IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

A Review On Temporal Co-occurrence Pattern Extraction Techniques

Pratiksha P. Ghule

Student

Department Of Computer Engineering,

Matoshri College Of Engineering & Research Centre, Eklhare, Nashik, India

Prof. Dr. Swati Bhavsar

Associate Professor

Department Of Computer Engineering,

Matoshri College Of Engineering & Research Centre, Eklhare, Nashik, India

Abstract : Frequent itemset mining is important task in data mining domain. This is applicable in variety of applications such as market-basket analysis, browsing history analysis, transaction record analysis, etc. Lot of work has been done in the domain of co-occurrence pattern extraction and association rule mining. The Complete dataset is given as an input to the system and frequent itemset are extracted from the dataset. Temporal data is the data containing timestamp information. Along with the co-occurrence information, timestamp information is also helpful in extracting time specific pattern extraction. This strategy helps to analyze lifespan of itemset, peak period of purchase, seasonal products etc. This work includes the analysis of existing system for temporal extraction of frequent patterns, its methodologies and limitations. After analyzing the existing work a new approach is proposed to overcome the existing system problem.

IndexTerms **- co-occurrence patterns, time cube, Temporal analysis, apriori, multithreaded application, frequent patterns**

I. INTRODUCTION

In data mining co-occurrence itemset extraction is important branch. For co-occurrence pattern extraction generally transaction dataset is used. Transaction records contain supermarket data selling information. Each record represents the order placed by a customer. Analysis of such order records, frequently sold items can be extracted. The frequently occurred patterns are called as co-occurrence patterns. For example: bread and butter are occurred in multiple transactions at a time hence it{bread, butter} is called as itemset. If this itemset occurs frequently then this is called as co-occurrence pattern. In co-occurrence pattern extraction the order of purchase is not important. The co-occurrence pattern extraction helps to make decision in marketing strategies such as : product placement, product promotion, offer announcement, etc.

For finding co-occurrence patterns, minimum support value should be defined first. Based on the itemset occurrence, the support value of each itemset is calculated. The support value is calculated based on the number of transaction in which itemset belongs. If the itemset follows the minimum support constraint then the item is called as frequent itemset.

The co-occurrence pattern extraction technique can also be applied in variety of domains such as: bank transaction history analysis, intrusion detection, browsing link history, medical records, bioinformatics etc.

Generally, the dataset is stored with timestamp information. Such dataset with time information is called as temporal dataset. The transaction records vary periodically hence pattern extraction should be done by considering not only the occurrence count but also the temporal information.

System is proposed by considering some issues of previous approaches. The record with time stamp entries is called temporal data. Sequential association rules, time interval association rules, calendar specific interval rules are various frequent item set mining technique based on temporal information

This system is designed to mine frequent item set from temporal data. Also as contribution module is added that reduces the time of this frequent itemset mining from static and temporal dataset. Following are the phases of the system execution.

1. Phase of set up

In this phase required transnational dataset is downloaded. If it is not temporal one , it is converted in temporal dataset by adding date time for each transactions and make if temporal one for execution as per requirement. Also by considering slots (day , hour , month) different data cubes are generated.\\

2. Phase of execution

In this phase data cubes are analyzed first. Any cube with low density is merged with nearest data cube. After that system expects minimum support value from end user and according to that frequent itemset mining is done using Apriori algorithm. As part of contribution, multi-threaded parallel processing is applied. This reduces time for mining frequent itemset from temporal dataset.

Following section includes the related work done in the domain of pattern extraction. In section III problem formulation is stated. In section IV, a proposed system details are discussed followed by the conclusion the paper.

II. RELATED WORK

Daichi Amagata, et. al. [2] proposes a technique for Mining Top-k Co-Occurrence Patterns. The patterns are extracted by mining multiple data streams. Streaming data is used as an input in this system. More than one data stream is handled at a time and frequent itemsets are extracted. Using this streaming data top co-occurrence patterns (top-k) are detected. For more correct predictions multiple streaming data is given as an input at the same time. With this experimental setup frequent itemsets are fetched. A CP-graph is generated from transaction records. CP-graph is generated to get better idea of transaction records. For each sliding window the CP graph is updated. Co-occurrence patterns are recognize on the basis of support value. Patterns that occur frequently more than one stream and support value is greater than minimum support value then such patterns are called as co-occurrence patterns. Minimum threshold has to be calculated to take cut-off for considerations of particular pattern and this is calculated automatically. Top K occurrence count of itemset helps to calculate this threshold value.

Xiao, et.,al[3] proposes a technique for mining association rules based on temporal information. The paper works on discovery of co-occurrence patterns and association rules from dataset. The system uses temporal dataset. Based on the temporal information, the maximum time frequent window for itemset is identified. This is the window in which the itemset support value crosses the minimum support threshold constraint. The system uses a Variable Neighborhood Search (VNS) algorithm.By mathematical modeling the time window is optimized. In total two factors for dataset are considered. One is association rules and other one is temporal information. Temporal information plays vital role in itemset finalization. Mathematical Modeling is used to optimize the time window.

L. T. Nguyen,et.,al [4] proposes a technique for mining association rules with generalization. For mining association rules, the item are arranges in hierarchical manner. The technique focuses on reducing the redundancy in association rule. While dealing with the association rules there is chance for redundancy in those rules. This puts an extra overhead on the system execution. To rectify this execution overhead due to association rules this paper proposes "generalization" which is a technique for mining association rules. In this technique hierarchy in items is determined and while performing association rules mining , item are arranged in hierarchical manner. In this hierarchical arrangement redundancy in association rules analyzed.

For mining class association rule L. T. Nguyen,,et. al. proposes a CAR-Miner algorithm[5]. This method is useful than heuristic and greedy methods. CAR-Miner algorithm is discussed in this paper over traditional greedy as well as heuristic algorithms and it is proved that this method is useful while mining class association rule. This methods work on removing noise and hence it improves the accuracy of the system. Also noise in the input can exhaust system at execution level. Hence it is considered great in this paper. For mining class association rules, system uses tree structure. Each node is decision making node for computation. Every node in a tree has some information for fast computation of support value for next candidate pattern. Candidate patterns are detected. For this Node pruning technique is used.

D. Nguyen, et.al.[6] proposes a class association rule mining technique named as CCAR. This is an extended version of Car algorithm. This technique is able to find CAR rules based on user defined minimum support threshold value and minimum confidence thresholds from a dataset. To improve efficiency of system it partition the algorithm in two phases: in preprocessing phase candidate rules are generated and candidate list is filtered in the post-processing step. Divide and conquer strategy in pre-processing and post-processing manner helps to improve efficiency. This is useful for effective memory management. Also time consumption is outperform in this.

Saleh and Masseglia [7] proposes a system to find frequent itemset on temporal data. It find the subset of dataset in which the frequent itemset occurs. This technique is useful for seasonal product purchase analysis. The system dynamically finds the period of items in a dataset. The itemset support value should be very less as compared to the other traditional algorithm. This paper claims that rather than picking up the itemsets from complete data set we can extract them from subsets within dataset. Meaningful and relevant subsets may be hidden in datasets and using time context we can extract desire itemsets. Loss of particular itemset's behavior as per season is missed when we consider complete dataset instead of considering subsets. This paper works to reduce this loss.

Progressive partitioning[8] technique proposes a solution on 2 problems: 1: Lack of exhibition period of each item 2 : lack of equitable support for each item. The system initially portioned the dataset in equal parts. And then progressively find the candidate items with the itemset size 2. Progressive Partition Miner (PPM) algorithm is used in this proposed system. Idea behind this algorithm is that , it partitioned the public exhibition dataset based on time (equal intervals) and then based on the partitioning characteristics , occurrence count of each candidate 2-itemset is calculated. Along with this task , PPM also employ the filtering of infrequent 2 itemsets.

Matthews et al. [9] proposes a generic algorithm to find association rules from temporal dataset. This technique does not require any prior knowledge for portioning the dataset. Algorithm searches each rule in rule space as well as in temporal space. The dynamic partition of datasets is generated as per each rule. Also it is the extension of already discussed or existing association rule mining procedures. Genetic algorithm is employed for searching of rule space and temporal space for synthetic dataset. In this algorithm majorly iteratively learns the rules and deals with targets in dataset in case of various difficulty levels.

The first study related to the association rules discovery is presented by Agrawal et al.[10] in 1993. In this technique 2 things are focused: Finding frequent itemsets and generating association rules. This processing is done over purchase history of large items dataset. Unlike other statistical techniques, this technique considers Association rules are closely related to frequent itemset extraction.

The cyclic pattern discovery is proposed by Ozden et al.[11]. The cyclic patterns are those patterns occurred after every regular cyclic variation over time. In this technique 2 algorithms are proposed: the sequential algorithm and interleaved algorithm. This technique extracts hourly daily, monthly , quarterly patterns. The pruning technique is applied to improve the performance of algorithm. The pattern can be called as cyclic pattern if it find in every cycle without any exception. This cyclic analysis helps in trend analysis and market forecasting.

Ramaswamy et al[12] proposes a new technique based on the technique proposed by Ozden et al. This technique uses user defined temporal patterns for rule discovery. The calendar algebra is used to process the time cycle. But this technique requires the prior knowledge of calendar data expression.

Mazaher Ghorbani et. al[1] proposes a technique for frequent itemset extraction from temporal dataset. The data is divided in number of time cube blocks. Each block is analyzed separately and frequent itemsets are extracted. The frequent itemsets are extracted based on minimum support value and cube density. The apriori algorithm is used to find frequent itemsets. This is a time consuming process because each time cube block need to be processed separately.

Paper	Description	Analysis
Daichi Amagata, et. al. [2]	Patterns that occur frequently in more than one stream and support value is greater than minimum support value then such patterns are called as co-occurrence patterns. Top K occurrence count of itemset helps to calculate this threshold value.	Unnecessary system execution overhead. Also top-k results are considered hence future frequent itemset patterns are not considered hence accuracy is missed.
Xiao, et.,al[3]	Factors such as temporal information and association rules are considered in this paper and Variable Neighborhood Search (VNS) algorithm is used.	Temporal factor is considered which is helpful for detail analysis of itemset co-occurrence. Also association rules are considered for data categorization.
L. T. Nguyen,et.,al [4]	The technique focuses on reducing the redundancy in association rule.	Redundancy in association rules is discusses and rectified majorly in this paper.
L. T. Nguyen,,et. al.[5]	CAR-Miner algorithm is discussed in this paper over traditional greedy as well as heuristic algorithms. For mining class association rules, system uses tree structure.	Association rule based mining is rectified at redundancy level and noise in data is rectified.
D. Nguyen, et.al.[6]	Class association rule mining technique named as CCAR is used which is extended version of CAR-Miner.	This is useful for effective memory management. Also time consumption is outperforming this.
Saleh and Masseglia [7]	Arbritari division of data may cause loss of particular behavior if itemset which is rectified in this paper.	Loss of particular itemset's behavior as per season is missed when we consider complete dataset instead of considering subsets. This paper works to reduce this loss
C.-H. Lee, M.-S. Chen, and C.-R. Lin [8]	PPM algorithm is used to generate candidate 2 itemsets by partitioning datasets in equal parts.	Correctness of Progressive Partition Miner algorithm is proved. Execution time for PPM execution is drastically reduced as far as other schemes an magnitude is considered.
Matthews et al. [9]	Proposes a generic algorithm to find association rules from temporal dataset. Major work is done on rule space as well as in temporal space.	With varying levels of difficulties , this iterative rule learning method fetches these target itemsets properly.
Agrawal et al.[10] in 1993	In this technique 2 things are focused: Finding frequent itemsets and generating association rules	This processing is done over purchase history of large items dataset
Ozden et al.[11].	The cyclic patterns are those patterns occurred after every regular cyclic variation over time and these are discussed in this. Sequential algorithm and interleaved algorithm are used.	This cyclic analysis helps in trend analysis and market forecasting.

Ramaswamy et al[12]	It proposes a new technique based on the technique proposed by Ozden et al.[11]. This technique uses user defined temporal patterns for rule discovery.	Drawback of this technique is that prior calendar knowledge is required.
Mazaher Ghorbani and Masoud Abessi [1]	Time cube blocks from dataset are created for analysis independently. Apriori algorithm is used to find frequent itemset.	Bit time consuming process but its accuracy overpower this thing as compared to the other existing system.

III. ANALYSIS AND PROBLEM FORMULATION

Co-occurrence pattern extraction plays important role in data mining. Lot of work has been done on static dataset. The transaction records vary periodically. The co-occurrence pattern should be extracted based on the temporal information. There is need to develop a system that fiend co-occurrence patterns on temporal dataset efficiently.

IV. PROPOSED METHODOLOGY

A. Architecture

Following figure shows the architecture of the system. Transaction dataset, BTC information and threshold values are input to the system. System finds co-occurrence patterns per time cube and also the final co-occurrence set. The apriori algorithm based mining frequent itemsets with time cubes is applied on the dataset. The system works with multithreaded execution. The multithreading execution increases system efficiency.

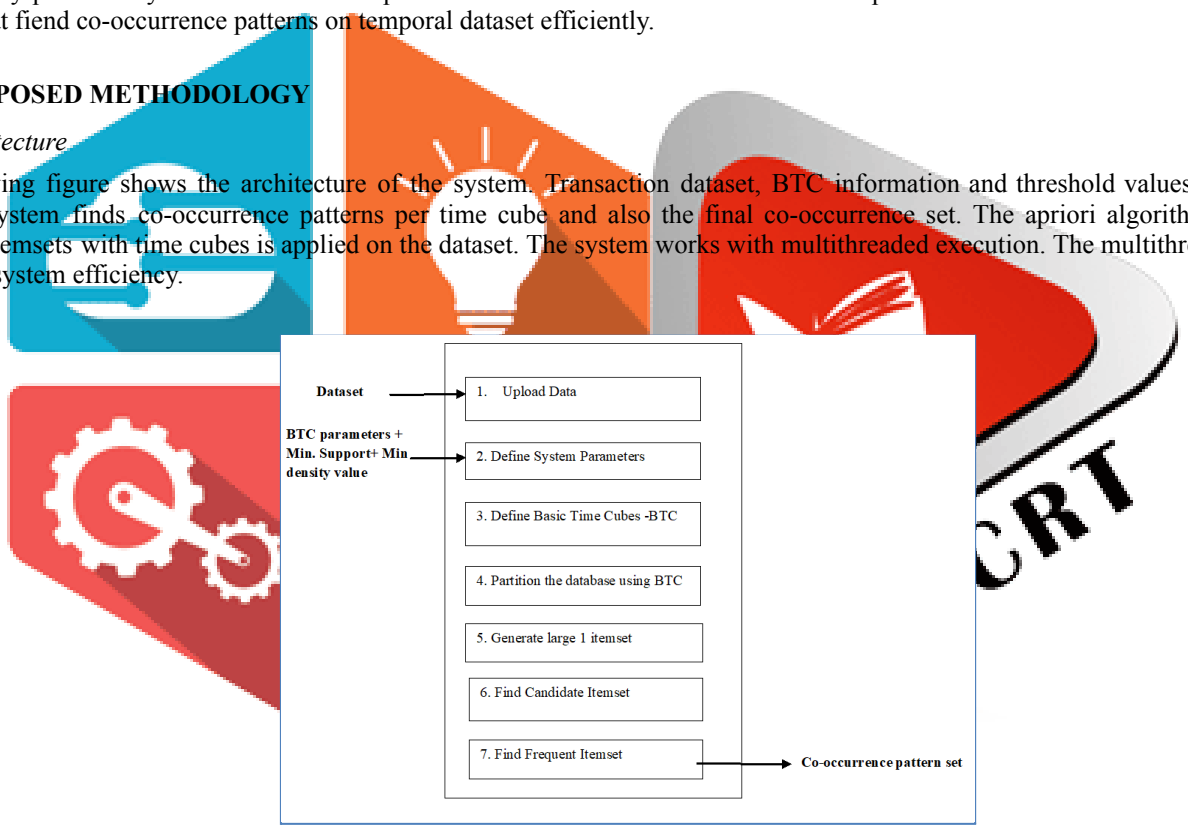

Figure 1 : System Architecture

B. System Working:

The transaction dataset with time information is partitioned in number of time cubes. User defines the time cube division parameters such as: (Hour, Month, Year), (Hour, Day, Month), (Day, Month, Year),etc. User also provides the number of partition count for each parameter. For example if user suggest to partition each parameter in 3 sections then total 27 cubes will be generated. A unique id is assign to each time cube. The input dataset is partitioned in time cube and time cube unique id is assigned to the transaction in which it belongs.

The whole dataset is partitioned in number of time cubes. Some data cube may contain large number of items whereas some data cubes contain very few items. The frequently occurred pattern extraction from data cube containing vary few items may not generate an appropriate result. A cube density constraint is introduced to measure and limit the minimum occurrence of items in a data cube. The density of time cube is calculated as:

Density = α *A (1)

Where, A is the average number of records per cube it is calculated as:

$$A = \frac{N}{N-BTC} \quad (2)$$

Where N = total number of transaction in a dataset,

N-BTC = Number of transaction in BTC

A is user defined threshold value between 0 and 1.

If the time cube density is less than A then the time cube data is merged with next time cube and revised time cubes are generated.

The large one itemset is extracted from each time cube. The large-one itemset represents the set of co-occurrence patterns. The support value for each co-occurrence pattern X is extracted using following formula:

$$\text{Support(X)} = \frac{N(X)-cube}{N-cube} \quad (3)$$

Where N(x)-cube : total number of itemsets containing itemset X in time cube TC

N-cube: Total number of transaction in time cube TC

User defines the minimum support value. If the support value of itemset X is greater than the minimum support value then the itemset is called as frequently occurred itemset. Such itemset is added in co-occurrence pattern list.

For co-occurrence pattern extraction following 2 conditions need to be satisfied:

1. The itemset should belong from a time cube whose density is greater than A
2. The itemset should have support value greater than minimum support value

The mining frequent itemset algorithm process each time cube block To improve system efficiency the parallel processing is introduced. In parallel processing more than one block is executed simultaneously. The collective result is displayed to the user.

V. CONCLUSIONS

Co-occurrence pattern extraction plays important role in data mining. Lot of work has been done on static dataset. The transaction records vary periodically. The co-occurrence pattern should be extracted based on the temporal information. The proposed system works on finding co-occurrence patterns on based on its time information. The dataset is divided in number of time cubes based on the time information. Apriori algorithm is used to find co-occurrence patterns in each cube and from the overall dataset. Density of each cube is checked t avoid over estimation problem. For efficiency improvement parallel processing is introduced. In parallel processing multiple cubes are processed simultaneously.

VI. REFERENCES

[1] Mazaher Ghorbani and Masoud Abessi, "A New Methodology for Mining Frequent Itemsets on Temporal Data", in IEEE Transactions on Engineering Management, Vol. 64, Issue. 4, pp. 566 - 573, Nov 2017

[2] Daichi Amagata, Takahiro Hara,"Mining Top-k Co-Occurrence Patterns across Multiple Streams", in IEEE Transactions on Knowledge and Data Engineering, Vol. 29, Issue 10, pp. 2249 - 2262, Oct 2017

[3] F. Benites and E. Sapozhnikova, "Hierarchical interestingness measures for association rules with generalization on both antecedent and consequent sides," Pattern Recognit. Lett., vol. 65, pp. 197–203, 2015.

[4] F. Benites and E. Sapozhnikova, "Hierarchical interestingness measures for association rules with generalization on both antecedent and consequent sides," Pattern Recognit. Lett., vol. 65, pp. 197–203, 2015.

[5] L. T. Nguyen, B.Vo, T.-P. Hong, and H. C. Thanh, "Car-miner: An efficient algorithm for mining class-association rules," Expert Syst. Appl., vol. 40, no. 6, pp. 2305–2311, 2013.

[6] D. Nguyen, B. Vo, and B. Le, "CCAR: An efficient method for mining class association rules with itemset constraints," Eng. Appl. Artif. Intell., vol. 37, pp. 115–124, 2015

[7] B. Saleh and F. Masseglia, "Discovering frequent behaviors: Time is an essential element of the context," Knowl. Inf. Syst., vol. 28, no. 2, pp. 311–331, 2011.

[8] C.-H. Lee, M.-S. Chen, and C.-R. Lin, "Progressive partition miner: An efficient algorithm for mining general temporal association rules," IEEE Trans. Knowl. Data Eng., vol. 15, no. 4, pp. 1004–1017, Jul./Aug. 2003.

[9] S. G. Matthews, M. A. Gongora, and A. A. Hopgood, "Evolving temporal association rules with genetic algorithms," in Research and Development in Intelligent Systems XXVII. New York, NY, USA: Springer, 2011, pp. 107–120.

[10] R. Agrawal, T. Imieli´nski, and A. Swami, "Mining association rules between sets of items in large databases," ACM SIGMOD Rec., vol. 22, no. 2, pp. 207–216, 1993.

[11] B.O zden, S. Ramaswamy, andA. Silberschatz, "Cyclic association rules," in Proc. IEEE 14th Int. Conf. Data Eng., 1998, pp. 412–421.

[12] S. Ramaswamy, S. Mahajan, and A. Silberschatz, "On the discovery of interesting patterns in association rules," in Proc. 24th Int. Conf. Very Large Data Bases, 1998, pp. 368–379.