OPEN ACCESS INTERNATIONAL JOURNAL OF SCIENCE & ENGINEERING

REDEFINING TRANSITIONAL PATTERNS FOR VALID ON DYNAMIC DATABASES

Somaraju Suvvari[1], RBV Subramanyam[2]

Assistant Professor, Department of CSE, TKR College of Engineering & Technology, Hyderabad, India[1]

Associate Professor, Department of CSE, NIT Warangal, Warangal, India[2]

soma2402@gmail.com[1], rbvs66@gmailcom[2]

Abstract: **The researchers have focused on extracting time included knowledge that reveals the behavior of item sets, such as finding the patterns that are more present on a specific time period; finding the specific time points, where the frequency of an item set before or after a time point increases or decreases significantly, etc. Many previous works also consider the time points in Frequent Pattern Mining (FPM) studies and presented temporal mining algorithms (TPM). All these TPM based works included time as an element, but nowhere in these works how dynamically is a frequent pattern P changing its behavior in the database D. The authors Wan and An presented "Transitional Patterns" which represent item sets whose support significantly changes from one time point to another time point in the database. Primarily Wan and a focused on finding time points at which negative (or positive) transitional patterns decreases (or increases) their support significantly with the change of time. TP-Mine algorithm has the limitation of when we add new transactions to the database and by reapplying the TP-Mine algorithm on the updated database, already identified time points may not valid on the updated database. The motive of this paper is to address the limitation existing in the TP-Mine algorithm proposed by Wan and An and it is done by redefining the definition of transitional pattern and proposing an efficient two scan algorithm for finding redefined transitional patterns in parallel to the extraction of frequent patterns.**

Keywords *– Data mining, Association rule, frequent pattern, Alternative Transitional Patterns, Major and Minor Landmarks.*

I INTRODUCTION

Mining of Frequent Patterns (FPM) is essential prerequisite to form the knowledge from Data Mining step in Knowledge Discovery in Databases (KDD) process. In Association Rule Mining, patterns are referred with item sets. Other kinds of patterns are used in various Data Mining algorithms which work on sequence databases, spatial databases, and time series databases. Apriori c kind of algorithms Apriori kind algorithms [2] [3] [17] [18] [19] uses the principle of level wise candidate generation and pruning of un promised candidate item sets, Tree based algorithms [8] [9] have motivated significant number of researchers to contribute fast methods for finding frequent item sets from databases.

The hidden agenda of finding frequent item sets present in a database is to know about the inherently hidden knowledge in the database and it aids in taking better decisions which drive for business improvement. It is noticed that frequent patterns need not uniformly present on the overall transactional database, rather its impact on some partition of the database may be enough and is motivated researchers to present new approaches to perform low level analysis on the database for knowing about the hidden information. In recent times, the researchers have focused on extracting time included knowledge that reveals the behavior of item sets, such as finding the patterns that are more present on a specific time period; finding the specific time points, where the frequency of the item set before or after a time point changes significantly, etc. Time-points of the database are normally ignored in extracting the item sets related to Association Rule Mining.

Wan, et.al. [20] Presented "Transitional Patterns" reveals the dynamic nature of the frequent item sets (FI). There are many applications of these new patterns include changing strategies in marketing by analyzing the dynamic behavior of the item sets in retail environments, finding the time stamps at which drugs causes the side effects in medical environment, in web mining environment web links can be re arranged based on support of visiting the web links, identifying the time points where profit increases/decreases in the stock market, etc. TP-Mine [20] runs in two phases and uses three scans for finding the transitional patterns. In the first phase, it uses two scans and finds the frequent item sets (by using FP-Growth algorithm [9]) and in second phase, by using another scan on the database identifies transitional patterns There may exist many time points exhibit the behavior of transitional patterns, the TP-Mine algorithm records only those time points where the transitional ratio is maximum among the transactions which exhibit the behavior of transitional patterns. Each such time point gives the knowledge about the behavior of the pattern, such as after the time point the patterns significance may increase or decreases drastically.

TP-Mine [20] algorithm and FTP-Mine algorithm [22] suffers in maintaining the milestones and transitions in case of incremental databases. Suppose ODB is a database and NDB is a new database of transactions to be appended to ODB to get UDB. The anticipated behavior is that a transitional pattern's milestones in UDB should agree with milestones found in ODB part and some more may be found in NDB part. TP-Mine algorithm fails to exhibit this anticipated behavior and transitions found earlier changing their behavior.

For example, in a stock market for a product X the time point 15-Aug-2014 identified as a decreasing time point with significance “30%”, indicates it loses its popularity 30% after the time point 15-aug-2014, by adding new transactions to the database the significance of X at the same mile stone must stay as a decreasing time point but its decreasing significance may changes.

Once a pattern was identified as a transitional pattern at some time point *t* and after updating the database with new transactions and by re applying the algorithm, the pattern's significance should not change at the milestones which were already recorded as an ascending or descending milestones, but there may evolve new time points to the pattern where the significance of the pattern may increases or decreases than its significance at its old time points t_1, t_2, ...t_n. In this paper we address the limitation present in the TP-Mine algorithm by redefining the definition of transitional pattern and an efficient algorithm is proposed for extracting the redefined transitional patterns in parallel to the extraction of frequent patterns. Our approach can also be applicable to the dynamic databases with minimal modification.

II RELATED WORK

In recent times, there is a more focus on finding knowledge which includes the time related information, such as such as finding the item sets that are more present on a specific time period; finding the specific time points, where the frequency of an item set changes before or after that time point significantly, etc. Traditional frequent pattern mining works also consider the time points in the databases while finding the frequent item sets, such as patterns extracted from sequence database [1] [12] [21] and extracting support threshold satisfied episodes [13], mining association rules present in the time included databases.

Marko and Mannila [14] proposed a solution for finding support threshold satisfied episodes in sequence databases was proposed. Mannila et al. 1997 [13], proposed a method to extract all support threshold satisfied episodes in an event sequence. The work proposed in this paper is different from the episode mining, where we study the in-depth analysis on each frequent pattern, rather than considering the relationship between events or item sets.

Temporal association rule (TAR) is different from standard association rule, standard association rules hold over the entire database, whereas TAR hold only at specific time periods. There are various types of TARs [16] [10], including cyclic association rules (CAR) and these association rules present I the database regularly [15], calendar type of temporal association rules [11], and TARs where the lifespan of items in TARs is the time period from its first occurrence to the last occurrence in the database [4], Gharib and et.al [7] presented an efficient approach for incremental mining of TARs, where it gets the advantage of the existing results to extract the TARs on the updated database.

Ozden, Ramaswamy and Silberschatz [15] observed the problem of CARs, where each CAR presents all the cycles. Li, Ning, Sean and Jajodia [11] presented an approach for finding the different kind of TAR and it needs less prior knowledge compared to the existing approaches. Li, et.al, presented two types of TARs, TAR with respective to full/relaxed match.

This paper extracts the time points of frequent item sets where the change in support from the beginning of the database with respective minimum support increases or decreases more than some user specified threshold value. Such a collection of time points (maximum possible number of time points are number of partitions in the database) related to a frequent item set can be used for analyzing the cyclic nature of the item set in the database.

Bashar Saleh and Florent Masseglia [5] presented an approach for finding the time intervals where patterns are frequent in the specified time intervals only. Our work is different from the work of Bashar and Masseglia [5], in our work the extracted patterns must be frequent in the entire database, in [5] the item sets not required to be frequent on the entire database and is sufficient to be frequent only some part of the database as a result the extracted patterns are huge in number.

TP-Mine algorithm limitation

Now we present the TP-Mine algorithm [20] using an example, the limitations in the algorithm are explored. As outlined in the introduction, the transitional pattern mining algorithm [20] extracts the significant mile stones of identified transitional patterns the following definitions are defined in Wan and An [20] and also in [22].

Definition 1. (Transitional Ratio ($tran^i(X)$) It is the ratio to the difference between the support it has accumulated after the i^{th} transaction to the end of the database (assume it is $s_+^i(X)$) and the support it has accumulated till the i^{th} transaction from the beginning of the database to the maximum one of $s_+^i(X)$ and $s_-^i(X)$ (i.e. $(s_+^i(X)-s^i(X))/(maximum(s_+^i(X),s_-^i(X)))$.

Definition 2. (Transitional Pattern (TP)) Any frequent item set is called transitional pattern in the given database if the transitional ratio of the item set is more than the user specified threshold $|\ (tran^k(X)\ | \geq t_{t,)}$.

Definition 3. (SFAM (SFDM)) Significant Frequency Ascending (Descending) Milestones are the transactions where the transitional ratio is more (less) than the transitional ratio at other transactions in the database.

In this section we explore the limitation of TP-Mine algorithm [20] using an example database. The database is in table 1, and the minimum support is 40%, T_ξ=[10%, 90%] (any pattern is qualified as a transitional pattern at i^{th} transaction if and only if the transaction should not present in the first or last 10% of the transactions), t_s=0.4 ($|s_+^i(X)|$ and $|s_-^i(X)|$ must be greater than t_s), and t_t=0.3 then FIs in D are {abc, bc, ad, ac, ab, d, c, b, a},the transitional patterns identified are {bc, ad, ac, c, b, a}. The patterns {c, ac, bc} are the only negative transitional patterns, the pattern {ad} is the positive transitional pattern and the patterns {a, b, ab} belongs to both the categories.

In table 3 the summary of all the transitional patterns present in table 1. The pattern a has a significant descending milestone at the end of the ODB and the significant ascending milestone at the beginning of the ODB, while the pattern b has significant descending milestone at the beginning of the ODB and significant ascending milestone in the middle of the ODB.

Table 2 shows the added transactions (NDB) to the example database shown in table 1. Table 4 shows the summary of transitional patterns in UDB (ODB+NDB) by reapplying the Wan and An [20] approach with the same thresholds used in finding the transitional patterns in ODB. The item set has negative transitional ratio at 14^{th} transaction in ODB but in UDB it was not observed at the 14^{th} transaction, the item set d also exhibited same behavior at 8^{th} transaction. The pattern abc was not qualified as transitional pattern in ODB but it was qualified as both positive and negative transitional pattern at 10^{th} and 2^{nd} transactions.

Due to the fact that TP-Mine algorithm uses the support it collected till i^{th} transaction and the support after i^{th} transaction, as and when new transactions are added or deleted from the database the transitional ratio changes and changes its status from transitional pattern to non-transitional pattern or from non-transitional pattern to transitional pattern. We redefined the transitional ratio in such a way that it depends only on the support accumulated till the i^{th} time point only.

III PROPOSED APPROACH

In this section we are redefining the transitional patterns and named these patterns are "Alternative Transitional Patterns". We assume that each transaction in a transaction database has transaction ID (TID), items in the transaction and the time stamp of the transaction. In order to define the formal definition of alternative transitional pattern we first introduce the following definitions.

Table 1 Transactional Database (ODB)

TID	Items in transaction	Time points
001	*a, b, c, d, e, f*	*Dec- 2009*
002	*a, b, c*	*Jan-2010*
003	*b, c, d*	*Mar- 2010*
004	*c, g, h, I, j*	*Apr-2010*
005	*f, g, h, i*	*May- 2010*
006	*a, d, f, g, h, i*	*Jun- 2010*
007	*a, c ,d*	*Aug- 2010*
008	*a, b, c, d*	*Oct-2010*
009	*a, b, f, g, h, j*	*Dec- 2010*
010	*a, b, c*	*Jan-2011*
011	*a, b, c, d, e*	*Feb- 2011*
012	*a, b, c, e, f*	*Apr-2011*
013	*a, b, c, d, e*	*May- 2011*
014	*a, b, d*	*Jun-2011*
015	*a, d*	*Sep-2011*
016	*f, g, h, i*	*Oct- 2011*

Table 2 Updated Database (NDB)

TID	Items in transaction	Time points
017	*a, b, c, d*	*Nov-2011*
018	*a, b, c*	*Dec-2011*
019	*a, b*	*Jan-2012*
020	*a, b, c*	*Feb-2012*

A. Definition of Landmark

Definition 4. (Landmark of item set X (LM(X))). The presence of each item set in a transaction treated as its landmark. The i^{th} landmark of an item set in D denotes the i^{th}

occurrence of X from the beginning of the database D. The landmark is analogous to milestone in TP-Mine algorithm.

Definition 5. The position of i^{th} occurrence of X in D denoted as $P^i(X, D)$, it gives the TID of the transaction where the i^{th} occurrence of X from the beginning of the database.

Definition 6. The frequency of an item set from the beginning of the database to its i^{th} land mark in the database D is denoted as $Sup^i(X, D)$ is the support accumulated from the beginning of the database to $P^i(X, D)$.

In TP-Mine algorithm the old significant milestones (the significant milestones mined before updating the database) of any item set changes because the transitional ratio depends on the support it has after the milestone to the end of the database. To overcome these limitations we modified the transitional ratio in terms of minimum support and are defined as follows.

Table 3 Transitional Patterns in ODB

Pattern (P)	**Type of TP**	**$Tran^i(P)$**	**SFAM(P) / SFDM(P)**
a	+Ve	0.45	6
A	-Ve	-0.37	14
B	+Ve	0.34	8
B	-Ve	-0.46	3
c	-Ve	-0.5	4
ab	+Ve	0.38	9
ab	-Ve	-0.5	2
ac	-Ve	-0.57	2
ad	+Ve	0.31	13
bc	-Ve	-0.57	2

Table 4 Transitional Patterns in UDB

Pattern (P)	**Type of TP**	**$Tran^i(P)$**	**SFAM(P) / SFDM(P)**
a	+Ve	0.46	6
b	+Ve	0.4	8
b	-Ve	-0.36	3
c	-Ve	-0.44	13
d	-Ve	-0.34	8
ab	+Ve	0.45	9
ab	-Ve	-0.39	2
ac	-Ve	-0.5	2
ac	+ve	0.30	7
bc	-Ve	-0.53	3
abc	+Ve	0.34	10
abc	-Ve	-0.56	2

Definition 7. (Transitional Ratio) The transitional ratio at i^{th} land mark of any item set X in D which is in the range of $T_ʒ$, denoted as $Tran^i(X, D)$, is defined as,

$Tran^i(X, D) = ((Sup^i(X, D) - s) / \text{Maximum}(Sup^i(X, D), (1-Sup^i(X, D)))$.

Note: Maximum (a, b) returns the maximum value between a and *b*.

The transitional ratio at any landmark *i* of an item set X signifies the significant change with respective minimum support.

B. Discussion on Range of $T_ʒ$

Any item set X has which more support in the beginning of the transactions in a database D has a transitional ratio near to one and any item set which does not have any support in the earlier part of the database D has a transitional ratio near to -1 and it shows only the sporadic nature of the database and it will not reveal the real nature of the database. It is necessary to skip some transactions from the beginning of the database. If the number of transactions to skip at the beginning of the database is too less or too more then it may not reveal the true nature of transition. We defined the range of Tʒ in a dynamic way based on the minimum support value and is defined in the following way,

$T_ʒ(X) = [(d)\ \%, 100\%]$

Where *d* has been defined in the following way,

1. if $s \le 10\%$ then d=10.
2. if $s > 10\%$ and $s \le 90\%$ then $d=(s*10+1)*5$.
3. if $s > 90\%$ then d=50.

The first condition is applicable when the minimum support is less than or equal to 10% of the database and it ensures that the portion of database ignored should not be less than 10% of the database, the second condition is applicable when the minimum support is greater than 10% and less than 90% and a portion of the ignored database is varied from 10% to 50% and the third condition is applicable when the support is greater than or equal to 90% and it restricts the ignored portion of the database to 50%.

Definition 8. (Substantial Landmark (SLM(X))) an i^{th} landmark of an item set in a D is called substantial landmark for the item set X if it fulfills the following conditions,

1. *$Sup^i(X, D) \ge t_s$, where t_s is a user defined pattern support threshold*
2. *For all $j \le i$, $Sup^j(X, D) < t_s$ $|t_j \in T_ʒ$.*

The substantial landmark of an item set X is the first landmark in $T_ʒ$ where the $Sup^i(X, D)$ is more than or equal to t_s. Each item set has its own substantial landmark.

C. Alternative Transitional Patterns

Definition 9. (Alternative Transitional Pattern) An item set X is called alternative transitional pattern if there exist at least one land mark *i* which is in the range of $T_ʒ$, such that:

1. *$i > SLM(X)$ AND $Sup^i(X, D) \ge t_s$*
2. *$|Tran^i(X, D)| >= t_t$, where* t_t is the user defined transitional pattern threshold.

The first condition ensures that the substantial landmark of the item set X has been already identified and the second condition ensures that the transitional ratio is at least the user defined transitional pattern threshold t_t. The alternative transitional patterns are also two types, Alternative Major Transitional Patterns (AMJTP) and Alternative Minor Transitional Patterns (AMITP); if there exist one land mark which identifies the pattern as a transitional pattern and the transitional ratio is positive value then the pattern is called as alternative major transitional pattern and if the transitional ratio is negative then it is called as alternative minor transitional pattern.

D. Discussion on transitional pattern threshold and Reliable values for transitional pattern threshold

Both transitional pattern threshold and Pattern support threshold are user defined parameters. The transitional pattern threshold helps in eliminating the landmarks which may have very low value as a transitional value and it does not help in finding the behavior of the pattern.

The transitional pattern threshold value should be set to be reliable one otherwise the nature of the pattern may be biased towards +1 or towards -1 and it will not reveal the dynamic nature of the pattern. The following result will give the range of the transitional pattern threshold value.

Result: The transitional pattern threshold of any alternative transitional pattern must be less than or equal to (*1-s*).

Proof: For contradiction assume that the transitional ratio threshold of an alternative transitional pattern is more than (1-s) for some item set X (Assume a = $Sup^i(X, D)$, b= Maximum ((1-a), a), $\alpha > 0$ and $\alpha<s$).

From the definition of transitional ratio,

$(a-s)/b = (1-s)+\alpha$

$a-s = b-bs+b\alpha$ ---------1

When b = a (i.e. a>= 1-a) and putting it in 1,

$a-s = a-as+a\alpha$,

$a(s-\alpha) = s$

$a = s/(s-\alpha)$ ----------2

When b = 1-a (i.e., 1-a > a) and putting it in 1,

$a-s = 1-s+\alpha-a+as-a\alpha$

$2a-as+a\alpha = (1+\alpha)$

$a(2-s+ \alpha) = (1+ \alpha)$

$a = (1+ \alpha) / ((1+ \alpha) +(1-s))$ ---------3

From 2 it is evident that if the transitional ratio of X at any land mark is greater than (1-s) if and only if the support at that land mark must be more than one and is not possible. -----------4

In equation 3 the value of *a* is greater than 0.5. Since the numerator is in between more than one and less than two and the denominator is also more than one and less than two, and in all cases denominator is more than the numerator. Since (1-*a*) > *a* and it is not possible to get more than 0.5 values for *a*. From 3 it is evident that if the transitional ratio of X at any land mark is greater than (1-s) if and only if the value of a is greater than 0.5 and is not possible. -------------5

From 4 and 5 it can be concluded that the transitional pattern threshold of an alternative transitional pattern must be less than or equal to (1-s).

E. Major and Minor landmarks

There may evolve many land marks which identifies the alternative transitional patterns, and it is very hectic to record such each and every landmark for every frequent item set. We report one landmark which reveals the major and minor transitional patterns for each pattern in a user defined period of length (partition the database into partitions P_1, P_2, ...,P_n). The user period of length may be a day, a week, a month or a year.

Definition 10. (Major landmark ($MJ_i(X)$) the landmarks j, j+a, j+b ... m identifies a pattern X as an alternative transitional pattern in a partition P_i. Among these landmarks the landmarks which give the maximum transitional ratio are called as a major landmarks of X in P_i.

Definition 11. (Minor landmark ($MI_i(X)$) the landmarks j, j+a, j+b ... m identifies a pattern X as an alternative transitional pattern in a partition P_i. Among these landmarks the landmarks which give the minimum transitional ratio are called as minor landmarks of X in $P_{i.}$

F. ATP-Mine Algorithm

Here we present ATP-Mine algorithm, for mining the set of major transitional patterns and minor transitional patterns and their respective major and minor landmarks in each partition. The algorithm is given as follows.

Algorithm: ATP-Mine. (Mine set of Alternative Transitional Patterns and their major and minor landmarks)

Input: A transaction database (D), Minimum support threshold s, Pattern support threshold (t_p), Transitional pattern threshold (t_t), no of partitions (n).

Output: The set of all Alternative Transitional Patterns S_{JTP} and S_{ITP} with their major and minor landmarks in each partition.

Method:

1. Calculate the T_3.
2. S_{JTP} =0, S_{ITP}=0, C=Φ;
3. Read the database from partition P_1 to partition P_n.
4. In each partition an item set X is put into the list candidate list C with the current partition number and support of the item set X in the current partition if it was not in X and it gains the minimum support in the current partition P_i. At the end of each partition P_i an item set X is removed from C if it is infrequent from its induction into list C otherwise the item set X will be kept in C by updating its support count value.
5. **for all** i←1 to n

6. **for all** item sets e ϵ C
7. MaxTran^i(e)←0, MinTran^i(e)←0, J_e^i←(0, 0), I_e^i←(0, 0), SLM(e)←0
8. **for each** t ϵ P_i
9. **for all** j←1 to |t| AND **for each** j subsets e of t **AND if** e ϵ C
10. C_e.count++; v← C_e.count;
11. **if** t's position satisfying T_3
12. **if** SLM(e) = 0
13. **if** Sup^v(e, D) ≥ t_s
14. SLM(e) = **v**
15. **else if** Sup^v(e, D) ≥ t_s **AND** Tran^v(e, D) ≥ t_t
16. **if** e ∉ S_{JTP}
17. Add e to S_{JTP}, J_e^i←(v, t) and MaxTran^i(e) ← Tran^v(e, D).
18. **else if** Tran^v(e, D)>MaxTran^i(e) J_e^i ← (v, t) and MaxTran^i(e) ←Tran^v(e, D).
19. **else if** Tran^v(e, D)=MaxTran^i(e)
20. Append (v,t) to J_e^i
21. **else if** Sup^v(e, D) ≥ t_s AND|Tran^v(e, D)| ≥ t_t
22. **if** e ∉ S_{ITP}
23. Add e to S_{ITP}, I_e^i←(v, t) and MinTran^i(e) ← Tran^v(e, D).
24. **else if** Tran^v(e, D)< MinTran^i(e)
25. I_e^i ← (v, t) and MinTran^i(e) ← Tran^v(e, D).
26. **else if** Tran^v(e, D)= MinTran^i(e)
27. Append (v,t) to I_e^i
28. **if** e ϵ C is infrequent
29. Remove e from C and all its entries from the respective sets.
30. **End.**

The algorithm takes two scans on the database, the steps 3 and 4 uses the first scan to extract the candidate item sets from the database, and it uses the variation of partition based algorithm [17] instead of collecting all the locally frequent item sets as global candidate item sets collect only those item sets which were frequent from its induction into the candidate set to the end of the database. The steps 5 to 28 used to extract the frequent patterns, alternative transitional patterns and their major & minor landmarks in each partition by scanning the database once. The sets S_{JTP} and S_{ITP} records the patterns which are identified as major transitional patterns and minor transitional patterns. The variables MaxTran and MinTran records the transitional ratios exist at the major and minor landmarks of respective patterns in each partition. The variables J and I records the landmarks and its positions where the major and minor landmarks take place for each pattern and at each partition. The step 11 verifies whether the position of the transaction is within the range of T_3 or not, if it is within the range of T_3 then it verifies whether the substantial landmark for this item set has been identified or not, it if has not been identified then it verifies the Sup^v(e, D), and if it is more than or equal to the pattern support threshold then S(e) is set to one in step 14. The steps 15 to 21 verify whether this landmark of e contributed to the major transitional pattern or not and if so, the landmark is a major landmark for this item set in the current partition. The steps 22 to 28 verify whether this landmark of e contributed to the minor transitional pattern and its respective minor landmarks. At the end of each partition P_i the item sets which were inducted into the list C with the current partition number in the first scan are tested for their frequency over the entire database. If they are infrequent they will be removed from C and all its entries in S_{JTP}, S_{ITP}, MaxTran, MinTran, J and I.

IV EXPERIMENTAL RESULTS

To present the usage of Alternative Transitional Patterns by the ATP-Mine algorithm we made experiments on two different data sets, the first one is a retail market basket dataset and the second one is a mushroom dataset and these two datasets can be downloaded from the Frequent Item Set Mining Dataset Repository (http://fimi.ua.ac.be/data/).

A. Retail Market dataset

The characteristics of retail dataset are presented in table 5. The retail data set was supplied by an anonymous Belgian retail supermarket store [6].

Table 5 shows the characteristics of the retail database when the minimum support is 0.6%, support threshold is 0.1% and transitional ratio is 0.001. The total frequent patterns are 417 and the total alternative transitional patterns are 405.

Table 5 Retail Database Characteristics

Support = 0.6 %,, No of Partitions = 8, Pattern support threshold=0.1%, Transitional Ratio=0.001, T_3 = [10% ,100%]								
Database	*items*	*# of transactions*	*# of frequent Patterns*	*# of alternative transitional patterns*	*# of patterns have MJTP*	*# of patterns have MITP*	*# of patterns have both MJTP and MITP*	*# of patterns are not alternative transitional patterns*
Retail	*16469*	*88162*	*417*	*405*	*374*	*85*	*54*	*12*

Table 6 Number of MJTP and MITP in Each Partition for the Retail Database

	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8
# of Alternative Major Transitional Patterns	282	305	318	324	272	277	303	323
# of Alternative Minor Transitional patterns	39	51	28	18	16	11	13	14

There are 12 patterns which do not have any transition with respective the threshold values set by the user. There are 54 patterns have both the Major and Minor transitions. Table 6 shows the number of alternative major transitional patterns and the number of alternative minor transitional patterns in each partition. The table 7 and table 8 shows all the major and minor landmarks of the patterns R39R12925 and R12925. The Pattern R39R12925 has no transition in the first three partitions and it has minor landmarks in the partitions 4, 5 and 6 and in the partitions 7 and 8 has major landmarks. From table 7 and its SLM it can be observed that the pattern R39R12925 is infrequent with respective pattern threshold at the beginning of the database and it is gaining more support as the database is progressing towards the end. For the pattern R12925 and from its SLM in table 8, it can be observed that it is infrequent with respective pattern threshold at the beginning of the database and it is gaining more support as the database is progressing towards the end.

Table 7 The Major and Minor Landmarks of Item set R39r12925

Pattern	**R39R12925**					
SLM						
	Major Landmarks			**Minor Land marks**		
	i, $P^i(X,D)$	$Sup^i(X, D)$	$Tran^i(X,D)$	i, $P^i(X, D)$	$Sup^i(X, D)$	$Tran^i(X,D)$
Partition-1	-	-	-	-	-	-
Partition-2	-	-	-	-	-	-
Partition-3	-	-	-	-	-	-
Partition-4	-	-	-	43,41207	0.001044	-0.00496
Partition-5	-	-	--	76,44509	0.001708	-0.0043
Partition-6	-	-	-	234,55515	0.004215	-0.00179
Partition-7	684,77131	0.008868	0.002894			
Partition-8	937,87994	0.010649	0.004698			

Table 8 The Major and Minor Landmarks of Item set R12925

Pattern	**R12925**					
SLM						
	Major Landmarks			**Minor Landmarks**		
	i, $P^i(X,D)$	$Sup^i(X, D)$	$Tran^i(X,D)$	i, $P^i(X, D)$	$Sup^i(X, D)$	$Tran^i(X,D)$
Partition-1	-	-	-	-	-	-
Partition-2	-	-	-	-	-	-
Partition-3	-	-	-	-	-	-
Partition-4	-	-	-	42,40697	0.001032	-0.00497
Partition-5	-	-	--	113,44509	0.002539	-0.00347
Partition-6	707,66052	0.010704	0.004455	-	-	-
Partition-7	1081,77140	0.014014	0.008127	-	-	-
Partition-8	1466,87994	0.01666	0.010841	-	-	-

Table 9 Mushroom Dataset Characteristics

	***Support = 50 %,, No of Partitions = 8, Pattern support threshold=10%, Transitional Ratio=0.1,* $T_ʒ$ = [30% ,100%]**						
Database	*# of items*	*# of transactions*	*# of frequent Patterns*	*# of Alternative Transitional Patterns*	*#of patterns have MJTP*	*# of patterns have MITP*	*#of patterns have both MJTP and MITP*
Mushroom	*119*	*8124*	*153*	*153*	*145*	*16*	*8*

Table 10 Number of MJTP and MITP for the Mushroom Database

	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8
Total Number of Alternative Major Transitional Patterns	0	0	137	137	137	111	119	115
Total Number of Alternative Minor Transitional patterns	0	0	16	16	16	16	0	0

Table 11 The Major and Minor Landmarks of Item set M24

Pattern	**M24**					
SLM	**Major Landmarks**			**Minor Landmarks**		
	i, P^i(X,D)	*Sup^i(X, D)*	$Tran^i$(X,D)	i, P^i(X, D)	*Sup^i(X, D)*	$Tran^i$(X,D)
Partition-1	-	-	-	-	-	-
Partition-2	-	-	-	-	-	-
Partition-3	-	-	-	909,3035	0.299506	-0.28622
Partition-4	-	-	-	915,3058	0.299215	-0.28651
Partition-5	-	-	--	1340,4061	0.329968	-0.25377
Partition-6	-	-	-	2192,5076	0.431836	-0.11997
Partition-7	-	-	-			
Partition-8	4728,8124	0.584441	0.144482			

B. Mushroom dataset

The Mushroom dataset was prepared by Roberto Bayardo from the UCI datasets. Table 9 shows the characteristics of mushroom dataset. When the minimum support is 50% then the total frequent patterns are 153. All the extracted frequent patterns exhibited the transitions at some part of the database. There are only eight patterns which have both major and minor landmarks. Table 10 shows the total number of alternative major and minor transitional patterns in each partition. The partition one and partition two are not in the range of Tʒ, and we observed all the 16 alternative minor transitional patterns in the partitions 3, 4, 5 and 6 are same patterns. Table 11 and table 12 shows all the major and minor landmarks of the patterns M24 and M2. The Pattern M24 has minor transition at the beginning of the database and it has major transition at the end of the database. The pattern M2 has no minor transition in the database since it has more support at the beginning of the database and it is losing the support at the end of the database.

Figure 1 to 3 shows the some of the patterns in Retail database. Figure 1 show the some patterns which have only major transitional patterns, the patterns R1004 and R270R271 steadily in almost in the same transitional from the beginning of the database to the end of the database, but the pattern R32R38R39R41 losing its support towards the end of the database. Figure 2 shows the patterns which have both major and minor landmarks.

Table 12 Major and Minor Landmarks of Item set M12

Pattern	M2					
SLM						
	Major Landmarks			Minor Landmarks		
	i, $P^i(X,D)$	*$Sup^i(X, D)$*	$Tran^i(X,D)$	i, $P^i(X,D)$	*$Sup^i(X, D)$*	$Tran^i(X,D)$
Partition-1				-	-	-
Partition-2				-	-	-
Partition-3	2710,3001	0.903032	0.44631	-	-	-
Partition-4	2743,3047	0.90023	0.444585	-	-	-
Partition-5	3327,4063	0.818853	0.38939	-	-	-
Partition-6	3452,5080	0.679528	0.264195	-	-	-
Partition-7	3684,6121	0.601862	0.169245	-	-	-
Partition-8	4748,8124	0.584441	0.144482	-	-	-

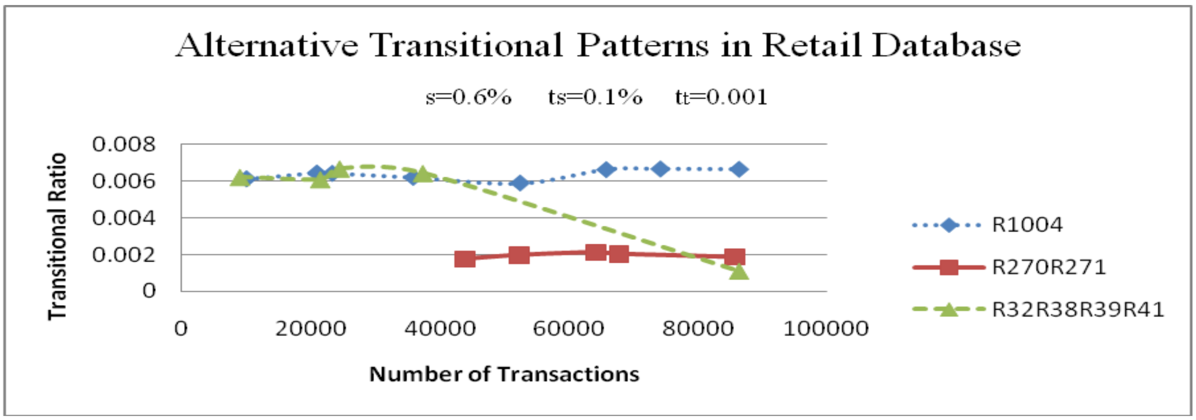

Figure 1 Some patterns which have only major landmarks in retail database

The patterns R39R78 and R48 and R413 are in the negative transition at the beginning of the database and are slowly gaining the more support towards the end of the support, but the pattern R12925 losing is in the positive transition at the beginning of the database and is losing the support toward the end of the database. Figure 3 shows the some patterns which have only minor landmarks.

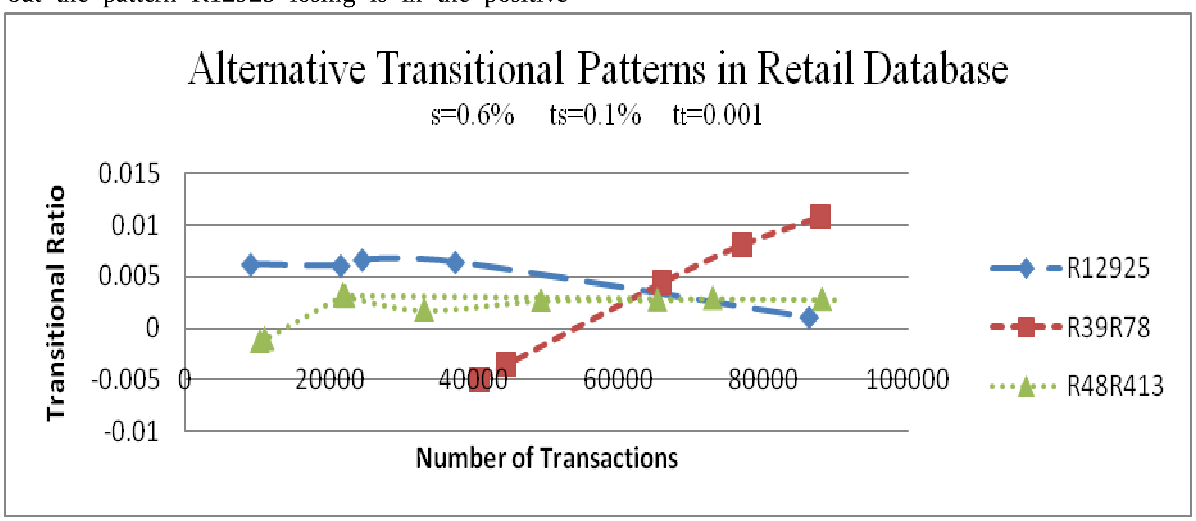

Figure 2 Some patterns which have both major and minor landmarks in retail database

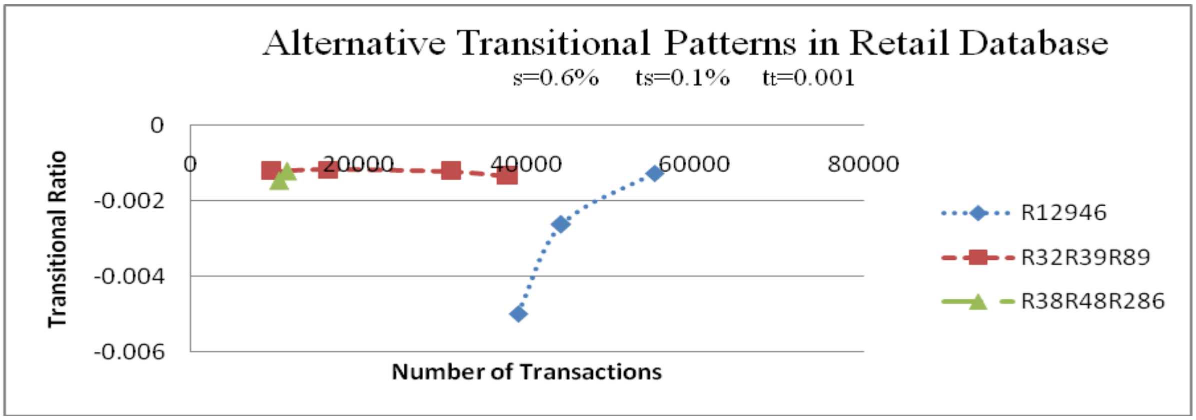

Figure 3 Some patterns which have only minor landmarks in retail database

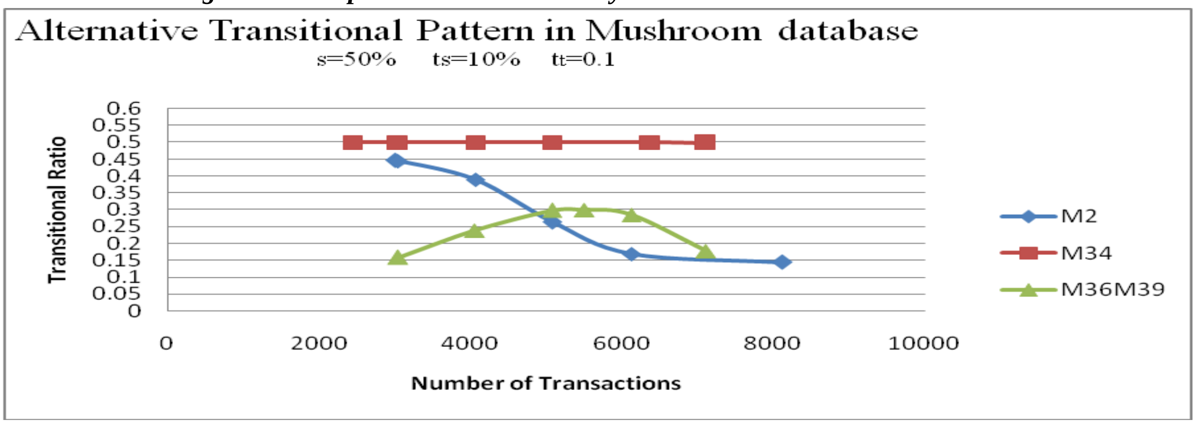

Figure 4 Some patterns which have only major landmarks in mushroom database

The figure 4 to 6 shows the some of the patterns in Mushroom database. Figure 4 shows the some patterns which have only major landmarks, Figure 6 shows some patterns which have only minor landmarks patterns and Figure 6 shows some of the patterns which have both major and minor landmarks.

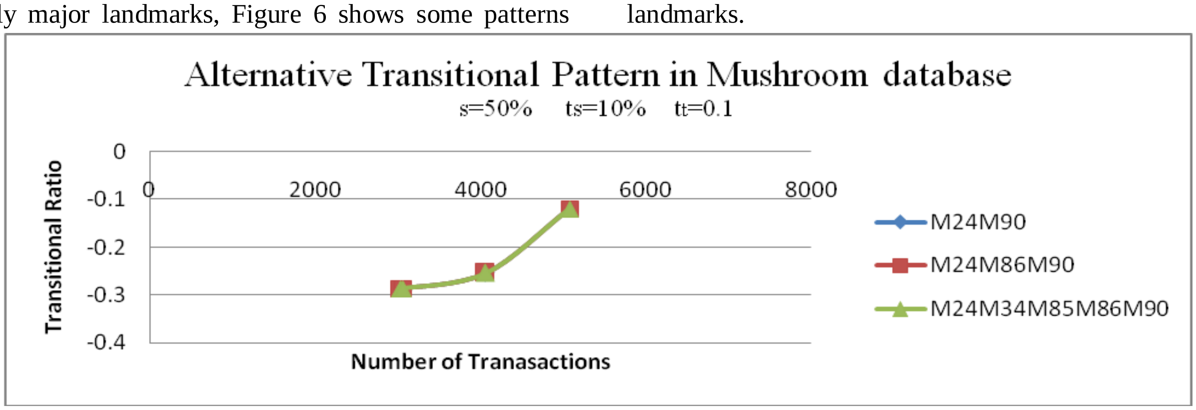

Figure 5 Some patterns which have only minor landmarks in mushroom database

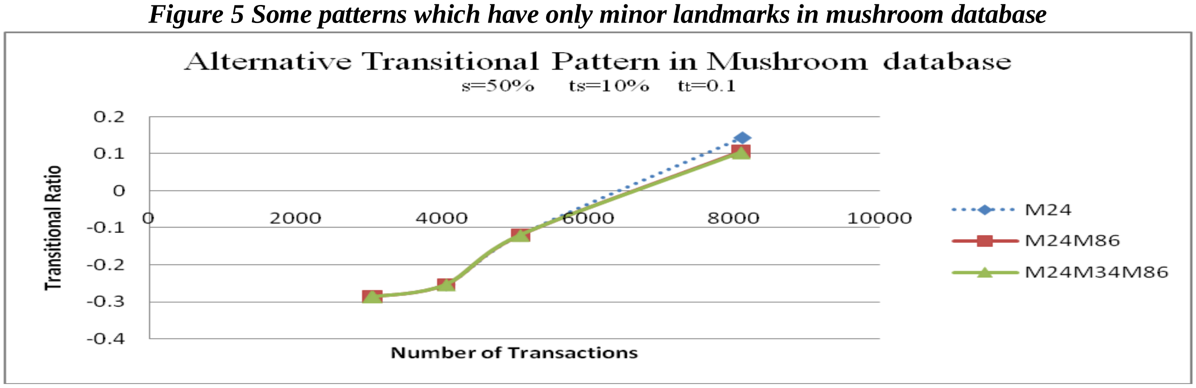

Figure 6 Some patterns which have both major and minor landmarks in mushroom database

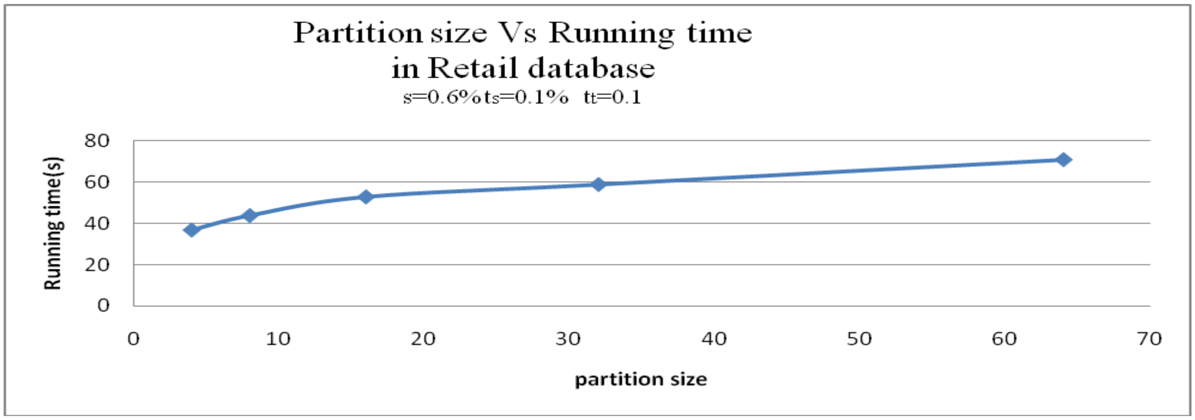

Figure 7 Running time for extracting ATP from candidate item sets with increasing partition size

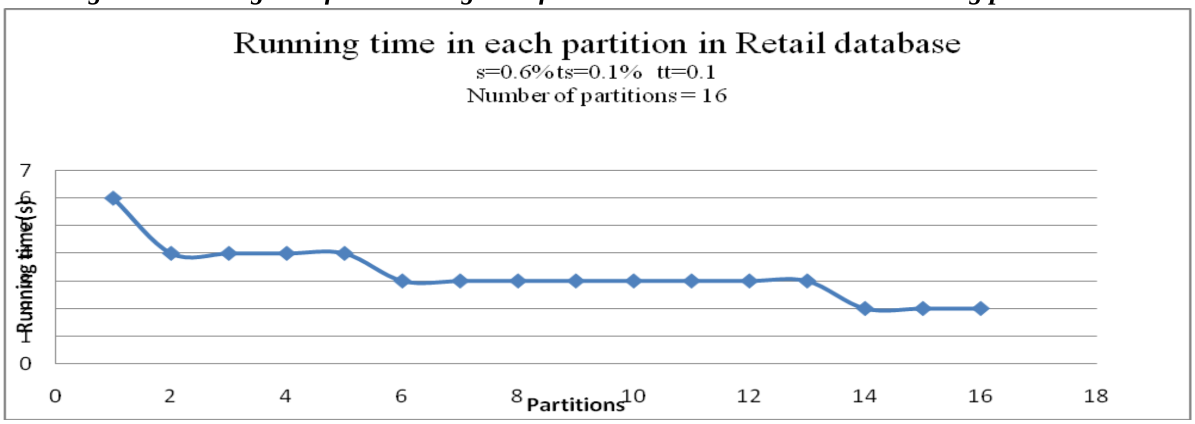

Figure 8 Running time for extracting ATP from the candidate item sets in each partition

The figure 7 and figure 8 shows the running times for extracting the frequent item sets and alternative transitional patterns from the candidate item sets (scan-2 of the ATP-Mine algorithm). In figure 7 as the size of the partition doubles the increase in time is negligible one and figure 8 shows the running time in each partition as the algorithm is moving towards the end of the database the running time is slowly decreasing.

V CONCLUSION

The limitation in the existing FPM frame work that the extracted knowledge does not give any knowledge about the dynamic behavior of the extracted frequent patterns. The TP-Mine algorithm [20] extracted transitional patterns does not valid on a dynamic database. In this paper, we re-define the transitional ratio and renamed the patterns as alternative transitional patterns and presented an algorithm which takes only two scans on the database. It is evident from the experimental results presented in this paper the algorithm presented is highly scalable.

REFERENCES

[1] Agrawal, A., Imielinski, T. and Swami, A. Mining Association Rules between Sets of Items in Large Databases, Proc. 1993 ACM SIGMOD International conference on Management of Data (SIGMOD'93), pp. 207-216, 1993.

[2] Agrawal, R. and Srikant, R., Fast Algorithms for Mining Association rules, Proc. 20th International conference on Very Large Databases, pp. 487-499, 1994.

[3] Agrawal, R. and Srikant, R. Mining sequential patterns. In proc. 1995 International Conference on Data Engineering. (ICDE'95), pages 3-14, Taipei, Taiwan, March 1995.

[4] Ale, J. M., Rossi, G. H. An approach to discovering temporal association rules. In SAC'00: Proceedings of the 2000 ACM symposium on applied computing, pp 294-300 2000.

[5] Bashar, S. and Masseglia, F. Discovering the behaviours: time is an essential element of the context, Knowledge and Informations systems, Vol 28, no. 3, 311-331, August 2011.

[6] Brigis, T., Swinnen, G., Vanhoof, K., Wets, G. Using Association Rules for Product Assortment Decessions: A Case Study, Proc. Fifth ACM SIGKDD, International Conference on Knowledge Discovery and Data Mining, pp. 254-260, 1999.

[7] Gharib, T. F., Nassar, H., Taha, M., Abraham, A. An Efficient Algorithm for Incremental Minig of Temporal Association Rules, Data and Knowledge Engineering 69 (2010) 800-815, 2010.

[8] Grahne, G., Zhu, J. Effeciently using prefix-trees in mining frequent item sets. In Proceedings of the ICDM'03 international workshop on frequent item sets mining

implementations (FIMI'03), Melbourne, FL, pp 123-132. 2003.
[9] Han, J. W., Pei, J., Yin, Y. W., Mao, R. Y. Mining Frequent Patterns without Candidate Generation: A Frequent-Pattern Tree Approach, Data Mining and Knowledge Discovery, vol. 8, no. 1, pp. 53-87, 2004.
[10] Huang, Y., and Zhang, P. A Freamework for Mining Sequential Patterns from Spatio Temporal Event Data Sets, IEEE transactionsons on Knowledge and Data Engineering, vol.20, no.4, 2008.
[11] Li, Y. J., Ning, P., Sean, X., Wang and Jajodia, S. Discovering Calendar-Based Temporal Association Rules, Proc. Eighth Int'l Symp. Temporal Representation and Reasoning (TIME '01), pp. 111-118, 2001.
[12] Liao, Z. X. and Peng, W. C. Mining sequential patterns across multiple sequence databases, Data and Knowledge Engineering 68 1014–1033, 2009.
[13] Mannila, H., Toivonen, H. and Inkeri Verkamo, I. Discovery of Frequent Episodes in Event Sequences, Data Mining and Knowledge Discovery, vol. 1, pp. 259-289, 1997.
[14] Marko, S. and Mannila, H. Using Markov Chain Monte Carlo and Dynamic Programming for Event Sequence Data, Knowledge and Information Systems, vol. 7, no. 3, pp 267-285, 2005.
[15] Ozden, B., Ramaswamy, S. and Silberschatz, A. Cyclic Association Rules, Proc. 14th Int'l Conf. Data Eng. (ICDE '98), pp. 412-421, 1998.
[16] Roddick, J. F., Spiliopolou, M. A survey of temporal knowledge discovery paradigms and methods, IEEE TKDE 14(4): 750-767, 2002.
[17] Savasere, A., Omieeinski, E., Navathe, E. An Efficient Algorithm for Mining association rules in Large Databases. In Proceedings of the 1995 international conference on very large databases (VLDB'95), Zurich, Switzerland, pp 432-443, 1995.
[18] Srikant, R. and Agrawal, R. Mining Generalized Association Rules, 21st VLDB conference, Zurich, Switzerland, 1995.
[19] Toivonen, H. Sampling large databases for association rules, In Proceedings of the 1996 international conference on very large databases (VLDB'96), Bombay, India, pp 134-145, 1996.
[20] Wan, Q., An, A. Discovering Transitional Patterns and their significant milestones in transaction databases - IEEE transactionson Knowledge and Data Engineering, Dec', 2009.
[21] Yan, X., Han, J. and Afshar, R. CloSpan: Mining Closed Sequential Patterns in Large Databases. In SDM'03, San Francisco, CA, May 2003.
[22] Subramanyam, R. B. & Suvvari, S.R. (2012). Partition-Based Approach for Fast Mining of Transitional Patterns. 2012 UK Sim 14th International Conference on Computer Modelling and Simulation, 151-155.