table 4.

Table 4. Modified arrangements

Valid time legacy	Transaction id	Transaction items
Quarter-1	t1	B
	t2	D, B
	t3	B
	t4	D, A
Quarter-2	t5	B
	t6	D
	t7	B
	t8	D

Reference: the authors

Step 6: Insert transactions into frequency-weighted time variant utility frequent pattern tree as shown below. After processing the first transaction for the first quarter into FP-tree (Fig.1).

Repeat subsequently after the insertion until reaching the last transaction for quarter 1 (Fig. 2).

Similarly, after inserting the transactions for the second quarter, get the FP-tree for all the transactions for all valid time periods as shown in Fig. 3.

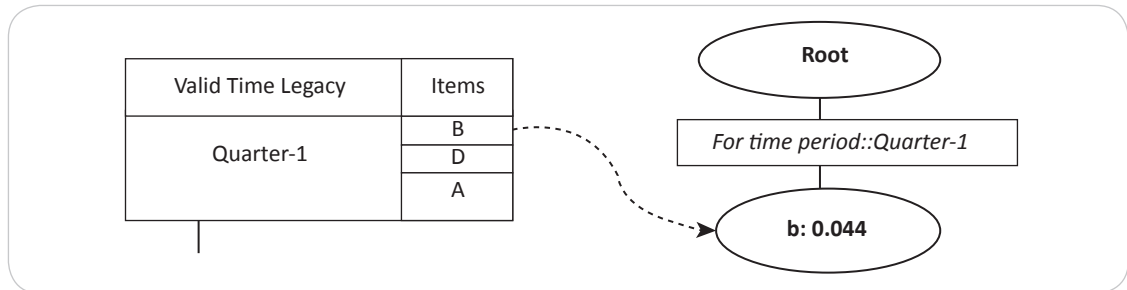


Fig. 1. FP-tree after processing first transaction for quarter1

Reference: the authors

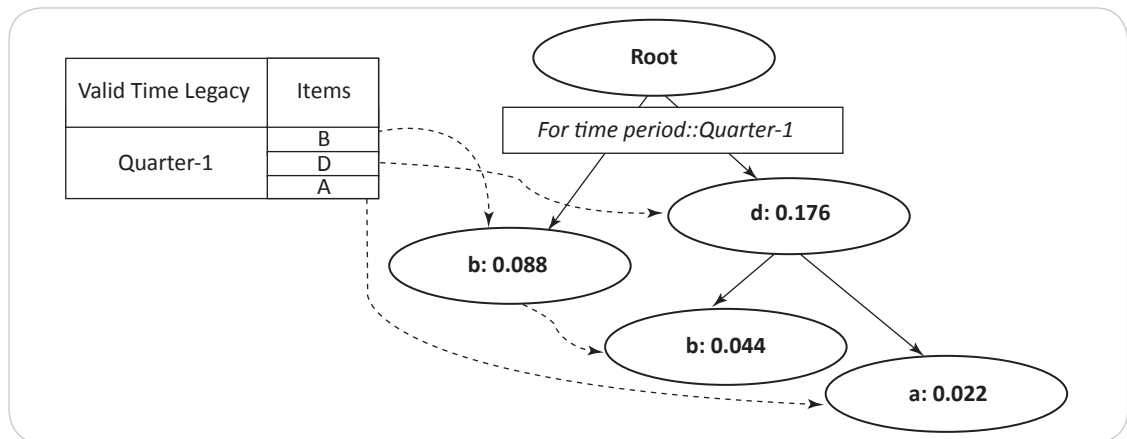


Fig. 2. FP-tree after processing last transaction for quarter 1

Reference: the authors

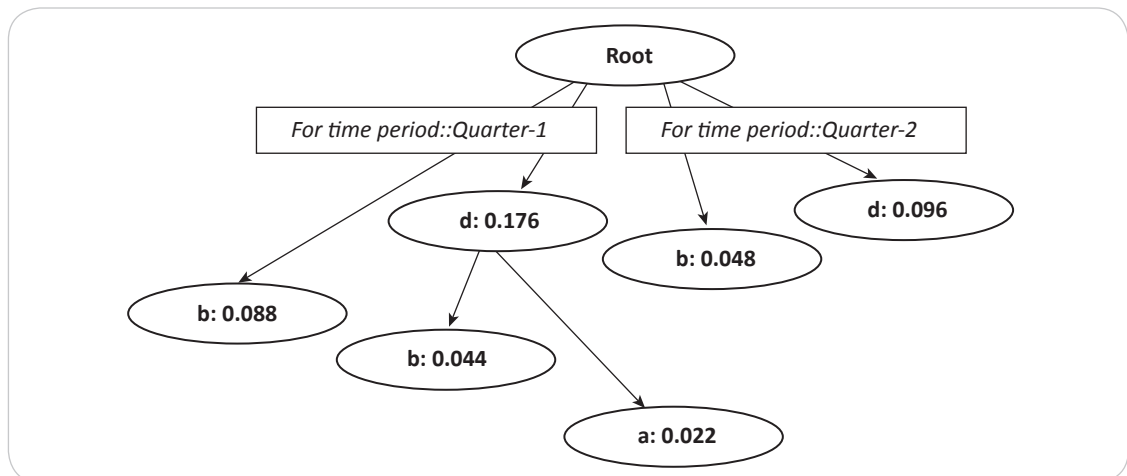


Fig. 3. Complete FP-tree for transactional dataset

Reference: the authors

Step 7: Mine the frequency-weighted utility item sets from FP-tree (Fig. 3) based on conditional frequent pattern methodology as shown in Table 5.

After mining the time variant frequent item sets, the association-rules as per required minimum utility support value are obtained.

So here, only two frequent item sets (DA and DB) have min_req_utility_support within its valid time period.

Thus the possible rules are:

- D → A, A → D (for time-period: quarter-1)
- D → B, B → D (for time-period: quarter-2)

6. Results

Test are carried out on a personal desktop Pentium-5, 3.6 MHz with 6GB RAM and we apply the proposed algorithm using java as programming language.

Time variant synthetic data sets and real world data sets “calamity” are both obtained from FIMI Repository (<http://fimi.cs.helsinki.fi/>). Descriptions and default values of synthetic and real data sets are shown in Table 6. Testing is done mixing both real and synthetic time variant data-sets with these three recurrent pattern-mining algorithms i.e. projected time variant frequency weighted utility algorithm, Efficient Tree Structures for High Utility Pattern Mining in Incremental Databases (IHUP-TWU algorithm) and standard frequent pattern growth. These algorithms are used to investigate IHUP-TWU algorithm and the standard frequent pattern tree-growth algorithm with our projected approach for efficiently mining elevated utility time-variant item sets; unusual outcomes are achieved by shifting the utility support values.

Table 5. Mining time variant frequent patterns from FP-tree

Item	Time period	Conditional pattern base	Conditional FP-tree	Frequent item
A	Quarter1	{d:0.022}	{d:0.022} a	{da:0.022}
A	Quarter2	∅	∅	∅
B	Quarter1	{d:0.044}	{d:0.044} b	{db:0.044}
B	Quarter2	∅	∅	∅
D	Quarter*	∅	∅	∅

Reference: the authors

Table 6. Test dataset description

Dataset	Size	Number of transactions (TD)	Number of distinct items (It)
T10I4D100K	3.95MB	100,000 (Divided into two terms)	867
Calamity	5.07MB	79,175 (Divided into two terms)	472

Reference: the authors

6.1. For Calamity

The outcomes obtained are designed in a grid (where the first 3 horizontal levels represent term 1, as time period and then the 3 upper horizontal levels represent term 2, as time period) as shown in Figure 4a-4c, which shows better evaluated performances by our projected approach in comparison to IHUP-TWU algorithm and standard frequent pattern tree in terms of number of different length patterns generated as the required minimum utility support varies.

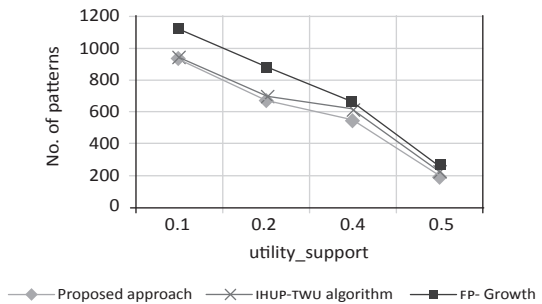


Fig. 4a. Number of patterns (one-length) produced using different utility-support

Reference: the authors

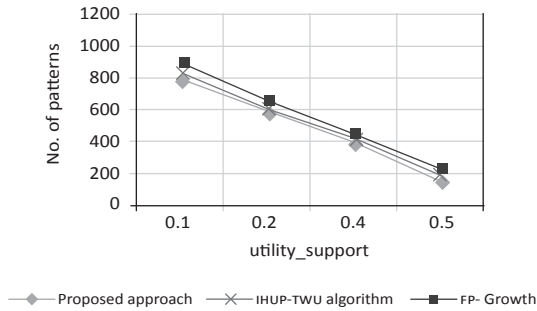


Fig. 4b. Number of patterns (two-lengths) produced using different utility-support
Reference: the authors

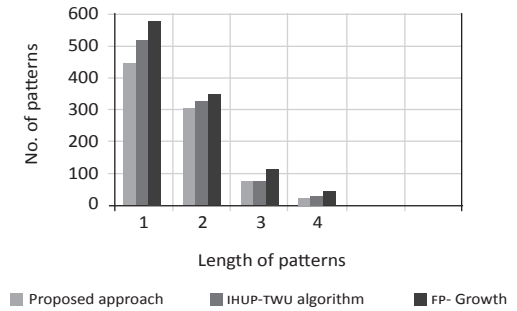


Fig. 5a. Number of patterns of varying length with utility-support=0.5
Reference: the authors

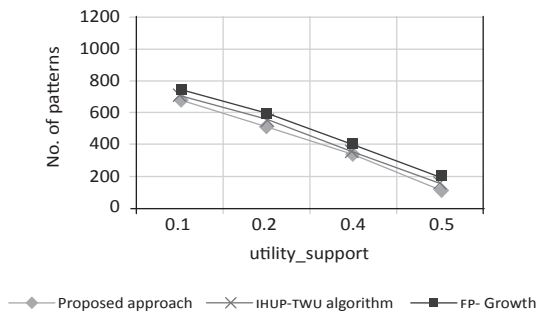


Fig. 4c. Number of Patterns (three-lengths) produced using different utility-support
Reference: the authors

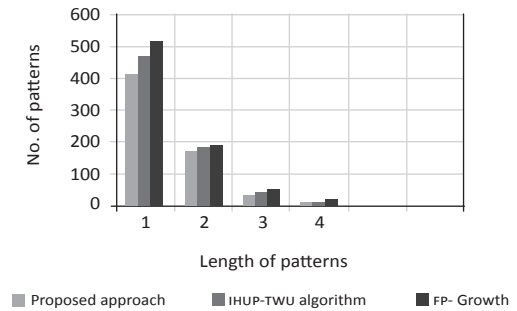


Fig. 5b. Number of patterns of varying length with utility-support=0.6
Reference: the authors

The test results show that the proposed approach works better than the state-of-the-art algorithms roughly in all cases.

6.2. For T10I4D100K

The outcomes are obtained by varying the utility support vales on T10I4D100K dataset. The plotted graphs are the results (where the first 3 horizontal levels represent term-1 as time period and the upper 3 horizontal levels represent term-2 as time period) shown in Fig.5a-5c, which shows the performance of our projected approach in comparison to the IHUP-TWU algorithm and standard frequent pattern tree in terms of number of patterns of different lengths generated as the required minimum utility support varies.

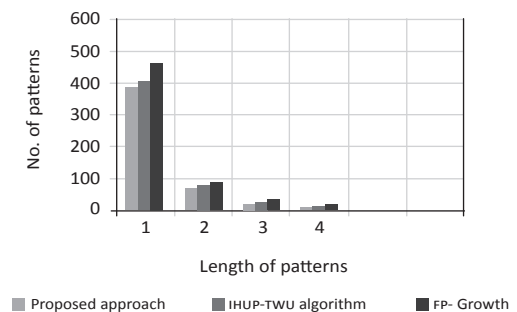


Fig. 5c. Number of patterns of varying length with utility-support=0.7
Reference: the authors

6.3. Scalability of the Proposed Approach

The following figure shows the scalability of the compared methods. The test is performed on time variant synthetic datasets, the outcomes are shown in Fig. 6.

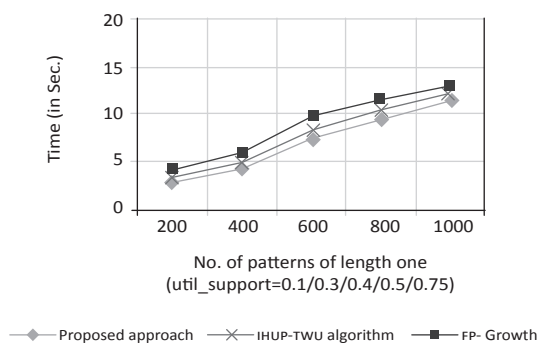


Fig. 6. Total runtime (sec) of different approaches with dataset 'T10I4D100K'

Reference: the authors

7. Discussion

The experimental results are obtained by varying the utility_support values on the calamity datasets that show the proposed approach's performance is capable in mining high weighted-utility itemsets. The performance is evaluated over different utility_support values (between 0.1 to 0.5) and the equivalent generated number of patterns. Our approach obtains better outcomes in comparison to standard FP-growth and IHUP-TWU algorithms by avoiding the repetitive item-sets.

The experimental results are taken by varying the utility_support values on the T10I4D100K datasets. The performance is studied by the corresponding number of patterns generated with varying utility_support, which shows that the projected approach generates less in comparison to the other two algorithms.

Also, the projected approach reduces the overall runtime by generating less numbers of intermediate candidate sets.

This study will help the researcher to uncover the critical attribute weight and utility of item-sets in transactional data, which subsequently reduces the generation of repetitive groups, which finally gives us more concrete results and association rules.

8. Conclusions

In the current study, we addressed an incipient mixed approach for the information mining process. The anticipated algorithm gives a proficient

time and search-space responsive approach for mining repeated items in data sets. This study offered a new frequency-weighted utility frequent pattern tree structure, for accumulating necessary information about recurrent patterns, for pulling-out the high-utility item-sets. In this study, the use of regular frequent pattern growth procedure for pulling-out the entire set of frequency-weighted high utility recurrent patterns by FP-growth, which avoid expensive production of a large number of candidate sets and thus, efficiently lessen execution time and search spaces is shown. The study also demonstrated that the projected approach is efficient on tested datasets.

References

- [1] R. Ivancsy and I. Vajk, "Fast discovery of frequent itemsets: a cubic structure-based approach", *Informatica*, vol. 29, pp. 71-78, 2005 [Online]. Available: <https://pdfs.semanticscholar.org/9356/284f90973ba254f2cea9ef5426773883da12.pdf>
- [2] R. Agrawal, T. Imielinski, and A. Swami, "Mining association rules between sets of items in large databases", in *Proceedings of the ACM sigmod international conference on management of data*, Washington, D.C., 1993, pp. 207-216. doi: <https://doi.org/10.1145/170036.170072>
- [3] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules", in *Proceedings of the 20th international conference on very large databases*, Santiago, Chile, 1994, pp. 487-499, [Online]. Available: <http://www.vldb.org/conf/1994/P487.pdf>
- [4] W. Wang, J. Yang, and P. S. Yu, "Efficient mining of weighted association rules", in *Proceeding of the sixth ACM SIGKDD international conference on knowledge discovery and data mining*, San Francisco, CA, USA, 2000. doi: <https://doi.org/10.1145/347090.347149>
- [5] N. B. Nazar and R. Senthilkumar, "An online approach for feature selection for classification in big data", *Turkish Journal of Electrical Engineering & Computer Sciences*, vol. 25, pp. 163-171, 2017. DOI: <https://doi.org/10.3906/elk-1501-98>
- [6] J. Pillai, S. Soni, O. P. Vyas, and M. Muyeba, "A conceptual approach to temporal weighted itemset utility mining", *International Journal of Computer Applications*, vol. 28, pp. 0887-0895, 2010. doi: <https://doi.org/10.5120/510-827>
- [7] Yh. Kim, Wy. Kim, and Um. Kim, "Mining frequent itemsets with normalized weight in continuous data streams", *Journal of Information Processing Systems*, vol. 6, pp. 79-90, 2010. doi: <https://doi.org/10.3745/JIPS.2010.6.1.079>

- [8] B. Christian, "An implementation of the FP-growth algorithm", in *Proceedings of the 1st international workshop on open source data mining: frequent pattern mining implementations*, Chicago, Illinois, 2005, pp. 1-5. doi: <https://doi.org/10.1145/1133905.1133907>
- [9] G. Gatuha and T. Jiang, "Smart frequent itemsets mining algorithm based on fp-tree and diffset data structures", *Turkish Journal of Electrical Engineering & Computer Sciences*, vol. 25, pp. 2096-2107, 2017. doi: <http://doi.org/10.3906/elk-1602-113>
- [10] V. S. Tseng, B. E. Shie, C. W. Wu, and P. S. Yu, "Efficient algorithms for mining high utility itemsets from transactional databases", *IEEE TKDE*, vol. 25, pp. 1772-1786, 2013. doi: <http://doi.org/10.1109/TKDE.2012.59>
- [11] V. Goyal and A. Sureka, "Efficient skyline itemsets mining", in *Proceedings of the eighth international C3S2E conference on computer science & software engineering*, Japan, 2015. doi: <https://doi.org/10.1145/2790798.2790816>
- [12] S. Cankurt and A. Subasi, "Tourism demand modeling and forecasting using data mining techniques in multivariate time series: a case study in turkey", *Turkish Journal of Electrical Engineering & Computer Sciences*, vol. 24, pp. 3388-3404, 2016. doi: <https://doi.org/10.3906/elk-1311-134>
- [13] J. Hu and A. Mojsilovic, "High utility pattern mining: a method for discovery of high utility item sets", *Pattern Recognition*, vol. 40, pp. 3317-3324, 2007. doi: <https://doi.org/10.1016/j.patcog.2007.02.003>
- [14] Y. C. Li, J. S. Yeh, and C. C. Chang, "Isolated items discarding strategy for discovering high utility itemsets", *Data & Knowledge Engineering*, vol. 64, pp. 198-217, 2008. doi: <https://doi.org/10.1016/j.datak.2007.06.009>
- [15] H. Yao and H. J. Hamilton, "Mining itemset utilities from transaction databases", *Data & Knowledge Engineering*, vol. 59, pp. 603-626, 2006. doi: <https://doi.org/10.1016/j.datak.2005.10.004>
- [16] C. F. Ahmed, S. K. Tanbeer, B. S. Jeong, and Y. K. Lee, "An efficient candidate pruning technique for high utility pattern mining", *Advances in Knowledge Discovery and Data Mining, Lecture Notes in Computer Science*, vol. 76, pp. 749-756, 2009. doi: https://doi.org/10.1007/978-3-642-01307-2_76
- [17] V. S. Tseng, C. J. Chu, and T. Liang, "An efficient algorithm for mining temporal high utility itemsets from data streams", *Journal of System and Software*, vol. 81, no. 7, pp. 1105-1117, 2008. doi: <https://doi.org/10.1016/j.jss.2007.07.026>
- [18] A. Ahmed, N. Makky, and Y. Taha, "Incremental mining of constrained association rules", in *Proceeding of the First Siam Conference on Data Mining*, Atlanta, 2008. doi: <https://doi.org/10.1137/1.9781611972719.24>
- [19] Y. Li, P. Ning, X. S. Wang, and S. Jajodia, "Discovering calendar based temporal association rules", *Data and Knowledge Engineering*, vol. 44, pp. 193-214, 2003. doi: [https://doi.org/10.1016/S0169-023X\(02\)00135-0](https://doi.org/10.1016/S0169-023X(02)00135-0)
- [20] P. Gupta and B. B. Sagar, "Discovering Interesting Weighted Temporal Relationship Rules", *International Journal of Control Theory and Applications*, vol. 9, no. 19, pp. 9091-9099, 2016. doi: <https://doi.org/10.17485/ijst/2016/v9i28/98455>
- [21] J. M. Ale and G. H. Rossi, "An approach to discovering temporal association rules", in *ACM Symposium on Applied Computing*, Italy, 2000, pp. 294-300. doi: <https://doi.org/10.1145/335603.335770>
- [22] V. Mittal and I. Kashyap, "Empirical study of impact of various concept drifts in data stream mining methods", *International Journal of Intelligent Systems and Applications (IJISA)*, vol. 8, pp. 65-72, 2016. doi: <https://doi.org/10.5815/ijisa.2016.12.08>
- [23] W. Novitasari, A. Hermawan, Z. Abdullah, and T. Herawan, "A method of discovering interesting association rules from student admission dataset", *International Journal of Software Engineering and Its Applications*, vol. 9, pp. 51-66, 2015. doi: <https://doi.org/10.14257/ijseia.2015.9.8.05>
- [24] J. Wang, "Application of association rule mining algorithm in logistics information system design", *Journal of Software Engineering*, vol. 11, pp. 217-22370, 2017. doi: <https://doi.org/10.3923/jse.2017.217.223>