Journal of Computer Science 10 (10): 2105-2116, 2014
ISSN: 1549-3636

doi:10.3844/jcssp.2014.2105.2116 Published Online 10 (10) 2014 (http://www.thescipub.com/jcs.toc)

FREQUENT CORRELATED PERIODIC PATTERN MINING FOR LARGE VOLUME SET USING TIME SERIES DATA

[1]Karthik, G.M. and [2]S. Karthik

[1]Assistant Professor, Department of Computer Science and Engineering,
SACS MAVMM Engineering College, Madurai, Tamil Nadu, India
[2]Professor and Dean,Department of Computer Science and Engineering
SNS College of Technology, Coimbatore, Tamil Nadu, India

Received 2014-02-27; Revised 2014-03-21; Accepted 2014-06-30

ABSTRACT

Frequent pattern mining has been a widely used in the area of discovering association and correlations among real data sets. However, discovering interesting correlation relationship among huge number of co-occurrence patterns are complicated, a majority of them are superfluous or uninformative. Mining correlations among large pile of useless information is extraordinarily useful in real-time applications. In this study, we propose a technique uses FP-tree for mining frequent correlated in periodic patterns from a transactional database. The analysis of time correlation measure tend to improvise the performance based on real time data sets and the result proves the algorithm efficiency by shifting the data sets to various domain towards time series, its correlation and noise-resilient ratio. This work addresses the time correlation factor achieved with the previous evaluated result of time series sequence of FP tree.

Keywords: Time Series Data, Time Correlation, Frequent Pattern Mining

1. INTRODUCTION

The concept of frequent pattern mining used extensively in the field of data mining. The association rule mining (Han and Pei, 2000), sequential pattern mining (Pei and Han, 2002), graph pattern mining (Yan and Han, 2002) are the few common approaches used in it. The real complication occurs in terms of real data sets. The real challenge is gather similar useful pattern collected from a large volume of information that catches the researcher concentration (Hasan *et al*., 2007; Chen *et al*., 2008).

The piles of data are gathered with similar behavior at identical time interval and its series which brings disrepute prior to analysis (Elfeky *et al*., 2005a; Han *et al*., 1999). The observation is to categorize duplicate patterns that provide important observations and its updated information of time series data (Weigend and Gershenfeld, 1994; Versaci, 2014) and assist in decision making based on the result achieved (Rasheed *et al*., 2011). A time series (Sheng *et al*., 2005a) is said to have three type of periodic pattern: (1) Symbol periodicity, (2) sequence periodicity or partial periodic pattern and (3) segment or full-cycle periodicity (Rasheed *et al*., 2011). For example, in time series contain the hourly number of transactions in retail store; the mapping different ranges of transactions (is referred as discreet process); a: {0} Transactions, b: {1-300} Transactions, c: {301-600} transactions, d: {601-1200} transactions, e: {>1200} Transactions. Based on this mapping, the time series T' = 0,212, 535, 0, 398, 178, 0, 78, 0, 0, 102, 423 can be discreet into *T* = *abdacbabaabc*. At least one symbol is repeated periodically in time series *T* is referred as *Symbol periodicity*. For example T = a bd a cb a ba a bc, symbol *'a'* is periodic with periodicity p = 3, starting at position zero. *Sequence periodic or partial periodic pattern* consists of more than one symbol, maybe periodic in a time series. For example T = ab dacb ab aabc, symbol *'ab'* is periodic with p = 5 starting at position zero. In whole time series, a repetition of pattern or segment is called *segment or full-cycle periodicity*. For

Corresponding Author: Karthik, G.M., Department of Computer Science and Engineering, SACS MAVMM Engineering College, Madurai, Tamil Nadu, India Tel: +91-94432 85673 Fax: +91-422-2665138

example T = abdc abdc abdc has segment periodicity of p = 5 starting at position zero. Many existing algorithms (Elfeky *et al*., 2005a; 2005b; Han *et al*., 1998; Indyk *et al*., 2000) detects periods that span through entire time series. Some algorithms detect all the above mentioned three type of periodicity, along with noise within subsection of time series, separately for each patterns (Rasheed *et al*., 2011).

The traditional association periodic pattern mining problem is well defined and has been thoroughly studied in last decade (Elfeky *et al*., 2005a; Rasheed *et al*., 2011), there is currently no canonical way to measure the degree of correlation between periodic patterns (Huang and Chang, 2005). We believe that there should intuitively be more than one solution to define this new type of pattern, especially among different scenarios. Although answer to whether a periodic pattern is correlated or not is not an absolute, we at least expect to match common knowledge. An appropriate measure of correlation between long periodic patterns should be allowed to correlate with its sub-patterns.

The concept of Frequent Correlated Periodic Pattern mining (FCPP) used with time series data was handled efficiently in this study. The process was initiated with TRIE data structure referred as consensus tree that will enable parallel pattern search within the tree search path. It was followed by period establishing position and finally results in time series and time correlated approach.

This study addresses the following:

- The novelty of pattern mining using Frequent Correlated Periodic Pattern (FCPP) was handled to the address the issues on frequent pattern tree path towards time series and time correlated approach
- In order to focus on its efficiency of the algorithm the periodicity was evaluated in trade off with CONV (Elfeky *et al*., 2005a) WARP (Elfeky *et al*., 2005b), ParPer (Han *et al*., 1999) and finally STRN (Rasheed *et al*., 2011). The result obtained shows scalable performance in single stretch
- To mine item pairs of a particular node, represent a periodic pattern and determine the correlated relationship among item pairs. We select measures appropriate to our mining task
- To demonstrate the outstanding performance of our algorithm based on correlated relationship in terms of both efficiency and effectiveness on datasets

The literature review was elaborated in section 2 and section 3 with initial ground work, section 4 results in correlated time series data and its approach, section 5 with algorithm and followed by Section 6 with its pros and cons.

2. RELATED WORKS

The time series query based approach and its classification based on the given querying sequences was addressed in (Fu *et al*., 2005; Vlachos *et al*., 2002; Zhu and Shasha, 2003). In trade off the algorithm that exist need to provide the specific time period (Han *et al*., 1998; Ma and Hellerstein, 2001; Yang *et al*., 2002; Berberidis *et al*., 2002; Chen *et al*., 2006) for getting the time series result and the time series trend set up was discussed in (Udechukwu *et al*., 2004) and the range was addressed in (Elfeky *et al*., 2005a; Indyk *et al*., 2000; Rasheed *et al*., 2011). The noise suppression ratio in time series data was addressed in (Elfeky *et al*., 2005a) where it fails to do so. In order to detect segment in periodicity and its sequence the concept of WARP (Elfeky *et al*., 2005b) was introduced. To detect the time series and its periodicity Sheng *et al*. (2005b) proposes an algorithm to retrieve the said data. The combination all these algorithms (Elfeky *et al*., 2005a; 2005b; Han *et al*., 1999) retrieves time series data along with periodicity based on its range. The time series sub section was addressed in (Rasheed *et al*., 2011) using STNR algorithm. The proposed algorithm (Mueen *et al*., 2010) results in time series followed by its time correlated approach. The time series and its correlated check in STNR prolongs for its entire pattern was also proposed.

The study of correlation pattern mining focused on two important aspects. The first aspect is the significance of the patterns. More specifically, it is relevant to provide significance measures for the correlation of attribute sets and the correlation patterns. The second aspect is related to the computational cost of the proposed task. FP-growth mining algorithm (Han and Kamber, 2006), offers the better performance in mining null transactions for subsequent scanning of conditional databases. Omiecinski (2003; Kim *et al*., 2004) which used to find correlated patterns satisfying given minimum all-confidence. Liu *et al*. (1999) used the method of pruning by discovering the time correlation using contingency table. The concept of independent and correlated pattern was addressed to get the exact time correlated data from the time series data’s was handled by (Zhou, 2008; Zhou *et al*., 2006). The mining periodicity in compared with transaction data requires unique identity. Nevertheless these works built obviously by scanning every item pairs in a particular node of the consensus tree.

3. PRELIMIARIES

3.1. For Mining Periodic Pattern

Suppose ∑ is a finite symbol set and |Σ| its cardinality. Our previous work reflects the following

(Pujeri and Karthik, 2012). For DNA, |Σ| is 4 and the symbols are the 20 amino acids. Let S = {S_1,S_2,…,S_N} of input time series sequences over a finite symbol set ∑ with |Σ| = *R*, such that |s_i| = *L*, 1≤i≤*N* and positive integer *d* and *q* such that 0≤*d*≤*L* and 1≤*q*≤*N*. Here given parameters *N* and *L* are the number and length of given input sequence. Let *a* is called a pattern (center string) if each of at least *q* input sequence contains a substring in *a*'s *d*-neighborhood. Find all center string t∊Σ^l with any length *l*, 0≤*d*<l≤*L* every t has at least q sequence posse's *x-mutated* copy (*x*≤*d*) of *t*. In real time, we have to investigate time series to identify repeating patterns along with its outliers. The proposed work concentrates on manipulated data that received as a result of time series patterns for exploring further patterns along with correlated approaches and it outliers. These outliers will further detect other patterns along with its sequence and its periodicity. The output of this approach is the TRIE data structure (referred as consensus tree) that helps to explore the patterns received as the result of the proposed approach.

The level of confidence to acquiring further pattern was done in two ways. First the pattern position, its sequence and its periodicity(as mentioned earlier) that result in initial pattern sequence to start up with followed by the level of confidence received as a result of exploring patterns. The level of the patterns and its periodicity makes the initial point of access to explore patterns of its kind as discussed in (Rasheed *et al*., 2011).

3.2. For correlated Patterns

A correlation provides results by considering the time series data as input and to detect the time interval of the series.

The concept of periodic mining assists in similar patterns gathering and also provides the way to recognize it. In order to do the comparison, the different signal phases are taken and reflected (autocorrelation) to its copy. The repeating periodic signals are then captured and analyzed subsequently.

In statistical theory from literature (Zhou, 2008; Zhou *et al*., 2006), A_1, A_2,..,A_n are independent if ∀k and:

$$\forall(1 \le i_{1,2,\dots} \le \dots \le i_k \le n,$$
$$P(A_{i_1}, A_{i_2}, \dots, A_{i_k}) = (A_{i_1})P(A_{i_2})\dots P(A_{i_k}))$$

A resultant pattern has two data followed by its pattern A and B, then the approach (2) is applicable has not more than two data's or items and result in the approach (3) from the approach (1 and 2) then Equation 1:

$$\rho(AB) = \frac{P(AB) - P(A)P(B)}{P(AB) + P(A)P(B)} \tag{1}$$

If in the case of two data or items that sets the minimum and maximum patterns along with the pattern confidence level (threshold). The correlation terms either result in the combination of two data (dependent) on the other case results in two separate data's (independent) such as pattern *X* = {i_1, i_2,…,i_n}, then Equatin 2:

$$\rho(X) = \frac{P(i_1, i_2, ..i_\text{n}) - P(i_1), P(i_2), ..., P(i_n)}{P(i_1, i_2, ..., i_n) + P(i_1), P(i_2), ..., P(i_n)} \tag{2}$$

1 and 2, results in t ρ that has two bounds, i.e., -1≤*ρ*≤1. Let *δδ* be a given minimum correlated confidence, if pattern *X* has two data's A, B are called correlated with each other, else A and B are called independent (Karim *et al*., 2013). If pattern X has more than two items, we define a correlated pattern and an independent pattern as follows:

Definition 1

Correlated pattern x result in y then both the patterns are correlated in the case of depended patterns then *Y*⊆*X* and |*ρ* (*AB*)|>*δ*; where δ is a predefined value of *ρ*.

Definition 2

In the case of independent pattern, there exists a pattern x then no such patterns reflect on the same pattern (subsets).

Let *T* = {i_1,i_2,..,i_m} be a set of *m* distinct literals called items and *D* is the set of variable length transaction over *T*. The interestingness measure all-confidence denoted by *α* of a pattern *X* can de defined as follows Equation 3:

$$\alpha(X) = \frac{Sup(X)}{MAx\_item\_Sup(X)} \tag{3}$$

Definition 3

In the case of dependent pattern, there exists a pattern x and y, where the confidence level is either maximum or equal to its value. Such patterns focus on its associated pattern.

Definition 4

Associated-correlated pattern-In case of associative pattern, there exist a association between pattern between two subsets of A and B.

4. FCPP MINING ALGORITHM

4.1. FP Tree Construct

The mapping of sub patterns along with the time series data and its path were denoted as `t' as pattern time series and `s' as its sequence. It starts from the root node'n' and it is mapped with the sub pattern string of (j,k,e) with pointers starting from the k^{th} position of sequence j and provides the results in terms of its pattern. The tree is fully balanced for the node that has balance pointers connecting its descends. The concept of backward closure property (Karthik and Pujeri, 2013) makes pre-pruning in connection with the constraints levels for every pointer linked with the sub pattern. The Frequent Correlated Periodic Pattern mining (FCPP) categorized based on constraints dealing with antimonotone, monotonic and succinct constraints which is also addressed in our previous work (Karthik and Pujeri, 2013). Nodes with confidence value as conf(b) = $((N\text{-}sup(b)))/((N\text{-}q)<1)$ will be pruned; it is used as an antimonotonic constraint and a node in the consensus tree will branch out only if a support value is $\le q$ which is used as a monotonic constraint (Lee and Raedt, 2004). Each pointer in a consensus node has to satisfy degree of mutation $e>d$, otherwise it will be also pruned which sustains all position in consensus node like succinct constraint (Lee and Raedt, 2004). For each pointers without the mutation level $e>d$ will not participate in production of pointers in next consensus node.

Algorithm1. FP-Tree Construction

```
For each string $j$ of given input sequence $N$ do
For each symbol $k$ of input string $j$ of lenght $L$ do
If the $k$th symbol $i$th sequence is $b_1 \in$ do
Put (j,k in new node $S_{b_1}$, find (j,k substring is in all $S_{b_1}$ for $b_1 \neq b_2$ and $j$ in $T_{b_1}$ for each $b_1 \in \Sigma$ if and only if sup $(b_i)$> threshold.
For each $i$th sequence from 1 to    do
Loop(1):
For each substring's conf $(b_1,b_2,b_3,\dots,b_{i+1}) \le$do
Loop(2):
For each entry (j,k,e) in each nodes $S_{b_1},b_2,\dots,b_i$ where $k<L-i+1$ do
Loop(3):
If the (j,k,e) th element of the $j$th sequence is $b_{i+1} \in \Sigma$ and sup $(b_{i+1})$<q do
Begin(1):
put (j,k,e) in $S_{b_1,b_2,\dots b_i,b_{i+1}}$;
if e<d then for all $b_{i+1} \neq b_{i+1}$
put (j,k,e+1) in $S_{b_1,b_2,\dots b_i,b_{i+1}}$ if and only if conf $(b_1,b_2,b_3,\dots,b_{i+1}) \ge 1$;
End Begin 4;
If conf $(b_{i+1})<1$ then Remove $S_{b_{i+1}}$;
End Loop 3;
For each node $S_{b_1,b_2,\dots,b_{i+1}} \neq j$ do
For each node in next level $S_{b_1,b_2}\dots,b_i$ with distance $(b_i, b_i) \le$d do
For each $S_{b_1,b_2,\dots,b_i} \neq \phi$ and $conf(S_{b_1,b_2,\dots,bi}) \ge q$ along with distance $(b_i, bi) \le$d do
Loop(5):
If conf $(b_i)<1$ then Remove $S_{b_i}$
Create a new level in consensus tree with $T_{b_1,b_2,\dots,b_i} \leftarrow T_{b_1,b_2,\dots,b_i} \cup S_{b_1,b_2,\dots,b_i}$
If no node exists in $T_{b_1,b_2,\dots,b_i}$ then
Increment $i$ ; End Loop2;
Else
Print the output sequence $(b_i, S_{b_i})$ ;
End Loop 5;
If all $S_{b_1,b_2,\dots b_i,b_{i+1}}$ are removed then stop the program else output all pairs $(b_1,b_2,\dots b_{i+1};S_{b_1,b_2,\dots,b_i,b_{i+1}})$
Remove all $S_{b_1,b_2,\dots,b_i}$ and $T_{b_1,b_2,\dots,b_i}$ ;
End Loop 2;
i = i+1;
End Loop 1;
```

4.2. Periodicity Detection Algorithm

The usage of consensus tree provides sufficient data for identifying the periodicity of time series database. The concept of linear distance was applied for estimating the distance between two sub pattern that creates distance vector and represents it in matrix format. The **Fig. 1** shows the results of distance vector with its subsequent starting and end position and also estimates the possible repetition of the sub pattern occurrence with respect to the consensus tree structure. It also maximizes the occurrence frequency based on the frequency count. The Starting Position and EndingPostion of the subpatterns along with its occurrence frequency was well recognized using FCPP algorithm. As a result the algorithm takes three parameters for consideration (starting position, ending position and its frequency)

4.3. Algorithm 2. Difference Matrix (Diff-matrix) Algorithm

Input:

Starting position pointer for time series data with its position

Output:

Difference vector of A

```
For $i$ = 1 to $N$-1
Begin Loop 1:
    Assign $j$ = 1
    if ($j<N-i$)
        A ($j,i$) = $S_j-S_{j+i}$;
        if ($j+1 \neq j+i$)
        Then
            $t = j+1$;
            While ($t<j+i-1$)
            Begin Loop 2:
                A ($t,i$) = $S_t-S_{t+i}$;
                $t = t+i$;
            End Loop 2;
        Endif;
        $j = j+i$;
    Endif;
End Loop 1;
```

4.4. Finding Correlation in Periodic Patterns

We use Discrete Fourier Transform (DFT) to identify correlated item pairs in consensus node. The DFT of a item $x = x_0, x_1,\dots,x_{m-1}$ is a sequence $X = X_0, X_1,\dots,X_{m-1}$ = DFT(x) of complex numbers given by $X_f = \frac{1}{m}\sum_{k=0}^{m-1} x_i e \frac{-2\pi if}{m} k$ f = 0, 1.., m-1. We also define the normalization of x as $x = x_0, x_1,\dots,x_{m-1}$ such that $x_k = (x_i-\mu_x)/\sigma_x$ are mean and standard deviation of the values $x = x_0,x_1\dots,x_{m-1}$. The correlation coefficient of two item x and y can be reduced to the Euclidean distance between their normalized series such as $corr(x,y) = 1 - \frac{1}{2m} d^2(x,y)$, where $d(x',y')$ is the Euclidean distance between x' and y'. By reducing the correlation coefficient to Euclidean distance, we can apply the technique (Zhu and Shasha, 2002) to report the correlation between the item pairs exists in consensus node which is higher than a specific threshold. Few item pairs can be ignored for which $d_k(X,Y) < \sqrt{2m(1-T)}$, since they cannot have correlation above a given threshold T. By ignoring such pairs, we will get a set of likely correlated signal pairs. Conceptually, the algorithm produces a matrix like one shown in **Fig. 2**, where all pairs with correlation above a threshold and some pairs with correlation below the threshold are marked as 1 and all other pairs are marked as 0. We can call this a pruning matrix *P* and use it in subsequent steps.

In our technique, the pattern occurrence of item in a node is partition based on the capacity of cache. If the cache does not fit with all instance of the node, we need to partition the instance of the node. Thus, computing correlation between signals in different batches incurs additional costs. Hence we chose existing algorithm F-M partitioning algorithm for partitioning instances of a node into equal size.

Consider a discretized sequence {a b a b a} for an interval range of 2. The 2*5 Matrix M produced for the input sequence is given as follows:

$$m = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \end{bmatrix}$$

In this matrix, the first row represents symbol ‘a’ and the second row represents symbol ‘b’. The application of autocorrelation on each of the rows separately will produce the below result:

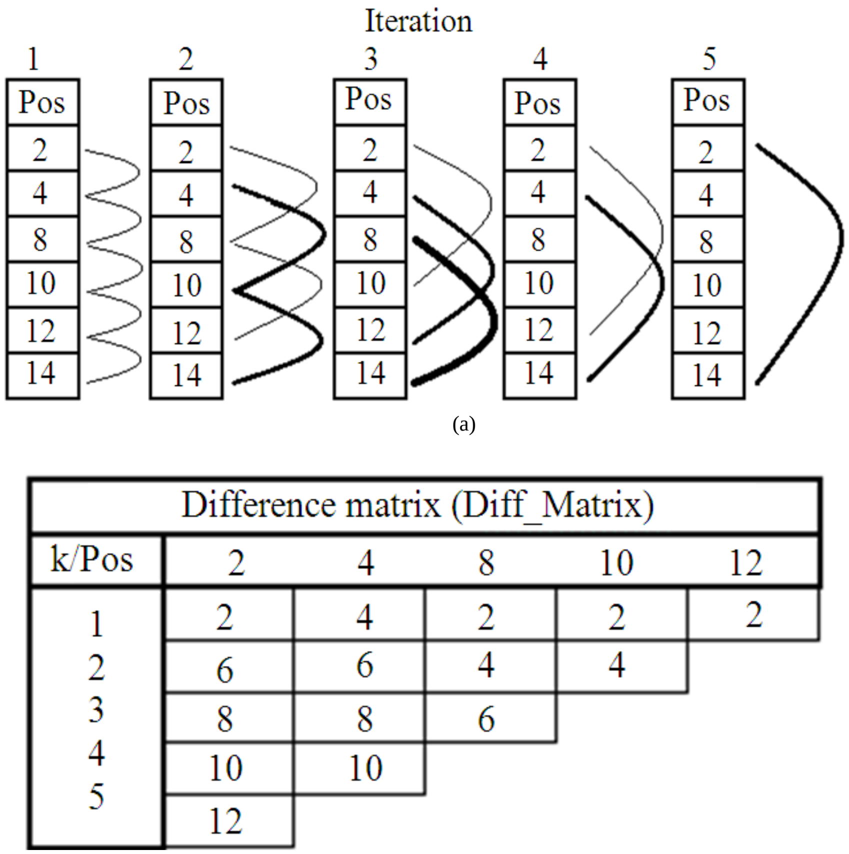
$$R = \begin{bmatrix} 3 & 0 & 2 & 0 & 1 \\ 0 & 2 & 0 & 1 & 0 \end{bmatrix}$$

In the correlated output R, every non-zero element represents the total number of occurrences of the symbol starting from that position. In that, the first element represents the total number of occurrences of the symbol. In this example, the output 3 in the first row represents the total number of occurrence of symbol a and 2 in the next row represents the total number of occurrence of symbol „b. The index positions of the non-zero elements are derived from the matrix. From that index position, the perfect and imperfect periodic rates are computed. In this example symbols a and b has occurred with a perfect periodic rate of 1. Every non-zero element of the row is auto correlated with the adjacent element of every other row until a zero value or end of the series is reached. The formula used is as follows:

$$R_{xx}(j) = \sum x_n x'_{n-j}$$

Where:

R = The discrete autocorrelation
J = The lag for a discrete signal x_n



(a)

Difference matrix (Diff_Matrix)					
k/Pos	2	4	8	10	12
1	2	4	2	2	2
2	6	6	4	4	
3	8	8	6		
4	10	10			
5	12				

(b)

Fig. 1. Time series pattern from FP tree node pointers

	1	2	3	4	5	6	7	8
1	■			■				■
2					■		■	
3			■			■		
4		■					■	
5					■			
6	■			■		■		■
7		■			■		■	
8			■			■		

Fig. 2. Computing a threshold correlation matrix

4.5. Experimental Evaluations

We tested our algorithm based on our previous work that gathers information based on time series approach (Pujeri and Karthik, 2012) over a number of data sets. For real data experiments, we used supermarket data which contains sanitized data of timed sales transactions for Wal-Mart stores over a period of 15 months. Synthetic data taken from Machine Learning Repository (Blake and Merz, 1998) were also used. We tested how *FCPP* satisfies this on both synthetic and real data. The algorithm can find all periodic patterns 100% along with their correlation coefficient. This is an important feature in using FP tree which guarantees identifying all repeating patterns.

In order to test the accuracy, we test the algorithm for various period sizes, distribution and time series length. We used synthetic data obtained from Machine Learning Repository (Blake and Merz, 1998), have been generated in the same done in (Elfeky *et al*., 2005a). **Figure 5a** shows the behavior of the algorithm against the number of the time points in the time series. **Figure 5b** shows that the algorithm speeds up linearly to symbol set |∑| of different size. *FCPP* checks the periodicity for all periods within synthetic data in absence of noise.

For real data experiments, we used the Wal-Mart data which contains hourly based records of all transaction performed at a Supermarket. The data contains the record of around 15 months of data with expected period value of 24. *FCPP* algorithm with periodicity threshold values ranging from 0.8 to 0.4 and observed: The number of periods captured by algorithm, *StPos* and *EndPos* of the sequence, confidence value and the Pattern shown in **Table 1**. The expected period 24 is captured at the threshold value 0.8. Periodic pattern obtained less in number but accurate, useful and meaningful. **Table 1** demonstrates that how periodic pattern are obtained without redundant period. *FCPP* algorithm does not produce duplicated period due to the presence of supper-pattern (Karthik and Pujeri, 2013) which holds the information of gathered patterns using Diff-matrix.

The result of time correlation shown in the **Table 1** based on the wall mart data analysis. As the data grows enormously the efficient growth analysis of time series data using frequent pattern mining deviates as shown in **Fig. 3 and 4** based on time factors. In also duplicates as there is change in data and its volume. To address this issue the concept time correlation factor is introduced using pearson correlation coefficient is define as Equatoin 4:

$$corr(x,y) = \frac{1}{m}\sum_{i=0}^{m-1}\left(\frac{x_i - \mu_x}{\sigma x}\right)\left(\frac{y_i - \mu_y}{\sigma y}\right) \quad (4)$$

In order to identify the exact factors and pairs used in correlation factor was identified by x the data received from **Table 1** and its sequence is followed by the variable y Equatoin5:

$$corr(x,y) = \frac{1}{m}\sum_{i=0}^{m-1} x_i\, e\left(\frac{-2\pi\,(y_i - \mu_y)}{\sigma_y}\right) \quad (5)$$

The normalization is estimated based on the parameter score of x and y is achieved, results in the correlation value as per Equation 5.

The correlation sequence is estimated for avoiding the repeated and duplicated index of the factor x and subsequently takes the y as a sequence factor and its further date analysis.

The results were discussed in the **Table 1** and shown in the time correlated column. Such factor affects the threshold value based on time series and time correlated value.

The frequent constraint algorithm does not allow duplicate entry information based on the data registered in Diff-Matrix. The representation of FCPP and Par Per algorithm and its impact over time series was shown in **Fig. 3 and 4**. **Figure 5** represents the performance analyzes of time series and its behavior over a period of time was shown. The performance comparison of FCPP and ParPer was checked with the data size between 1 to 10 lac. As the result the FCPP shown better performance compared with WARP and STNR and projects less impact compared with CONV based on runtime. FCPP maximizes its performance over a period of time as there is persistent progress in data and period size were such combination affects the performance of ParPer (Han *et al*., 1999) and WARP (Elfeky *et al*., 2005b) in terms of its data size.

4.6. Estimating Time Correlation Based on Time Series

Step 1

Testing for a unit root in StPos
Augmented Dickey-Fuller test for StPos
including one lag of (1-L)StPos (max was 1, criterion modified AIC)
sample size 13
unit-root null hypothesis: a = 1
with constant and quadratic trend
model: (1-L)y = b0 + b1*t + b2*t^2 + (a-1)*y(-1) + ... + e
1st-order autocorrelation coeff. for e: -0.212
estimated value of (a - 1): -1.20505
test statistic: tau_ctt(1) = -1.9731
asymptotic p-value 0.8273

Step 2

Testing for a unit root in EndPos
Augmented Dickey-Fuller test for EndPos
including one lag of (1-L) EndPos
(max was 1, criterion modified AIC)
sample size 13
unit-root null hypothesis: A = 1
with constant and quadratic trend
model: (1-L)y = b0 + b1*t + b2*t^2 + (a-1)*y(-1) + ... + e
1st-order autocorrelation coeff. for e: -0.333
estimated value of (a - 1): -1.08833
test statistic: tau_ctt(1) = -2.26869
asymptotic p-value 0.6951

Figure 6 and 7 shows the algorithm performance in terms of threshold value with it periods also by considering the factors of starting and ending position with its occurrence.

Table 1. FCPP algorithm output for Wal-Mart data

	Periodicity	No. of					Time
Data	threshold	periods	StPos	EndPos	Conf.	Pattern	correlated
Store 1	0.8	4	109968	145081	0.42	AAA*******AAA******AA***	AAA****AAA
	0.7	9	134887	161412	0.40	AAABBBCCC************AA*	AABBCC**AA
	0.6	11	151141	194123	0.30	AABBBBCCCD*******AAAA***	AABBBBCCCD
	0.5	16	213476	263129	0.32	AAAABBCCD***AADD********	AABBCCDD*A
	0.4	25	234980	280673	0.40	AAA***AAAAA*********AAA*	AA*****BBAA
Store 2	0.8	6	180613	199457	0.44	AAA****BCCC*****DD******	AA**BCC**DD*
	0.7	7	164319	200312	0.47	AAB****AAABBBDDD****BB*	AB****AABBD*
	0.6	13	229846	273422	0.37	AAAABBB*******CC****DDD*	AABB**CC*DD*
	0.5	17	215978	286421	0.40	AAAAACCCC****BBBCC***DD	AACC*BBC*DD
	0.4	20	283149	304231	0.41	AAAAAACCC****BB*********	AACCC**BB***
Store 3	0.8	5	147030	155044	0.42	AA****BBB***BCDDD*******	AA*BB**BCDD*
	0.7	8	152783	167329	0.30	AAAABBBC*******CCCCD****	AABCC***D***
	0.6	12	182390	186064	0.46	AAAAAACCCCCD************	AAACCCD*****
	0.5	14	177389	216892	0.47	AAAAABBBBBCCCCD*********	AABBBCCD****
	0.4	27	258202	299582	0.49	AAAABBBBBC*******BCCDDD	AABBBCC**DD

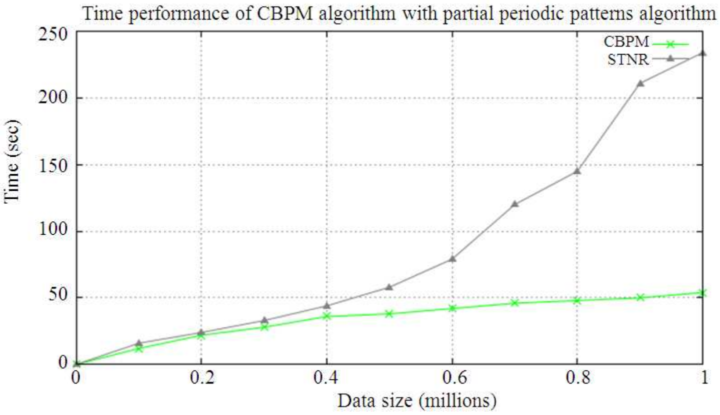


Fig. 3. Time performance of FCPP with ParPer algorithm

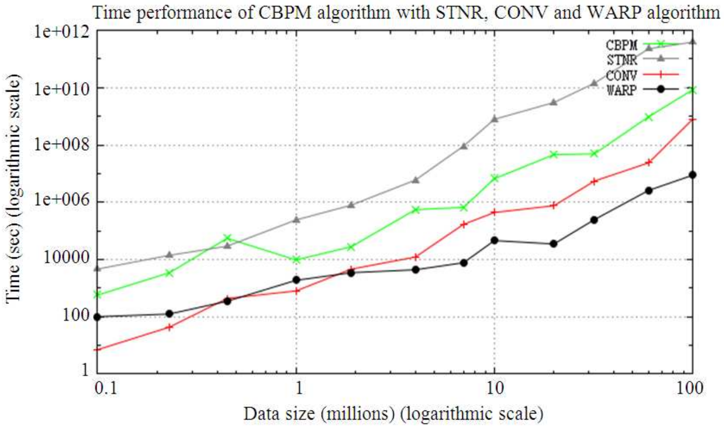
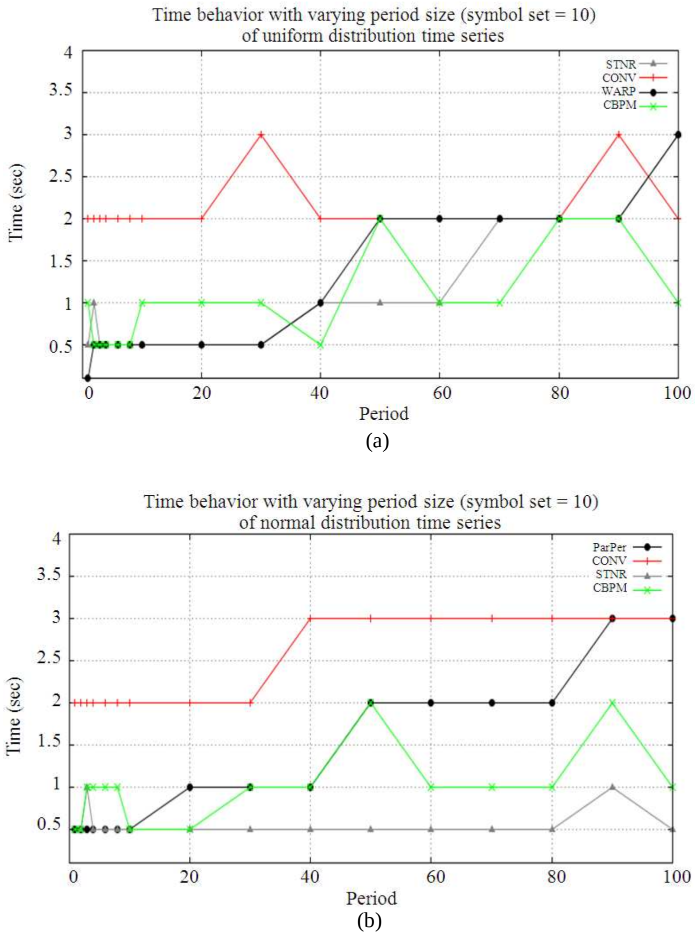


Fig. 4. Time performance of FCPP algorithm with STNR, CONV and WARP



(a)

(b)

Fig. 5. (a) (b): Time behavior with varying period size

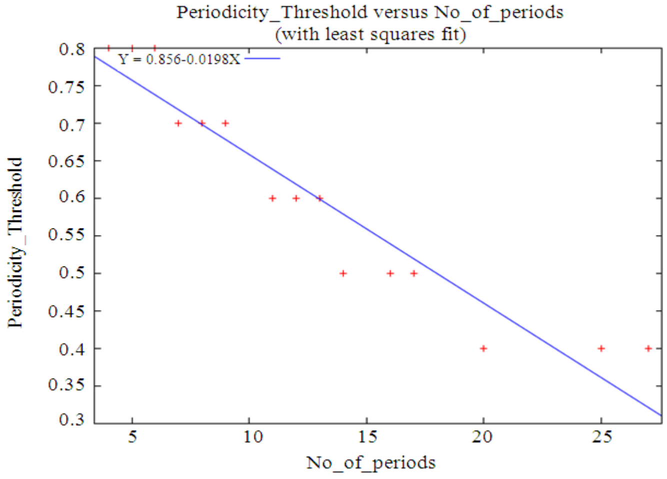


Fig. 6. Threshold vs periods

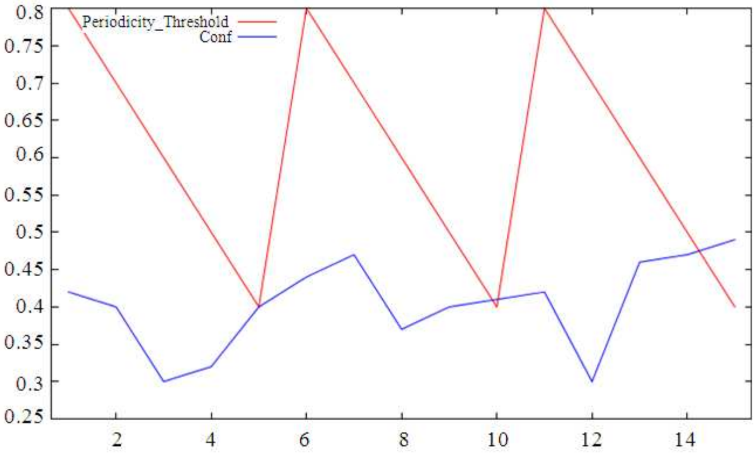


Fig. 7. Time series correlation

The performance measures of proposed algorithm proven to be effective in terms of its threshold value towards its time series correlation.

5. CONCLUSION

In this work, the proposed Frequent pattern growth algorithm mines large database which in turn addresses the issues dealing with time series and time correlated approach. The achieved threshold hold from time correlated approach proven to be effective compared with the value achieved using time series. The work also addresses the need for combining both time series and time correlated approach for providing better theroshold value with efficient feasible result that too in terms of large datasets.

6. REFERENCES

Berberidis, C., W.G. Aref, M. Atallah, I. Vlahavas and A.K. Elmagarmid *et al*., 2002. Multiple and partial periodicity mining in time series databases. Proceedings of the European Conference Artificial Intelligence, (CAI’ 02).

Blake, C.L. and C.J. Merz, 1998. UCI Repository of machine learning databases. University of Kassel, L3S Research Center, Germany.

Chen, C., C.X. Lin, X. Yan and J. Han, 2008. On effective presentation of graph patterns: A structural representative approach. Proceedings of the 17th ACM conference on Information and Knowledge Management, Oct. 26-30, ACM, Napa Valley, New York, pp: 299-308. DOI: 10.1145/1458082.1458124

Chen, F., J. Yuan and F. Yu, 2006. Finding periodicity in pseudo periodic time series and forecasting. Proceedings of the IEEE International Conference on Granular Computing, May 10-12, IEEE Xplore Press, pp: 534-537. DOI: 10.1109/GRC.2006.1635858

Elfeky, M.G., W.G. Aref and A.K. Elmagarmid, 2005a. Periodicity detection in time series databases. IEEE Trans. Knowl. Data Eng., 17: 875-887. DOI: 10.1109/TKDE.2005.114

Elfeky, M.G., W.G. Aref and A.K. Elmagarmid, 2005b. WARP: Time warping for periodicity detection. Proceedings of the 5th IEEE International Conference on Data Mining, Nov. 27-30, IEEE Xplore Press, Washington, USA, pp: 138-145. DOI: 10.1109/ICDM.2005.152

Fu, A.W.C., E.J. Keogh, L.Y.H. Lau, C.A. Ratanamahatana and R.C.W. Wong *et al*., 2005. Scaling and time warping in time series querying. VLDB J., 17: 899-921. DOI: 10.1007/s00778-006-0040-z

Han, J. and J. Pei, 2000. Mining frequent patterns by pattern-growth: Methodology and implications. ACM SIGKDD Exp. Newsletter, 2: 14-20. DOI: 10.1145/380995.381002

Han, J., L.V.S. Lakshmanan and T.N. Raymond, 1999. Constraint-based, multidimensional data mining. IEEE Comput., 32: 46-50. DOI: 10.1109/2.781634

Han, J., W. Gong and Y. Yin, 1998. Mining segment-wise periodic patterns in time related databases. Proceedings of the ACM International Conference Knowledge Discovery and Data Mining, pp: 214-218.

Han, J., Y. Yin and G. Dong, 1999. Efficient mining of partial periodic patterns in time series database. Proceedings of the 15th International Conference on Data Engineering, Mar. 23-26, IEEE Xplore Press, Sydney, NSW, pp: 106-115. DOI: 10.1109/ICDE.1999.754913

Han, J. and M. Kamber, 2006. Data Mining, Southeast Asia Edition: Concepts and Techniques. 2nd Edn., Morgan Kaufmann, Amsterdam, San Francisco, CA, ISBN-10: 0080475582, pp: 800.

Hasan, M.A., V. Chaoji, S. Salem and J. Besson, 2007. Origami: Mining representative orthogonal graph patterns. Proceedings of the Seventh IEEE International Conference on Data Mining, Oct. 28-31, IEEE Xplore Press, Omaha, NE, pp: 153-162. DOI: 10.1109/ICDM.2007.45

Huang, K.Y. and C.H. Chang, 2005. SMCA: A general model for mining asynchronous periodic patterns in temporal databases. IEEE Trans. Knowl. Data Eng., 17: 774-785. DOI: 10.1109/TKDE.2005.98

Indyk, P., N. Koudas and S. Muthukrishnan, 2000. Identifying representative trends in massive time series data sets using sketches. Proceedings of International Conference Very Large Data Bases, Sept. 10-14, ACM, Cairo, Egypt, pp: 363-372.

Karim, M., C.F. Ahmed, B.S. Jeong and H.J. Choi, 2013. An efficient distributed programming model for mining useful patterns in big datasets. IETE Tech. Rev., 30: 53-63. DOI: 10.4103/0256-4602.107340

Karthik, G.M. and R.V. Pujeri, 2013. Constraint based periodic pattern mining in multiple longest common subsequences. Indian J. Sci. Technol., 6: 5046-5057.

Kim, W.Y., Y.K. Lee and J. Han, 2004. CCMine: Efficient Mining of Confidence-Closed Correlated Patterns. In: Advances in Knowledge Discovery and Data Mining, Dai, H., R. Srikant and C. Zhang (Eds.), Springer Berlin Heidelberg, ISBN-10: 978-3-540-22064-0, pp: 569-579.

Lee, S.D. and L.D. Raedt, 2004. Constraint based mining of first order sequences in seqlog. Database Support for Data Mining Applications: Discovering Knowledge with Inductive Queries, Meo, R., P.L. Lanzi and M. Klemettinen (Eds.), Springer Berlin Heidelberg, ISBN-10: 978-3-540-22479-2, pp: 154-173.

Liu, B., W. Hsu and Y. Ma, 1999. Pruning and summarizing the discovered associations. Proceedings of the 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Aug. 15-18, ACM, San Diego, CA, pp: 125-134. DOI: 10.1145/312129.312216

Ma, S. and J.L. Hellerstein, 2001. Mining partially periodic event patterns with unknown periods. Proceedings of the 17th International Conference on Data Engineering, (CDE’ 01), IEEE Xplore Press, Heidelberg, pp: 205-214. DOI: 10.1109/ICDE.2001.914829

Mueen, A., S. Nath and J. Liu, 2010. Fast approximate correlation for massive time-series data. roceedings of the ACM SIGMOD International Conference on Management of data, Jun. 06-11, ACM, Indianapolis, IN, pp: 171-182. DOI: 10.1145/1807167.1807188

Omiecinski, E.R., 2003. Alternative interest measures for mining associations in databases. IEEE Trans. Knowl. Data Eng., 15: 57-69. DOI: 10.1109/TKDE.2003.1161582

Pei, J. and J. Han, 2002. Constrained frequent pattern mining: A pattern-growth view. ACM SIGKDD Exp. News., 4: 31-39. DOI: 10.1145/568574.568580

Pujeri, R.V. and G.M. Karthik, 2012. Constraint based periodicity mining in time series databases. Int. J. Comput. Netw. Inform. Sec., 10: 37-46.

Rasheed, F., M. Alshalalfa and R. Alhajj, 2011. Efficient periodicity mining in time series databases using suffix trees. IEEE Trans. Knowl. Data Eng., 23:79-94. DOI: 10.1109/TKDE.2010.76

Sheng, C., W. Hsu and M.L. Lee, 2005a. Efficient mining of dense periodic patterns in time series. Technical report, Nat'l Univ. of Singapore.

Sheng, C., W. Hsu and M.L. Lee, 2005b. Mining dense periodic patterns in time series data. Proceedings of the 22nd International Conference on Data Engineering, Apr. 03-07, IEEE Xplore Press, pp: 115-115. DOI: 10.1109/ICDE.2006.97

Udechukwu, A., K. Barker and R. Alhajj, 2004. Discovering all frequent trends in time series. Proceedings of the Winter International Synposium on Information and Communication Technologies, (ICT’ 04), ACM, Trinity College Dublin, pp: 1-6.

Versaci, M., 2014. Soft computing approach to predict intracranial pressure values. Am. J. Applied Sci., 11: 844-850. DOI: 10.3844/ajassp.2014.844.850

Vlachos, M., G. Kollios and D. Gunopulos, 2002. Discovering similar multidimensional trajectories. Proceedings of the 18th International Conference on Data Engineering, (CDE’ 02), IEEE Xplore Press, San Jose, CA, pp: 673-684. DOI: 10.1109/ICDE.2002.994784

Weigend, A.S. and N.A. Gershenfeld, 1994. Time Series Prediction: Forecasting the Future and Understanding the Past. 1st Edn., Eprint, Revised, Reading, Addison-Wesley, ISBN-10: 0201626020, pp: 643.

Yan, X. and J. Han, 2002. Gspan: Graph-based substructure pattern mining. Proceedings of the IEEE International Conference on Data Mining, (CDE' 02), IEEE Xplore Press, pp: 721-724. DOI: 10.1109/ICDM.2002.1184038

Yang, J., W. Wang and P.S. Yu, 2002. InfoMiner+: mining partial periodic patterns with gap penalties. Proceedings of the IEEE International Conference on Data Mining, (CDE' 02), IEEE Xplore Press, pp: 725-728. DOI: 10.1109/ICDM.2002.1184039

Zhou, Z., 2008. Mining frequent independent patterns and frequent correlated patterns synchronously. Proceedings of the 5th International Conference on Fuzzy Systems and Knowledge Discovery, Oct. 18-20, IEEE Xplore Press, Jinan Shandong, pp: 552-556. DOI: 10.1109/FSKD.2008.27

Zhou, Z., Z. Wu, C. Wang and Y. Feng, 2006. Mining Both Associated and Correlated Patterns. Computational Science-ICCS, In: Alexandrov, V.N., G.D.V. Albada, P.M.A. Sloot and J. Dongarra (Eds.), Springer Berlin Heidelberg, ISBN-10: 978-3-540-34385-1, pp: 468-475.

Zhu, Y. and D. Shasha, 2002. StatStream: Statistical monitoring of thousands of data streams in real time. Proceedings of the 28th International Conference on Very Large Data Bases, (LDB' 02), ACM, pp: 358-369.

Zhu, Y. and D. Shasha, 2003. Warping indexes with envelope transforms for query by humming. Proceedings of the ACM SIGMOD International Conference on Management of Data, Jun. 09-12, San Diego, CA, pp: 181-192. DOI: 10.1145/872757.872780