International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 1 Issue 6, August - 2012

Transitional Pattern Mining for Stream Data of Sensor Networks

P. Sudheer Babu[1], V. Ranjith Naik[2]

[1]*II/II M.Tech Department of C.S.E .*, Swarnandhra college of Engg & tech
[2]Assoc. Professor CSE Dept, Swarnandhra college of Engg &tech

ABSTRACT

Mining Association rules is important for Knowledge discovery from Transactional data bases. To generate Association rules ,process must find all frequent patterns .Most of the existing association rule mining algorithms do not consider the time stamp associated with transactions. In this paper we extend the existing frequent pattern mining for web click data base, to take into account the time stamp of each transaction and discover patterns whose frequency dramatically changes over time(called Transitional patterns). Transitional patterns are frequent patterns whose occurrences high when time goes on. We use the concept called significant Milestones. For a transitional pattern , milestones are time points at which the frequency of patterns most significantly. More over we develop an algorithm to mine offline data stream(bulk arrival) and produce transitional patterns along with their milestones. Experimental studies on web click data illustrate that mining positive and negative transitional patterns is highly promising as a practical and useful approach for discovering knowledge from web click data stream.

KEYWORDS : Data Mining, Data Streams, Transitional Pattern, Significant Milestone.

1. INTRODUCTION

A transaction database usually consists of a set of time stamped transactions. Mining frequent item sets or patterns from a transaction database is one of the fundamental and essential operations in many data mining applications, such as discovering association rules, strong rules, correlations, and many other important discovery tasks. The problem of mining frequent itemsets is formulated as finding all the itemsets from a transaction database that satisfy a user specified support threshold.

A data stream[12] is an ordered sequence of items that arrives in timely order. Different from data in traditional static databases, data streams are continuous, unbounded, usually come with high speed and have a data distribution that often changes with time. As the number of applications on mining data streams grows rapidly, there is an increasing need to perform Transitional Pattern mining on stream data.

This paper extend the previous work[2][3][4][6][7] on traditional frequent pattern mining framework to take into account the time stamp of each transaction, i.e., the time when the transaction occurs. It use a new type of patterns, called transitional patterns, to represent patterns whose frequency dramatically changes over time. Transitional patterns include both positive and negative transitional patterns.

The frequency of a positive transitional pattern increases dramatically at some time point of a Web click data stream, while that of a negative transitional pattern decreases dramatically at some point of time.

The contributions of this paper are summarized as follows:

. - Propose a framework for mining a new class of patterns, called transitional patterns. The frequencies of these patterns change significantly at some time points of a Data stream.

-Use[13] the concept of significant milestones for each transitional pattern, which are specific time points at which the frequency of the pattern increases or decreases most significantly.

. - An algorithm, called TPM-DS, is designed to mine the set of transitional patterns along with their significant milestones

The remaining of the paper is organized as follows: Section 2 describes the terminologies used in Transitional Pattern Mining for Data streams and the concepts of positive and negative transitional patterns and their significant milestones. Sections 3 present Frame work and an algorithm for mining transitional patterns and their significant milestones for Data streams. Section 4 present an experimental study to demonstrate the utility of transitional patterns in Web click Data stream. with related work. Finally, Section 8 conclude the paper.

2.PROBLEM DEFINITION

Let I={i1,i2,i3…,im} be a set of m distinct items. A subset X subset or equal to I is called an itemset or a pattern. A k-itemset is an itemset that contains k items.. A transaction over I is a couple T =(tid,I}, where tid is the transaction identifier (or time stamp) and I I is an itemset. A transaction T =(tid,I} is said to support an itemset X I if and only if X subset or equal to I. A Data stream, DS over I is a set of transactions ..The following Table represent Web click data stream format

TID	**Sist of Sinks CSicked**	**Timestamp**
1	S1,S2,S3,S5	0.0.1
2	S1,S2	0.0.2
3	S1,S2,S3,S8	0.0.3
4	S1,S2,S5	0.0.4
5	S1,S2,S4	0.0.5
6	S1,S2,S4,S5,S6	0.0.6
7	S1,S2,S3,S4,S6	0.0.7
8	S1,S4,S6	0.0.8
9	S4,S5,S6	0.0.9
10	S1,S2,S3,S4,S5,S6	0.1.0
11	S1,S3,S4,S6	0.1.1
12	S1,S3,S5	0.1.2
13	S1,S2,S3,S6,S7	0.1.3
14	S1,S3,S4,S5	0.1.4
15	S1,S3,S4	0.1.5
16	S1,S2,S3,S5	0.1.6

The cover of an Itemset X in DS, denoted by Cov(X,DS) consist of the set of transactions in DS that support X. An itemset X in DS has Support denoted by Sup(X,DS),which is the ratio of transactions in DS containing X. For the above example Sup({S1,S2},DS) is equal to 10/16. If the Support of a pattern exceeds minimum support then the pattern is called frequent pattern.

Definition 2.1. Assuming that the transactions in a Data stream DS are ordered by their time-stamps, the position of a transaction T in D, denoted by ρ(T), is the number of transactions whose time-stamp is less than or equal to that of T.

Definition 2.2. The ith transaction of a pattern X in D, denoted by $f^i(X)$, is the ith transaction in cov(X) with transactions ordered by their positions.

Definition 2.3 (ith milestone). The ith milestone of a pattern X in D, denoted by $€^i(X)$, is defined as

$$€^i(X)=\rho(f^i(X))/\|DS\|*100\ \%$$

According to this definition, the ith milestone of pattern X represents the relative position (expressed in a percentage) of the ith transaction of X in D. For instance, in the example Data stream we have $€^4$ (S1,S2)=25% and $€^4$ (S1,S3)=62:5%.

Definition 2.4. The support of a pattern X before its ith milestone in D, denoted by $Sup^i_{-}X)$, is defined as:

$$Sup^i_{-}(X) = i/\rho(f^i(X)).$$

Definition 2.5. The support of a pattern X after its ith milestonein D, denoted by $Sup^i_{+}(X)$, is defined as:

$$Sup^i_{+}(X)=(\|Cov(X)\|-i)/(\|DS\|-\rho(f^i(X)))$$

For the above example, Sup^6_{-}(S1,S2)=1.0 and p Sup^6_{+}(S1,S2)= 0:4. that mean up to the time stamp 0.0.6 support of web clicks S1,S2 are occurred 100 % after that only 40%.

The main Problem for finding Transitional patterns of data streams is to find frequent patterns in one scan of data base.

3.Transitional pattern mining for Data Stream Framework and Algorithm

The frame work Transitional Pattern Mining for Data Stream(TPM-DS) is depicted in the following diagram. It reads Web click data stream along with input parameters like minimum support(t_s),mile stone range($t_€$), pattern threshold(t_t) and sliding window size. It produce Transitional patterns and their milestones.

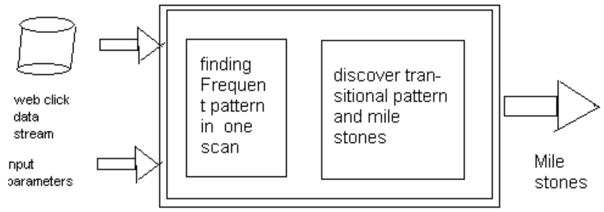

Fig.1. TPM-DS Framework

3.1 FINDING FREQUENT PATTERNS IN ONE SCAN.

Due to the characteristics of stream data, there are some inherent challenges for stream data Transitional pattern mining . First, due to the continuous, unbounded, and high speed characteristics of data streams, there is a huge amount of data in both offline and online data streams, and thus, there is not enough time to rescan the whole database or perform a multi-scan as in traditional data mining algorithms whenever an update occurs. Furthermore, there is not enough space to store all the stream data or online processing. Therefore, a one scan [11] of data and compact memory usage of the Transitional pattern mining technique are necessary.

We have existing algorithms like apriori[1] and FP growth[6] algorithms for finding frequent patterns both requires more than one scans of data base. Hao and Wu[11] proposed method for finding frequent patterns in one scan.

3.2 PROCESSING MODEL OF TPM-DS

Data streams consist of an ordered sequence of items. Each set of items is usually called "transaction". The issue of data processing model here is to find a way to extract transactions for Transitional pattern mining from the overall data streams. Because data streams come continuously and unboundedly the extracted transactions are changing from time to time there are three stream data processing models, Landmark, Damped and Sliding Windows.[8][9][10]

In this paper frame uses Sliding window model Sliding Windows model[10] finds and maintains frequent itemsets in sliding windows. Only part of the data streams within the sliding window are stored and processed at the time

3.3 TRANSITIONAL PATTERN MINING FOR DATA STREAM ALGORITHM.

Algorithm.

TPM-DS (Mine the set of Transitional Patterns and their significant milestones for data stream)

Input::

A Stream Data (DS), an appropriate milestone range that the user is interested ($T_{€}$), pattern support threshold (t_s), and transitional pattern threshold (t_t). Sliding Window size(W

Output:

The set of transitional patterns (S_{PTP} and S_{NTP}) with their significant milestones.

Method:

1: Extract frequent patterns, P1; P2; . . . ; Pn, and their supports using a frequent pattern generation algorithm with min_sup = t_s.

2: Scan the transactions from the first transaction to the last transaction before $T_{€}$ to compute the support counts, ck of all the n frequent patterns on this part of the data stream.

3: S_{PTP} =O;; SNTP=O;
4: **for all** k =1 to n do
5: MaxTran(Pk)=0, ;MinTran(Pk)=0
6: SFAM(Pk)=O ;; SFDM(Pk)=O ;
7: **end for**
8: **for all** transactions Ti whose position satisfying $T_{€}$ do
9: for k =1 to n do
10: **if** T i subset or equal to Pk then
11: ck =ck+ 1;
12: **if** Sup^{ck}_{-} (P_k)≥ ts and Sup^{ck}_{+} (P_k)≥ ts then
13: **if** $tran^{ck}$ (Pk) ≥ t_t then
14: **if** Pk not belongs to S_{PTP} then
15: Add Pk to S_{PTP}
16: **end if**
17: **if** $tran^{ck}$ (Pk) > MaxTran(Pk) then
18: $S_{FAM}(P_k)$={[f^{ck} (P_k); ,$tran^{ck}$ (P_k)]}
19: MaxTran(Pk) =$tran^{ck}$ (Pk)
20: **else if** $tran^{ck}$ (Pk) =MaxTran(Pk) then
21: Add [f^{ck} (Pk), $tran^{ck}$ (Pk)] to S_{FAM} (Pk)
22: **end if**
23: **else if** $tran^{ck}$ (Pk) ≤-t_t then
24: **if** Pk does not belongs to S_{NTP} then
25: Add Pk to S_{NTP}
26: **end if**
27: **if** $tran^{ck}$ (Pk) < MinTran(Pk) then
28: $S_{FDM.}$(Pk) ={[f^{ck} (Pk),$tran^{ck}$ (Pk)]}
29: MinTran(Pk) = $tran^{ck}$ (Pk)
30: **else if** $tran^{ck}$ (Pk) = MinTran(Pk) then
31: Add [f^{ck} (Pk); tranck(Pk)] to S_{FDM} (Pk)
32: **end if**
33: **end if**
34: **end if**
35: **end if**
36: **end for**
37: **end for**
.

There are two major phases in this algorithm. During the first phase (Step 1), all frequent itemsets along with their supports are initially derived using a standard frequent pattern generation algorithm, such as *Apriori* [1] or *FPgrowth* [6] but due high speed characteristics of data streams first step construct pattern tree after words it converted into FP tree[11] , with *ts* as the minimum support threshold. In the second phase (starting from Step 2 to the end), the algorithm finds all the transitional patterns and their significant milestones based on the set of frequent itemsets.

4 EXPERIMENTAL RESULTS.

To demonstrate the utility of transitional patterns and the efficiency of the *TPM-DS* algorithm, performed experiments using dataset from two real-world domain: web log data. summaries the parameters of each dataset along with the threshold values used in our experiments.

Table 1 shows the first 10 positive transitional patterns in Retail. These patterns are ranked by the transitional ratios at their significant frequency-ascending milestones. For positive transitional patterns, the greater the ratio, the higherthe rank; while for negative transitional patterns (table 2) , the less the ratio, the higher the rank.

#	*PTP*	sup_- (‰)	sup_+ (‰)	$\langle \mathcal{M}^+, tran_{\mathcal{M}^+} \rangle$ (%)	sup (‰)
1	{15000}	5.03	25.12	⟨44.17, +79.97⟩	16.25
2	{1375}	5.04	22.72	⟨62.87, +77.79⟩	11.61
3	{1859}	5.54	17.92	⟨75.00, +69.10⟩	8.63
4	{8106}	5.03	15.60	⟨71.49, +67.75⟩	8.04
5	{544}	5.05	15.27	⟨56.96, +66.92⟩	9.45
6	{1381}	5.00	15.03	⟨73.24, +66.72⟩	7.68
7	{273}	5.53	16.33	⟨57.96, +66.16⟩	10.07
8	{1509}	5.03	13.92	⟨45.50, +63.87⟩	9.87
9	{545}	5.02	13.80	⟨57.36, +63.66⟩	8.76
10	{544, 545}	5.02	13.77	⟨57.98, +63.55⟩	8.70

Table.1 Pasitive transitional patterns

#	*NTP*	sup_- (‰)	sup_+ (‰)	$\langle \mathcal{M}^-, tran_{\mathcal{M}^-} \rangle$ (%)	sup (‰)
1	{355}	50.31	7.24	⟨40.42, -85.60⟩	24.65
2	{384}	26.56	5.01	⟨52.32, -81.15⟩	16.28
3	{11034}	18.60	5.03	⟨32.35, -72.97⟩	9.42
4	{434}	33.81	9.76	⟨59.47, -71.14⟩	24.06
5	{15001}	17.03	5.04	⟨46.84, -70.39⟩	10.66
6	{15000, 15001}	16.62	5.04	⟨46.81, -69.68⟩	10.46
7	{1735}	22.00	7.75	⟨60.78, -64.76⟩	16.41
8	{396}	14.15	5.00	⟨52.92, -64.66⟩	9.84
9	{225, 396}	13.54	5.07	⟨52.90, -62.56⟩	9.55
10	{1322}	15.69	5.96	⟨41.26, -62.03⟩	9.97

Table.2 Negative transitional patterns

5 CONCLUSION

In this paper, we used a novel type of patterns, positive and negative transitional patterns, to represent frequent patterns whose frequency of occurrences changes significantly at some point of time in the transaction database.

And used the concepts of significant frequency ascending milestones and significant frequency-descending milestones to capture the time points where the frequency of patterns changes most significantly. Moreover, we develop the *TPM-DS* algorithm to mine from a Web click data stream the set of transitional patterns along with their significant milestones.

REFERENCES

[1] R. Agrawal, T. Imielinski, and A. N. Swami. Mining association rules between sets of items in large databases. In *Proc. Of the 1993 ACM SIGMOD Int. Conf. on Management of Data*, pages 207–216, 1993.

[2] R. Agrawal and R. Srikant. Fast algorithms for mining association rules. In *Proc. of the 20th Int. Conf. on Very Large Data Bases*, pages 487–499, 1994.

[3] D. Burdick, M. Calimlim, and J. Gehrke. Mafia: A maxima frequent itemset algorithm for transactional databases. In *Proc. of the 17th Int. Conf. on Data Engineering*, 2001.

[4] G. Dong and J. Li. Efficient Mining of Emerging Patterns: Discovering Trends and Differences. *Knowledge Discovery and Data Mining*, pages 43–52, 1999.

[5] J. Han, J. Pei, and Y. Yin. Mining frequent patterns without candidate generation. In *Proc. of ACM-SIGMOD Int. Conf. on Management of Data*, pages 1–12, 2000.

[6] J. Pei, J. Han, and R. Mao. CLOSET: An efficient algorithm for mining frequent closed itemsets. In *ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery*, pages 21–30, 2000.

[7] P. Tan, V. Kumar, and J. Srivastava. Indirect association: mining higher order dependencies in data. In *Proc. of the 4th European Conf. on Principles and Practice of Knowledge Discovery in Databases*, pages 632–637, 2000.

[8] Moses Charikar, Kevin Chen, Martin Farach-Colton; Finding Frequent Items in Data Streams; Theoretical Computer Science; January 2004.

[9] Joong Hyuk Chang, Won Suk Lee; A Sliding Window Method for Finding Recently Frequent Itemsets over Online Data Streams; Journal of Information Science and Engineering; July 2004.

[10] Chih-Hsiang Lin, Ding-Ying Chiu, Yi-Hung Wu, Arbee L. P. Chen; Mining Frequent Itemsets from Data Streams with a Time Sensitive Sliding Window; SIAM Int'l Conf. on Data Mining; April 2005.

[11] Hao Huang, Xindong Wu, Richard Relue; Association Analysis with One Scan of Databases; IEEE Int'l Conf. on Data Mining; December 2002.

[12] Q. Wan and A. An, “Transitional Patterns and Their Significant Milestones,” Proc. Seventh IEEE Int’l Conf. Data Mining, 2007.

[13] Jiang, N., & Gruenwald, L. (2006). Research issues in data stream association rule mining. SIGMOD Record, 35(1).