International Journal of Information Technology and Knowledge Management
January-June 2011, Volume 4, No. 1, pp. 27-32

DISCOVERING ACTIVE AND PROFITABLE PATTERNS WITH RFM (RECENCY, FREQUENCY AND MONETARY) SEQUENTIAL PATTERN MINING–A CONSTRAINT BASED APPROACH

C K Bhensdadia[1] & Y. P. Kosta[2]

Sequential pattern mining is an extension of association rule mining that discovers time-related behaviors in sequence database. It extends association by adding time to the transactions. The problem of finding association rules concern with intra-transaction patterns whereas that of sequential pattern mining concerns with inter-transaction patterns. Generalized Sequential Pattern (GSP) mining algorithm is a well known Apriori-based algorithm used for sequential pattern mining. The GSP algorithm suffers from several deficiencies whenever the database size is large, like: too many scanning of database when seeking frequent sequences and very large amount of candidate sequences generated unnecessary. These problems can be solved by applying various constraints in sequential pattern mining process. Constraint based sequential pattern mining discovers only those patterns which satisfy certain constraints; hence it improves the effectiveness and efficiency of sequential pattern mining process. Our proposed algorithm modifies the traditional sequential pattern mining algorithm GSP, so that, except the frequency two additional constraints, recency and monetary are considered to discover the RFM (Recent, Frequent and Monetary) sequential patterns. The advantage of considering these two additional factors is that this can ensure all patterns are recently active and profitable. Proposed approach works on time constraint. Proposed RFM sequential pattern mining approach discovers those sequential patterns from large database which are recent, frequent and which also satisfies monetary constraint.

1. Introduction

Sequential pattern mining is an extension of association rule mining that discovers time-related behaviors in sequence database. It extends association by adding time to the transactions. The problem of finding association rules concern with intra-transaction patterns whereas that of sequential pattern mining concerns with inter-transaction patterns. Sequential pattern mining technology has been applied in many domains, including web-log analysis, the analyses of customer purchase behavior, medical record analysis, etc. As the data to be mined is large, the time taken for accessing data is considerable. Generalized Sequential Pattern (GSP) mining algorithm is a well known Apriori-based algorithm used for sequential pattern mining. The GSP algorithm suffers from several deficiencies whenever the database size is large, like: too many scanning of database when seeking frequent sequences and very large amount of candidate sequences generated unnecessary. These problems can be solved by applying various constraints in sequential pattern mining process. Constraint based sequential pattern mining discovers only those patterns which satisfy certain constraints; hence it improves the effectiveness and efficiency of sequential pattern mining process.

[1,2]Dharmsinh Desai University(DDU-Nadiad), U & P U. Patel Department of Computer Engineering, Charotar University of Science & Technology-CHARUSAT, Gujarat, India

Email: [1]ckbhensdadia@yahoo.co.in, [2]ypkostaresearch@yahoo.com

Except the frequency two additional constraints, recency and monetary are considered to discover the RFM (Recent, Frequent and Monetary) sequential patterns. Recency is the period since the last purchase occurred and monetary is the amount of money spent during a certain period. The advantage of considering these two additional factors is that this can ensure all patterns are recently active and profitable. The proposed algorithm modifies GSP algorithm in two different ways. First, the proposed algorithm uses itemset as a unit to expand the patterns rather than item, which can reduce the number of phases needed to complete the algorithm, and thus can improve the efficiency. Second it builds inverse candidate tree for support counting to speed up the process of discovering recent patterns.

Although efficiency of mining the complete set of sequential patterns has been improved substantially, in many cases, sequential pattern mining still faces tough challenges in both effectiveness and efficiency. On the one hand, there could be a large number of sequential patterns in a large database. A user is often interested in only a small subset of such patterns. Presenting the complete set of sequential patterns may make the mining result hard to understand and hard to use. On the other hand, although efficient algorithms have been proposed, mining a large amount of sequential patterns from large data sequence databases is very expensive task. If we can focus on only those sequential patterns interesting to users, we may be able to save a lot of computation cost by those uninteresting patterns.

2. PROBLEMS WITH THE EXISTING APPROACH

If we consider only frequency, we may find a lot of patterns, though frequent, that are of little value. This is especially true for retailing industry, since consumers frequently buy items that are cheap such as tissue, milk, juice or soap, but they rarely buy expensive goods such as jewelry or electronic appliance. Without considering the monetary value of a pattern, decision makers will find a huge number of cheap patterns, which has little or no use in increasing the profit of a company. Also the users' behavior changes over time. So we should also consider the users' recent behavior. So it is necessary to include the recency factor and monetary factor into the sequential pattern mining. Main purpose of proposed algorithm is to find sequential patterns which satisfy frequency, recency and monetary constraints which is usually used by marketing researchers to do customer or market segmentation.

3. IMPORTANCE OF RECENCY AND COMPACTNESS INTO SPM

The sequential patterns we want to discover must satisfy not only the frequency minimum support for the whole sequence database but also the recency minimum support for the recent sequence database (i.e., the subset of the sequence database that occurs recently). The concept of recency makes the patterns quickly adapt to the latest behaviors in sequence databases. Compactness means the time span from the first item to the last item in the pattern must be no more than a given threshold Maximum Span Length. The concept of compactness ensures the discovered patterns having reasonable time spans.

If we consider only recency and compactness a huge number of cheap patterns will be generated so it is essential to consider monetary constraint. Ron Kohavi, Rajesh Parekh has introduced the concept of performing segmentation based on RFM as follows:

RFM (Recency, Frequency and Monetary)

Market segmentation is critical for a good marketing and customer relationship management program. Segmentation divides markets into customer clusters with similar needs and/or characteristics that are likely to exhibit similar purchasing behaviors. With proper market segmentation, enterprises can arrange the right products, services and resources to a target customer cluster and build a close relationship with them. A critical issue to successful market segmentation is the selection of the segmentation variables RFM is easy to use and can generally be implemented very quickly.

It is a method that managers and decision makers can understand. The major benefit of the RFM methodology is that it allows marketers to test marketing campaigns to smaller segments of customers, and direct larger campaigns only towards those customer segments that are predicted to respond profitably.

Due to the usefulness of RFM in marketing, some data mining techniques have been conducted in RFM. In general, the most common seen techniques are: (1) cluster analysis and (2) classification. The cluster analysis segments the customers into a number of clusters with similar characteristics from the RFM point of view. Regarding to classification, RFM attributes are used for classifying customers to different categories of customer value and they are also used to classify unseen cases by using classification techniques.

Benefit of the RFM Methodology

Market segmentation is critical for a good marketing and customer relationship management program. Segmentation divides markets into customer clusters with similar needs and/or characteristics that are likely to exhibit similar purchasing behaviors. With proper market segmentation, enterprises can arrange the right products, services and resources to a target customer cluster and build a close relationship with them. A critical issue to successful market segmentation is the selection of the segmentation variables RFM is easy to use and can generally be implemented very quickly. Furthermore, it is a method that managers and decision makers can understand. The major benefit of the RFM methodology is that it allows marketers to test marketing campaigns to smaller segments of customers, and direct larger campaigns only towards those customer segments that are predicted to respond profitably.

Noticeably, all the previous researches consider only the concept of frequency. In other words, if a pattern is not frequent, then it will not be found. In the proposed system except the frequency constraint two additional constraints, recency constraint and monetary constraint are considered. The proposed system applies the concept of RFM (Recency, Frequency and Monetary) to the sequential mining process to discover the RFM-patterns. By these three indexes, the company can easily classify their customers, and give the individual customer a particular score according to these three indexes evaluated. Furthermore, they can help the company to determine which customers are more important, and do the personalization marketing to these customers.

Traditional sequential pattern mining only distinguishes whether a pattern appears or not, while RFM pattern mining approach not only determines the existence of a pattern but also checks whether it conforms to the recency and the monetary constraints.

Proposed System

The proposed algorithm improves the performance of finding frequent sequences by applying modified candidate

generation and support counting approach. Traditional sequential pattern mining only distinguishes whether a pattern appears or not, while RFM pattern mining approach not only determines the existence of a pattern but also checks whether it satisfies the recency and the monetary constraints. Some key concepts required to understand the proposed algorithm is as follows:

Recency Constraint: Recency constraint is specified by giving a recency minimum support (r_minsup), which is the number of days away from the starting date of the sequence database. For example, if our sequence database is from 27/12/2007 to 31/12/2008 and if we set r_minsup = 200 then the recency constraint ensures that the last transaction of the discovered pattern must occur after 27/12/2007+200 days. In other words, suppose the discovered pattern is < (a), (bc)>, which means "after buying item a, the customer returns to buy item b and item c". Then, the transaction in the sequence that buys item b and item c must satisfy recency constraint.

Monetary Constraint: Monetary constraint is specified by giving monetary minimum support (m_minsup). It ensures that the total value of the discovered pattern must be greater than m_minsup. Suppose the pattern is <(a), (bc)>. Then we can say that a sequence satisfies this pattern with respect to the monetary constraint, if we can find an occurrence of pattern <(a), (bc)> in this data sequence whose total value must be greater than m_minsup.

Frequency Constraint: Frequency constraint is specified by giving frequency minimum support (f_minsup). The frequency of a pattern is the percentage of sequences in database that satisfy the recency constraint and monetary constraint. And a pattern could be output as an RFM-pattern if its frequency is greater than f_minsup.

f-pattern(LI_k^f), rf-pattern(LI_k^{rf}), rfm-pattern(LI_k^{rfm}) of Length k: Let $B = <I_1 I_2 ... I_S>$ be a sequence of itemsets. If the percentage of data sequences in database containing B as a subsequence, called f-support, is no less than f_minsup, B is called an f-pattern. B is called an rf-pattern if the percentage of data sequences in database containing B as a recent subsequence (which satisfies recency constraint), called rf-support, is no less than f_minsup. Finally, B is called an rfm-pattern if the percentage of data sequences in database containing B as a recent monetary subsequence (which satisfies recency and monetary constraints), called rfm-support, is no less than f_minsup.

Representation of Data Sequence: In a traditional approach data sequence is represented as a list of itemsets ordered by transaction time. But for the proposed approach the data sequence is represented in another way : a data-sequence A is represented as $<(a_1, t_1, m_1), (a_2, t_2, m_2), ..., (a_n, t_n, m_n)>$, where (a_j, t_j, m_j) means that item a_j is purchased at time t_j with total value m_j, $1 \le j \le n$, and $t_{j-1} \le t_j$ for $2 \le j \le n$. In the data-sequence, if items occur at the same time, they are ordered alphabetically.

The Proposed Algorithm:

Input : Data-sequence database

The threshold of support given by the user (f_minsup)

The threshold of recency given by the user (r_minsup)

The threshold of monetary given by the user (m_minsup)

Output: The set of all large RFM-patterns

Method:

1. Scan the sequence database to count f-support,rf-support and rfm-support for each itemset.
2. Generate 1-frequent patterns (LI_1^f, LI_1^{rf}, LI_1^{rfm}) which satisfies the user specified support threshold (f_minsup) using apriori process.
3. if k = 2 then merge frequent patterns (LI_1^f, LI_1^{rf}) to generate candidate sequence (CI_2).
4. if k > 2 then merge frequent patterns (LI_{k-1}^{rf}, LI_{k-1}^{rf}) which have same k-2 postfix to generate candidate sequence (CI_k).
5. Build inverse candidate tree by inserting the itemsets in all candidate patterns of candidate sequence (CI_k) into an empty tree in reverse order.
6. Take one by one data sequences from sequence database and traverse the tree to match different subsequences of data sequence with all candidate patterns of current candidate sequence.
7. If matched subsequence satisfies both recency(r) and monetary(m) constraint then increase rf-support and rfm-support for current candidate pattern by one.
8. Eliminate candidate patterns that have small support count (rf-support and rfm-support) compared to user specified support threshold (f_minsup) and find frequent patterns (LI_k^{rf}, LI_k^{rfm}).
9. Repeat through step 4 until no more frequent pattern (LI_k^{rf}) is found.
10. Output the set of all large RFM-patterns.

The Constrained Prefixspan Algorithm

Algorithm : Constrained PrefixSpan

Input: A Sequence Database

Minimum Support

Minimum Confidence

Maximum Gap (Max_gap)

Maximum

Compactness(Max_compact)

Recency Support (R_sup)

Output: The complete set of Sequential patterns, Emerging patterns

Method:

1. Scan database to find length-1 sequential patterns.
2. Generate pseudo sequence database by removing items of length-1 from sequence database which are not frequent.
3. Divide complete set of sequential patterns into different subsets according to set of length-1 sequential patterns (prefix).
4. Construct projected database for each prefix.
5. Find frequent item b from each projected database which satisfies Max_gap.
6. For each frequent item b append it to prefix to generate new prefix in such a way that
   (a) b can be assembled to the last element of prefix to form a sequential pattern or;
   (b) <b> can be appended to prefix to form a sequential pattern.
7. Recursively generate projected database for each new prefix which satisfies Max_compact and mine it to find local frequent patterns.
8. Merge local frequent patterns which satisfies R_sup to generate global frequent patterns.
9. Output the complete set of sequential patterns.

   For generating Emerging Patterns

10. Supply the new sequence database file which contains updated data.
11. Generate global frequent patterns from the new sequence database file.
12. Calculate the difference between frequent patterns in both the files which have drastic change in support count are the emerging patterns.

Dataset Format

For the purpose of implementing the Constrained Prefixspan algorithm for finding sequential patterns, the dataset generated by illimine synthetic data generator is used. The dataset is provided in text format.

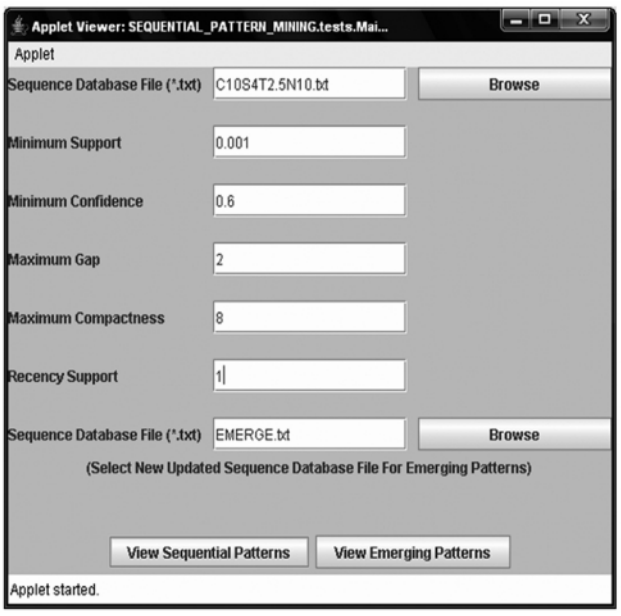
Table
The Parameters of Synthetic Dataset

Parameters	Description
|C|	No. of Customers
|S|	Average number of transactions per sequence
|T|	Average number of items per transaction
|N|	No. of distinct items

The Constrained Prefixspan algorithm uses different dataset format than the dataset generated by illimine generator. So, by java coding above dataset format has been changed so that it can be directly applied to Constrained Prefixspan algorithm. The Constrained Prefixspan algorithm applies gap, compactness, recency and frequency constraints on input dataset to generate sequential patterns which are based on timestamp at which each transaction occurred.

Implementation of Constrained Prefixspan Approach

The Constrained Prefixspan approach produces sequential patterns which satisfies frequency, gap, compactness and recency constrains and minimum confidence. The original Prefixspan algorithm only applies frequency constraints to discover sequential patterns from sequence database. To apply above constraints in sequential pattern mining process changes are done in original Prefixspan algorithm. The system is implemented as an object oriented program using java programming language.The new approach also produces the emerging patterns by using the recent sequence database.



Screen 1

Fig.: An algorithm for Discovering RFM Sequential Patterns

Screen 2

SEQUENTIAL PATTERN TABLE

SEQUENCE	SUPPORT	CONFIDENCE
(t=1, 8068)	0.003	
(t=1, 8068) => (t=2, 5094 9549)	0.0022	0.7143
(t=1, 8068) => (t=2, 5094)	0.0022	0.7143
(t=1, 8068) => (t=2, 705 5094 9549)	0.0022	0.7143
(t=1, 8068) => (t=2, 705 5094)	0.0022	0.7143
(t=1, 8068) => (t=2, 705 9549)	0.0022	0.7143
(t=1, 8068) => (t=2, 705)	0.0022	0.7143
(t=1, 8068) => (t=2, 9549)	0.0022	0.7143
(t=1, 8068) => (t=3, 5329)	0.0019	0.619
(t=1, 8068)(t=2, 5094 9549) => (t=3, 5329)	0.0016	0.7333
(t=1, 8068)(t=2, 5094 9549) => (t=4, 4438)	0.0014	0.6667
(t=1, 8068)(t=2, 5094 9549)(t=3, 5329) => (t=4, 4438)	0.0014	0.9091
(t=1, 8068)(t=2, 5094) => (t=3, 5329)	0.0016	0.7333
(t=1, 8068)(t=2, 5094) => (t=4, 4438)	0.0014	0.6667
(t=1, 8068)(t=2, 5094)(t=3, 5329) => (t=4, 4438)	0.0014	0.9091
(t=1, 8068)(t=2, 705 5094 9549) => (t=3, 5329)	0.0016	0.7333
(t=1, 8068)(t=2, 705 5094 9549) => (t=4, 4438)	0.0014	0.6667
(t=1, 8068)(t=2, 705 5094 9549)(t=3, 5329) => (t=4, 4438)	0.0014	0.9091
(t=1, 8068)(t=2, 705 5094) => (t=3, 5329)	0.0016	0.7333
(t=1, 8068)(t=2, 705 5094) => (t=4, 4438)	0.0014	0.6667
(t=1, 8068)(t=2, 705 5094)(t=3, 5329) => (t=4, 4438)	0.0014	0.9091
(t=1, 8068)(t=2, 705 9549) => (t=3, 5329)	0.0016	0.7333
(t=1, 8068)(t=2, 705 9549) => (t=4, 4438)	0.0014	0.6667
(t=1, 8068)(t=2, 705 9549)(t=3, 5329) => (t=4, 4438)	0.0014	0.9091
(t=1, 8068)(t=2, 705) => (t=3, 5329)	0.0016	0.7333
(t=1, 8068)(t=2, 705) => (t=4, 4438)	0.0014	0.6667
(t=1, 8068)(t=2, 705)(t=3, 5329) => (t=4, 4438)	0.0014	0.9091
(t=1, 8068)(t=2, 9549) => (t=3, 5329)	0.0016	0.7333

Screen 3

EMERGING PATTERN VIEW

EMERGING PATTERN	SUPPORT
(t=1, 1704)	0.0047
(t=1, 183)	0.0037
(t=1, 1877)	0.0043
(t=1, 193)	0.0038
(t=1, 2236)	0.0051
(t=1, 3366)	0.0031
(t=1, 3627)	0.0032
(t=1, 3920)	0.0035
(t=1, 434 8696)	0.0032
(t=1, 4671)	0.0039
(t=1, 5483)	0.0037
(t=1, 5552)	0.004
(t=1, 5779)	0.0039
(t=1, 601)	0.0057
(t=1, 6106)	0.004
(t=1, 624)	0.0046
(t=1, 6692)	0.0032
(t=1, 6825 7342)	0.003
(t=1, 6825)	0.003
(t=1, 6910 7354)	0.0044
(t=1, 6910)	0.0044
(t=1, 8004)	0.004
(t=1, 8693)	0.0034
(t=1, 8731)	0.0062
(t=1, 9384)	0.0057
(t=1, 9605)	0.0051
(t=1, 9901)	0.0041
(t=1, 991)	0.0049

Screen 4

Fig. : An algorithm for Discovering Sequential Patterns with Constraints

4. CONCLUSION

The proposed algorithm modifies traditional sequential pattern mining algorithm GSP (Apriori-based), so that, except the frequency it also considers two additional constraints, the last purchasing time (Recency) and purchasing money (Monetary) to discover the RFM(Recent, Frequent and Monetary) patterns. The advantage of considering these two additional factors is that this can ensure all patterns are recently active and profitable.

REFERENCES

[1] R. Agrawal and R. Srikant, "Mining Sequential Patterns", In Proceedings of the 1995 International Conference on Data Engineering, pp. 3-14, 1995.

[2] R. Agrawal and R. Srikant, "Mining Sequential Patterns: generalizations and Performance Improvements", In Proceedings of the 5th International Conference on Extending Database Technology, pp. 3-17, Avignon, France, 1996.

[3] Yen-Liang Chen, Ya-Han Hu, "The Consideration of Recency and Compactness in Sequential Pattern Mining", In Proceedings of the Second Workshop on Knowledge Economy and Electronic Commerce, 42, Iss. 2, pp. 1203-1215, 2006.

[4] Jian Pei, Jiawei Han, Wei Wang, "Constraint-based Sequential Pattern Mining : the Pattern Growth Methods", J Intell Inf Syst, 28, No.2, pp. 133 –160, 2007.

[5] Y. L. Chen, M. C. Chiang, and M. T. Kao, "Discovering Time-interval Sequential Patterns in Sequence Databases", Expert Systems with Applications, 25, No. 3, pp. 343-354, 2003.

[6] Ming-Yen Lin and Suh-Yin Lee, "Incremental Update on Sequential Patterns in Large Databases by Implicit Merging and Efficient Counting", Information Systems, 29, No. 5, pp. 385-404, 2004.

[7] M. N. Garofalakis, R. Rastogi, K. Shim, "SPIRIT: Sequential Pattern Mining with Regular Expression Constraints", In Proceedings of 25th VLDB Conference, pp. 223-234, San Francisco, California, 1999.

[8] C. Antunes, A. L. Oliveira, "Generalization of Pattern-growth Methods for Sequential Pattern Mining with Gap Constraints", Machine Learning and Data Mining in Pattern Recognition, Third International Conference, MLDM 2003, Leipzig, Germany, July 5-7, 2003, Proceedings 2003.

[9] M. J. Zaki, "SPADE: an Efficient Algorithm for Mining Frequent Sequences", Machine Learning Journal, 42, Iss. (1-2), pp. 31-60, 2001.

[10] Show-Jane Yen and Yue-Shi Lee, "Mining Sequential Patterns with Item Constraints", DaWaK 2004: Data Warehousing and Knowledge Discovery: International Conference on Data Warehousing and Knowledge Discovery, Zaragoza, ESPAGNE, 3181, pp. 381-390, 2004.

[11] Jian Pei, Jiawei Han and Helen Pinto, "PrefixSpan: Mining Sequential Patterns Efficiently by Prefix-projected Pattern Growth", In Proceedings of 12th International Conference

on Data Engineering, pp. 215-224, Heidelberg, Germany, 2001.

[12] Helen Pinto, and Jiawei Han, "Multidimensional Sequential Pattern Mining", In Proceedings of the 10th International Conference on Information and Knowledge Management, pp 81-88, Atlanta, Georgia, USA, 2001.

[13] Ron Kohavi, Rajesh Parekh, "Visualizing RFM Segmentation", In Proceedings of the Fourth SIAM International Conference on Data Mining, San Mateo, CA, 2004.