International Journal of Enhanced Research in Science, Technology & Engineering
ISSN: 2319-7463, Vol. 6 Issue 7, July-2017, Impact Factor: 4.059

Literature Review on Customer Purchase Pattern Analysis using different approaches

Usha Gill[1], Yatin Chopra[2]

[1]M.Tech. Student (CSE Dept.), CBS Group of Institutions Jhajjar, Haryana
[2]Assitant Professor (CSE Dept.), CBS Group of Institutions Jhajjar, Haryana

ABSTRACT

The data mining technique has been used to predict the customer purchase pattern. Frequent item set mining has been a heart favorite theme for data mining researchers for over a decade. A large amount of literature has been dedicated to this research and tremendous progress has been made, ranging from efficient and scalable algorithms for frequent itemset mining in transaction databases to numerous research frontiers, such as sequential pattern mining, structured pattern mining, correlation mining, associative classification, and frequent pattern-based clustering, as well as their broad applications.

Keywords: data mining, predicting, customer, purchase, pattern.

INTRODUCTION

Data mining is the process of extracting hidden patterns from data. As more data is gathered, with the amount of data doubling every three years, data mining is becoming an increasingly important tool to transform this data into knowledge. It is commonly used in a wide range of applications, such as marketing, fraud detection and scientific discovery. Data mining can be applied to data sets of any size, and while it can be used to uncover hidden patterns, it cannot uncover patterns which are not already present in the data set.

Data mining extracts novel and useful knowledge from data and has become an effective analysis and decision means in corporation.

Knowledge Discovery in Databases (KDD) is an automated extraction of novel, understandable and potentially useful patterns implicitly stored in large databases, data warehouse and other massive information repositories. KDD is a multi-disciplinary field drawing work from areas including database technology, artificial intelligence, machine learning, neural networks, statistics, pattern recognition, information retrieval, high performance computing and data visualization.

Data mining is an essential step in the process of knowledge discovery in databases, in which intelligent methods are applied in order to extract patterns. Other steps in knowledge discovery process include pre-mining tasks such as data cleaning (removing noise and inconsistent data) and data integration (bringing data from multiple sources to a single location and into a common format), as well as post mining tasks such as pattern evaluation (identifying the truly interesting patterns representing knowledge) and knowledge presentation (presenting the discovered rules using visualization and knowledge representation techniques).

In data mining, **association rule learning** is a popular and well researched method for discovering interesting relations between variables in large databases. Piatetsky-Shapiro describes analyzing and presenting strong rules discovered in databases using different measures of interestingness. Based on the concept of strong rules, Agrawal et al introduced association rules for discovering regularities between products in large scale transaction data recorded by point-of-sale (POS) systems in supermarkets

The discovered knowledge can be used in many ways in corresponding applications. For example, identifying the frequently appeared sets of items in a retail database can be used to improve the decision making of merchandise placement or sales promotion. Discovering patterns of customer browsing and purchasing (from either customer records or Web traversals) may assist the modeling of user behaviors for customer retention or personalized services. Given the desired databases, whether relational, transactional, spatial, temporal, or multimedia ones, we may obtain useful information after the knowledge discovery process if appropriate mining techniques are used

A typical process of knowledge discovery in databases is illustrated in Fig. 1.

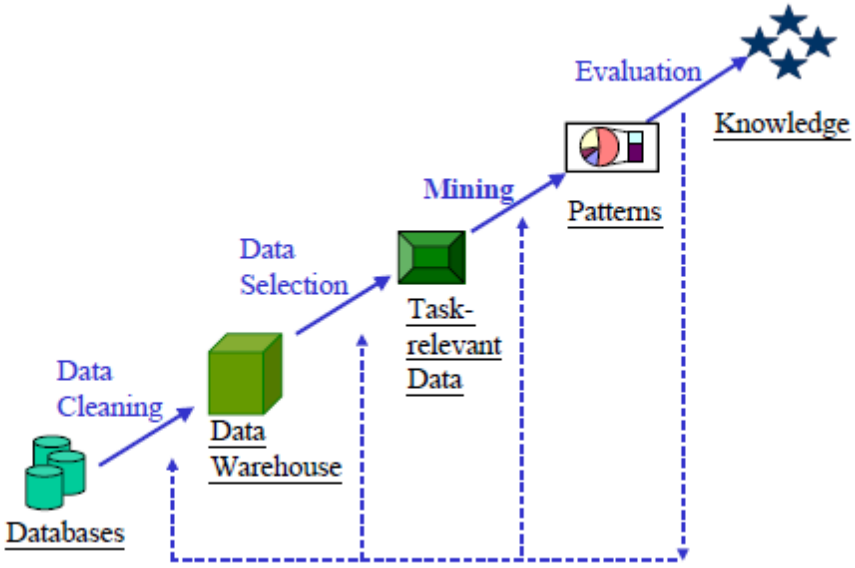

Fig. 1: The process of knowledge discovery in databases

Having the databases, relevant prior knowledge, and the goals of the application domain, the target data set is created by selecting the data required. The data cleaning in Fig. 1 may removes those 'dirty' data, e.g. data with incomplete fields, missing or wrong values, in the preprocessing stage. The 'clean' data is then reduced and/or transformed so that the data is represented by the useful features and actionable dimensions. To find the patterns of interest, the users perform the required mining functions, which include summarization/generalization of data characteristics, classification/clustering of data for future prediction, association finding for data correlation, trend and evolution analysis, etc. The discovered patterns are evaluated and presented as knowledge. The process may iterate and contain certain loops between any two steps.

Of all the mining functions in the knowledge discovering process, frequent pattern mining is to find out the frequently occurred patterns. The measure of frequent patterns is a user-specified threshold that indicates the minimum occurring frequency of the pattern. We may categorize recent studies in frequent pattern mining into the discovery of association rules and the discovery of sequential patterns. Association discovery finds closely correlated sets so that the presence of some elements in a frequent set will imply the presence of the remaining elements (in the same set). Sequential pattern discovery finds temporal associations so that not only close correlated sets but also their relationships in time are uncovered.

The National Basketball Association (NBA) is exploring a data mwining application that can be used in conjunction with image recordings of basketball games. The Advanced Scout software analyzes the movements of players to help coaches orchestrate plays and strategies. For example, an analysis of the play-by-play sheet of the game played between the New York Knicks and the Cleveland Cavaliers on January 6, 1995 reveals that when Mark Price played the Guard position, John Williams attempted four jump shots and made each one! Advanced Scout not only finds this pattern, but explains that it is interesting because it differs considerably from the average shooting percentage of 49.30% for the Cavaliers during that game. By using the NBA universal clock, a coach can automatically bring up the video clips showing each of the jump shots attempted by Williams with Price on the floor, without needing to comb through hours of video footage. Those clips show a very successful pick-and-roll play in which Price draws the Knick's defense and then finds Williams for an open jump shot.

LITERATURE REVIEW

A. Savasere et. al [1] defined the problem of finding the association rules from the database. This section introduces the basic concepts of frequent pattern mining for discovery of interesting associations and correlations between itemsets in transactional and relational database.

Alexandra's Nanopoulos et al [2] proposed a web mining method based on the concept of web perfecting. It helped in the decreasing the user latency perception ratio.

Mathis Gerry et al [3] proposed three different web mining approaches. These three methods are based on association rules, frequent sequences, and frequent generalized sequences. The authors have developed and implemented the algorithms for all three methods. Association rule learning [10] is a very common learning method from discovery of useful patterns from data & also for representing the useful patterns in form of a rule.

Agrawal. R [4] is inspired to Closet, thus using the same divide et impera approach and same FP-tree data structure. What makes FP-Close different from other CFIM algorithms is the application of the projecting approach to the historical collection of closed frequent itemsets. Not only a small dataset is associated to each node of the tree, but also a pruned subset of the closed itemsets mined so far is forged and used for duplicate detection. Indeed, this technique is called progressive focusing and it was introduced by [10] for mining maximal frequent itemsets. Together with other optimizations, this truly provides dramatic speed-up, making FP-Close order of magnitudes faster than Charm and Closet, and also making it worth to be celebrated as the fastest algorithm at the FIMI workshop 2003 [11].

Chi et al. [5] propose an algorithm called *Moment* for mining frequent closed itemsets over data streams. It uses a *CET Tree (Closed Enumerate Tree)* to maintain the main information of itemsets. Each node in CET Tree represents an itemset with different node type. Some nodes in CET Tree are not closed so that there are still some redundant nodes in CET Tree. *Moment* must maintain huge CET nodes for a frequent closed itemset. Chi et al. indicated that the ratio of CET nodes for a closed itemsets is about 20:1.

G. Pradeepini and S. Jyothi [6] have proposed algorithm called Tree-based incremental Association rule mining (TIARM) algorithm. This algorithm has two different mechanisms. First, is to generate INC-tree which is more enhanced than FP-tree to make tree more compact in nature. Second, TIARM is applied on INC-tree to discover frequent patterns of different sizes. The process of generating INCtree is same as that of the FP-tree with single pass processing. By using conditional pattern base and FP-tree, frequent patterns are generated without candidate itemset.

Liu Jian-ping et al [7] present an algorithm called FUFPtree based incremental association rule mining algorithm (Pre-FP). It is based FUFP [12] [14] (Fast Updated Frequent Pattern) concept. The major idea of FUFP is re-use of previously mine frequent items to update with incremental database. It reduces number of candidate set in updating process. All the links in FUFP are bidirectional where in FP-tree links are only in single direction. Advantage of bidirectional link is easy to add remove child node without much reconstruction. This FUFP structure is used as input to the Pre-large, which gives positive count difference whenever small amount of data is added to original database. It deals with change in database in case of inserting new transaction.

Chowdhury Farhan Ahmed et al. [8] have proposed two Single-pass incremental and interactive frequent itemsets mining algorithms with single database scan. One is weight in ascending order (i.e. IWFPwa) in which each item is having specific weight (different degree of importance). In this algorithm the given weight of items are used to calculate support of items in the database. Those weights are sorted in ascending order with highest weight in bottom this leads to database size reduction. This compressed structure is used to build FP-tree and then FP-growth algorithm is applied to discover frequent pattern. Another algorithm is based on frequency by arranging it in descending order (i.e. IWFPfd). The main advantage of this algorithm is prefix sharing of node [9] with compact structure of the tree. Numbers of nodes are less as compared to the previous method which saves memory space.

Siqing Shan et al. [10] have presented Incremental Association Rules Mining method based on Continuous Incremental Updating Technique. Transaction Amalgation Algorithm is used to merge the transaction in transaction database based on quantity present in transaction in descending order. That reduces the overall size of the database drastically saves memory space. T-tree algorithm is applied on these database which works as FP-tree. Finally T-tree is given as input to the FP-growth algorithm to discover frequent pattern.

D. Kerana Hanirex et. al [11] have proposed clustering based incremental algorithm to discover Frequent Patterns. The partitioning algorithm has proposed to generate cluster. Then Improved Apriori Algorithm [21] is applied to generate frequent patterns. If pattern is frequent then it is present in any of the cluster. Whenever new transaction is added to the database it treated as new cluster. Again Improved Apriori algorithm is applied to discover newly frequent pattern in incremental database. This algorithm has better efficiency than previous Apriori algorithm by reducing memory space and number of passes.

Liu Han-bing, [12] has proposed Incremental Frequent Pattern mining algorithm based on AprioriTidList Algorithm. This algorithm also improves Apriori performance by pruning transaction. It requires only one database scan which make it more efficient. It scans a database and creates a Tid List .It does not uses whole database to count support value instead it

consider particular large item in transaction with identifier TID. If transaction does not contain that large item then that transaction is deleted which reduces database size drastically. Tid list of Item „I" contain list of all the transaction in which I is present.

Shih-Sheng Chen et al. [13] have proposed a method for discovery of frequent periodic pattern using multiple minimum supports. This very efficient approach to find frequent pattern because it is based on multiple minimum support based on real time event. All the items in the transactions are arranged according to their MIS (Minimum Item Support). It does not hold downward closure property instead it uses sorted closure property based on ascending order. Then it uses PFP [15] (Periodic Frequent Pattern) whose construction is same as that of the FP-tree. Finally, PFP-growth algorithm is applied which is same as that FPgrowth and conditional pattern base is used to discover frequent pattern.

ASSOCIATION AND CORRELATIONS BETWEEN ALGORITHMS

This section introduces the basic concepts of frequent pattern mining for discovery of interesting associations and correlations between itemsets in transactional and relational database. Association rule mining can be defined formally as follows:

Association rule is an implication of the form X -> Y where X,Y subset of I are the sets of items called Item sets and $X \cap Y = \Phi$. Association rules show attributes value conditions that occur frequently together in a given dataset. A commonly used example of association rule mining is Market Basket Analysis [2]. We use a small example from the supermarket domain. The set of items is-

I = {Milk, Bread, Butter, Beer}

A rule for the shopping market could be **{Butter, Bread} => {Milk}** meaning that if butter and bread are bought, customers also buy milk. For example, data are collected using bar-code scanners in supermarkets. Such shopping market databases consist of a large number of transaction records. Each record lists all items bought by a customer on a single purchase transaction. Managers would be interested to know if certain groups of items are consistently purchased together. They could use this data for adjusting store layouts (placing items optimally with respect to each other), for cross-selling, for promotions, for catalog design and to identify customer segments based on buying patterns.

Association rules provide information in the form of "if-then" statements. These rules are computed from the data and, unlike the if-then rules of logic, association rules are probabilistic in nature. If 90% of transactions that purchase bread and butter, then also purchase milk.

Antecedent: bread and butter

Consequent: milk

Confidence factor: 90%

In addition to the antecedent (the "if" part) and the consequent (the "then" part), an association rule has two numbers that express the degree of uncertainty about the rule. In association analysis the antecedent and consequent are sets of items (called item sets) that are disjoint (do not have any items in common).

Support for an association rule X->Y is the percentage of transaction in database that contains X U Y .The other number is known as the **Confidence** of the rule. Confidence or Strength for an association rule X U Y is the ratio of number of transactions that contains X U Y to number of transaction that contains X is an item set (or a pattern) is frequent if its support is equal to or more than a user specified minimum support (a statement of generality of the discovered association rules). Association rule mining is to identify all rules meeting user-specified constraints such as minimum support and minimum confidence (a statement of predictive ability of the discovered rules). One key step of association mining is frequent item set (pattern) mining, which is to mine all item sets satisfying user specified minimum support. [8]

However a large number of these rules will be pruned after applying the support and confidence thresholds. Therefore the previous computations will be wasted. To avoid this problem and to improve the performance of the rule discovery algorithm, mining association rules may be decomposed into two phases:

1. Discover the large itemsets, i.e., the sets of items that have transaction support above a predetermined minimum threshold known as frequent Itemsets.

2. Use the large itemsets to generate the association rules for the database that have confidence above a predetermined minimum threshold.

The overall performance of mining association rules is determined primarily by the first step. The second step is easy. After the large itemsets are identified, the corresponding association rules can be derived in straightforward manner. Our main consideration of the paper is First step i.e. to find the extraction of frequent itemsets.

FP-Growth Algorithm

The most popular frequent itemset mining called the FP-Growth algorithm was introduced by [5]. The main aim of this algorithm was to remove the bottlenecks of the Apriori-Algorithm in generating and testing candidate set. The problem of Apriori algorithm was dealt with, by introducing a novel, compact data structure, called frequent pattern tree, or FP-tree then based on this structure an FP-tree-based pattern fragment growth method was developed. FP-growth uses a combination of the vertical and horizontal database layout to store the database in main memory. Instead of storing the cover for every item in the database, it stores the actual transactions from the database in a tree structure and every item has a linked list going through all transactions that contain that item. This new data structure is denoted by FP-tree (Frequent-Pattern tree) [4]. Essentially, all transactions are stored in a tree data structure.

Broglet's FP-Growth

Broglet implemented an efficient FP-Growth[1] algorithm using C Language. The FP-growth in his implementation preprocesses the transaction database according to [1] is as follows:

1.In an initial scan the frequencies of the items (support of single element item sets) are determined. 2. All infrequent items, that is, all items that appear in fewer transactions than a user-specified minimum number are discarded from the transactions, since, obviously, they can never be part of a frequent item set. 3. The items in each transaction are sorted, so that they are in descending order with respect to their frequency in the database.

PROPOSED ALGORITHM

Step 1: Start
Step 2: Transaction Data Set & Minimum Support Threshold
Step 3: First the algorithm scans the transaction data base and calculates the support of each single size item.
Step 4: In this step, the transaction data base is transformed into a new compressed data structure based table by pruning of all those items from the transaction database, whose support is lesser then the minimum support threshold because they will not appear in any frequent patterns.
Step 5: Call Algorithm recursively to generate bigger frequent patterns by using the union or expansion of lower size items

Example: Given a table transaction data base and also the minimum support threshold is 3 There are 10 transactions with their transaction id, list of item with their item count.

Table 1: Transaction Data Base

TID	List of Items
1	MILK, BREAD, TOAST
2	BREAD, JAM , TEA
3	BREAD, BUTTER
4	MILK, BREAD, JAM
5	MILK, BUTTER
6	BREAD, BUTTER
7	MILK, BUTTER
8	MILK, BREAD, BUTTER, TOAST
9	MILK, BREAD, BUTTER
10	SUGAR

Scan the transactional Database, D for count of each Candidate items shown in table 2:

Table 2 : Of item set after eliminate item below minimum support

Item Set	Support Count
MILK	6
BREAD	7
BUTTER	6
JAM	2
TOAST	2
SUGAR	1
TEA	1

Table 3 : Data Base after reducing the transaction which does not contain any frequent item (TID 10 reduced)

TID	List of Items
1	MILK, BREAD, TOAST
2	BREAD, JAM,TEA
3	BREAD, BUTTER
4	MILK, BREAD, JAM
5	MILK, BUTTER
6	BREAD, BUTTER
7	MILK, BUTTER
8	MILK, BREAD, BUTTER, TOAST
9	MILK, BREAD, BUTTER

Table 4: infrequent items eliminated from transactions

TID	List of Items
1	MILK, BREAD
2	BREAD
3	BREAD, BUTTER
4	MILK, BREAD
5	MILK, BUTTER
6	BREAD, BUTTER
7	MILK, BUTTER
8	MILK, BREAD, BUTTER
9	MILK, BREAD, BUTTER

Now we will use size 1 frequent items to generate larger size frequent items

Size 1 frequent items are:

MILK
BREAD
BUTTER

First we generate candidates of size 2 as

(MILK, BREAD)
(MILK, BUTTER)
(BREAD, BUTTER)

Now we calculate the count of each candidate by using the new data base

(MILK, BREAD)- 4

(MILK, BUTTER)- 4

(BREAD, BUTTER)- 4

THESE ALL ARE FREQUENT

NOW WE GENERATE CANDIDATES OF SIZE 3 AS FOLLOWS:

(MILK, BREAD, BUTTER)

NOW WE CALCULATE COUNT USING NEW DATA BASE

(MILK, BREAD, BUTTER)- 02

IT IS INFREQUENT.

So, there is no size 3 frequent item found. Therefore the algorithm halts

CONCLUSION

In this paper, we surveyed the list of existing web mining techniques. We restricted ourselves to the classic web mining problem. It is the generation of all frequent item sets that exists in market basket like data with respect to minimal thresholds for support & confidence. Frequent item set mining is crucial for association rule mining. We have evaluated the performance of our proposed algorithm. It is fast. Also it is taking less main memory for computation in comparison to previous algorithm.

FUTURE WORK

- More compact data structure can be proposed to reduce space consumption
- Our proposed algorithm works for the normal data set. The same algorithm can be extended to work for the uncertain data set.
- One limitation though data mining can help reveal patterns and relationships, it does not tell the user the value or significance of these patterns. It does not tell the users which patterns are sensitive and which are not.
- It can be said that software privacy failures can be direct result of one or more of the following points that are taken from risk management:

REFERENCES

[1]. A. Savasere, E. Omiecinski, and S. Navathe. “An efficient algorithm for mining association rules in large databases”. In Proc. Int’l Conf. Very Large Data Bases (VLDB), Sept. 1995, pages 432–443.

[2]. Aggrawal.R, Imielinski.t, Swami.A. "Mining Association Rules between Sets of Items in Large Databases". In Proc. Int'l Conf. of the 1993 ACM SIGMOD Conference Washington DC, USA.
[3]. Agrawal.R and Srikant.R. "Fast algorithms for mining association rules". In Proc. Int'l Conf. Very Large Data Bases (VLDB), Sept. 1994, pages 487–499.
[4]. Brin.S, Motwani. R, Ullman. J.D, and S. Tsur. "Dynamic itemset counting and implication rules for market basket analysis". In Proc. ACM-SIGMOD Int'l Conf. Management of Data (SIGMOD), May 1997, pages 255–264.
[5]. C. Borgelt. "An Implementation of the FP- growth Algorithm". Proc. Workshop Open Software for Data Mining, 1–5.ACMPress, New York, NY, USA 2005.
[6]. Han.J, Pei.J, and Yin. Y. "Mining frequent patterns without candidate generation". In Proc. ACM-SIGMOD Int'l Conf. Management of Data (SIGMOD), 2000
[7]. Park. J. S, M.S. Chen, P.S. Yu. "An effective hash-based algorithm for mining association rules". In Proc. ACM-SIGMOD Int'l Conf. Management of Data (SIGMOD), San Jose, CA, May 1995, pages 175–186.
[8]. [8] Pei.J, Han.J, Lu.H, Nishio.S. Tang. S. and Yang. D. "H-mine: Hyper-structure mining of frequent patterns in large databases". In Proc. Int'l Conf. Data Mining (ICDM), November 2001.
[9]. C.Borgelt. "Efficient Implementations of Apriori and Eclat". In Proc. 1st IEEE ICDM Workshop on Frequent Item Set Mining Implementations, CEUR Workshop Proceedings 90, Aachen, Germany 2003.
[10]. Toivonen.H. "Sampling large databases for association rules". In Proc. Int'l Conf. Very Large Data Bases (VLDB), Sept. 1996, Bombay, India, pages 134–145.
[11]. Nizar R.Mabrouken, C.I.Ezeife. Taxonomy of Sequential Pattern Mining Algorithm". In Proc. in ACM Computing Surveys, Vol 43, No 1, Article 3, November 2010.
[12]. Yiwu Xie, Yutong Li, Chunli Wang, Mingyu Lu. "The Optimization and Improvement of the Apriori Algorithm". In Proc. Int'l Workshop on Education Technology and Training & International Workshop on Geoscience and Remote Sensing 2008.
[13]. "Data mining Concepts and Techniques" by By Jiawei Han, Micheline Kamber, Morgan Kaufmann Publishers, 2006.
[14]. Pei.J, Han.J, Lu.H, Nishio.S. Tang. S. and Yang. D. "H-mine: Hyper-structure mining of frequent patterns in large databases". In Proc. Int'l Conf. Data Mining (ICDM), November 2001.
[15]. C.Borgelt. "Efficient Implementations of Apriori and Eclat". In Proc. 1st IEEE ICDM Workshop on Frequent Item Set Mining Implementations, CEUR Workshop Proceedings 90, Aachen, Germany 2003.
[16]. Agrawal.R and Srikant.R. "Fast algorithms for mining association rules". In Proc. Int'l Conf. Very Large Data Bases (VLDB), Sept. 1994, pages 487–499.
[17]. Park. J. S, M.S. Chen, P.S. Yu. "An effective hash-based algorithm for mining association rules". In Proc. ACM-SIGMOD Int'l Conf. Management of Data (SIGMOD), San Jose, CA, May 1995, pages 175–186.
[18]. ZHOU Jun, CHEN Ming, XIONG Huan A More Accurate Space Saving Algorithm for Finding the Frequent Items.IEEE-2010.
[19]. Yong-gong Ren,Zhi-dong Hu,Jian Wang. An Algorithm for Predicting Frequent Patterns over Data Streams Based on Associated Matrix. Ninth Web Information Systems and Applications Conference, 2012. 95-98.
[20]. Mahmood Deypir, Mohammad Hadi Sadreddini, A New Adaptive Algorithm for Frequent Pattern Mining over Data Streams, ICCKE,2011, 230-235 FLEX Chip Signal Processor (MC68175/D), Motorola, 1996.
[21]. Abdullah Al-Mudimigh, Farrukh Saleem, Zahid Ullah Department of Information System: Efficient implementation of data mining: improve customer's behavior, 2009 IEEE, (2014), pp.7-10.
[22]. Euiho Suh, Seungjae Lim, Hyunseok Hwang, Suyeon Kim: A prediction model for the purchase probability of anonymous customers to support real time web marketing: a case study, Expert Systems with Applications 27, (2015), pp. 245-250.
[23]. Anthony Danna, Oscar H. Gandy, and Journal of business ethics: 2016, all that glitters is not gold: Digging Beneath the surface of data mining.
[24]. Sreekumar Pulakkazhy (2016).Journal of computer science 9(10): 1252-1259, 2013. Noorul Islam university, Tamil Nadu, India.
[25]. Ulaanbaatar (2015), Developing a cost-benefit analysis of mining sites in Mongolia, annual report -2015.
[26]. Nan-Chen, H. & C. (2014), In Enhancing consumer Behavior analysis by data minin. ROC. Taipei.
[27]. Balaji Padmanabhan and Alexander Tuzhilin, Institute for Operation Research and Management Science: 2016, On the use of optimization for data mining: Theoretical Interaction and e-CRM opportunities, 2016.