CHAPTER 2

LITERATURE SURVEY

2.1 Introduction

A survey of application of association rule mining in e-commerce, various algorithms developed for frequent itemsets discovery, and the architectures and models that integrates e-commerce and data mining are presented in this chapter.

Association Rule mining is the mostly preferred data mining technique when it comes to large amount of data especially in the field of online business. E-commerce utilizes association rules to find frequent patterns in purchase records of users and page visits to boost online business, improve web site design and appearance to attract more customers. Web personalization, recommender systems are also implemented with the aid of association rule mining.

The first phase of Association rule mining is finding frequent itemsets. This phase is the field of research and many algorithms are devised in history for finding frequent itemsets.

Many architectures and models are developed that integrates e-commerce and data mining. Some are stand-alone models and others are built in architectures.
A survey is made in the above discussed three areas to develop a new and efficient model for finding frequent patterns in e-commerce domain using a new Association rule mining algorithm.

2.2 Applications of Association Rule mining in E-commerce

Association rule mining is applied to e-commerce environment in various forms like web personalization, recommender system, collaborative filtering, cross selling analysis etc., A survey of the application of association rule mining to e-commerce is made here.

Web Personalization

Personalization is the use of customer information for delivering a customized solution to that customer thus satisfying personal needs [65][80]. In the e-commerce environment, the available choices for the visiting customers are more. The search cost and time increase due to this is overload. Personalization can help the customer in decision-making process. Personalization can also communicate appropriate messages to the right customers based on customer profiles. Murthi &sarkar [65] discuss typical stages in personalization. Common way to include personalization in a firm’s interaction with customers is with recommender system [77]. Mobasher [61] discusses how association rule mining is performed on web data to make a web personalized system.

Recommender systems

Customers prefer to have a recommender system by which customer can see the feedback from other users who already purchased
the products. E-commerce makes use of recommender system to not only show feedback from other users but also suggest interesting and useful products to customers. Geyer et al [33] describes a recommender system that uses association rules derived from past purchases, for making recommendations to new anonymous customers. Diverse recommendation systems are proposed for different business, which guides the customers to find products they would like to purchase [18][55][77]. Most of them are based on either content filtering or collaborative filtering. Content Based Filtering (CBF) approach recommends products to target customers according to the preferences of their neighbors. The Collaborative Filtering (CF) approach recommends products to object customers based on their past preferences. The drawbacks in these traditional approaches are rectified and the personalized system was proposed by Yiyangzhang e al [97]. Zhang Xizheng [102] proposes a personalized recommendation system using association rule mining and classification. Set of association rules are mined from customer requirements database using Apriori algorithm and then apply CBA-CB algorithm to produce best rules out of completely set of rules.

Cross-selling analysis

The association rule mining is a powerful tool to realize cross selling. Cross selling is a marketing strategy to sell a new product or service to the customer who already used the products of the same enterprise. To introduce a new product or service to a new customer and an old customer, the old customer is more likely to accept it and the success rate is higher [52]. Cross selling is a marketing method, which can improve customer value. The more the relations between the
enterprise and the customer, the more dependent the customer will be on the enterprise and the higher the loyalty will be. Xiao-li Yin [94] discusses how the banks analyze the association relations between the deal activities and other properties like customer age, gender, education, occupation, etc, and can get the result which bank services or financial products the customer will be interested in.

**Purchasing and traveling behavior of customers.**

In e-commerce environment, finding association rules between purchasing items is very important. There are many algorithms devised in this field [3] [13] [35][67]. Path traversal pattern mining is the technique that finds most of the navigation behaviors of customers in the e-commerce environment. This information can be used to improve the website design and performance. Navigational suggestions can also be given to customers using this information. Many works are carried out in this field [20][21] [96]. Yeu-shi-lee et al [101] propose an algorithm IPA that considers both purchasing behavior and travelling patterns of customer at the same time.

2.3 Algorithms for Mining Frequent Itemsets

**AIS algorithm**

Rakesh Agrawal et al [75] invented a series of algorithms and new systems called artificial immune systems were designed. AIS algorithm uses candidate generation to detect the frequent itemsets. The candidates are generated on the fly and are compared with previously found frequent itemsets. The disadvantage of the algorithm is that it generates and counts
too many candidate itemsets that turn out to be small. AIS was the first algorithm that introduced the problem of generating association rules.

**SETM algorithm**

SETM algorithm was actually created by Houtsma and Swami [60] in October 1993 in IBM’s Imaden Research Center. However, was officially released in 1995. The algorithm generates candidates on the fly based on the transaction read from the database. SETM separates candidate generation from counting. It generates the candidates using equijoins first and then it sorts them all and removes the ones that do not meet the minimum support. The disadvantage is SETM generates many candidate itemsets, which at the end turn out not be frequent.

**APRIORI algorithm**

Agrawal and Srikant[74] created APRIORI, which are the most important data mining algorithms for mining frequent itemsets and associations. It opened new doors in the mining field. Many Apriori-like algorithms were developed based on Apriori algorithm. Mainly these authors created the IBM’s Intelligent Miner. Apriori uses a breadth-first search strategy to count the support of itemsets and uses a candidate generation function, which uses the downward closure property of support.

**DHP - Direct Hashing and Pruning**

J.S. Park et al [68] developed DHP and it uses a hash technique. This Hashing technique helps to generate candidate itemsets efficiently, in particular for the large two itemsets. This greatly improves the performance of the whole process. In addition, DHP employs effective pruning techniques to progressively reduce the transaction database size. It is however, a variation of the Apriori algorithm.
**Partition algorithm**

Ashok et al [9] presented an algorithm Partition that is fundamentally different from all the previous algorithms. Partition algorithm reads the database at most two times to generate all significant association rules. In the first scan of the database, it generates a set of all potentially large itemsets. This is done by scanning the database once and dividing it into a number of non-overlapping partitions. This set is a superset of all frequent itemsets, so it may contain itemsets that are not frequent. During the second scan, counters for each of these itemsets are set up and their actual support is measured.

**ECLAT - Equivalence Class Clustering and Bottom-up Lattice Traversal**

ECLAT algorithm presented by Zaki et al, [63] is the first algorithm that uses a vertical data (inverted) layout. ECLAT is very efficient for large itemsets but less efficient for small ones. The frequent itemsets are determined using simple tid-list intersections in a depth-first graph.

**DIC- Dynamic Itemset Counting**

Brin et al [13] put forth the DIC algorithm that separates the database into intervals of a fixed size to reduce the number of traversals through the database.

**FP-Growth algorithm**

The two main drawbacks of Apriori are the possible need of generating a huge number of candidates if the number of frequent 1-itemsets is high and repeated scan of the database to match the candidates and determine the support. The Frequent-pattern growth (FP-growth) algorithm devised by Jiawei Han et al., [44] does overcome the above
said drawbacks. It adopts a divide-and-conquer strategy and a frequent-pattern tree.

**TREE-PROJECTION algorithm**

Ramesh C. Agarwal et al.’s [1] TREE-PROJECTION algorithm uses a lexicograph tree, which requires substantially less memory than a hash tree. The support of the frequent itemsets is counted by projecting the transactions onto the nodes of this tree. This improves the performance of counting the number of transactions that have frequent itemsets. The lexicograph tree is traversed in a top-down fashion.

**PASCAL**

PASCAL algorithm is named after the French mathematician Blaise Pascal who invented an early computing device and it is an optimization of the Apriori algorithm. Bastide, Y [10] introduced the notion of key patterns and show that other frequent patterns can be inferred from the key patterns without access to the database. The algorithm finds both frequent and closed sets and it is twice as fast as Close and 10 times as fast as Apriori but is only practical when the pattern length is short.

**RELIM (Recursive Elimination)**

Christian Borgelt [22] developed RELIM algorithm, which is strongly inspired by FP-growth and very similar to H-mine. It does not use prefix trees and any other complicated structures. The work is done in one simple recursive function, which can be written with relatively few lines of code.
A comparative study of the algorithms is shown in the table 2.1.

Table 2.1 Comparison of frequent itemsets algorithms

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Authors and year</th>
<th>Methodology</th>
<th>Technique used</th>
<th>Merits</th>
<th>Demerits</th>
</tr>
</thead>
<tbody>
<tr>
<td>SETM</td>
<td>Rakesh Agrawal et al., 1993</td>
<td>Candidate generation exploiting downward closure property</td>
<td>Breadth first search</td>
<td>Supports SQL based computing</td>
<td>Generates too many candidate itemsets</td>
</tr>
<tr>
<td>APRIORI</td>
<td>Maurice Houtsma, Arun Swami 1995</td>
<td>Generates candidates based on the transaction read from the database</td>
<td>Breadth first search</td>
<td>Pruning</td>
<td>Multiple scans of database.</td>
</tr>
<tr>
<td>DHP - Direct Hashing and Pruning</td>
<td>J. S. Park, M. Chen, P.S. Yu. 1995</td>
<td>Employs hashing techniques</td>
<td>Breadth first search</td>
<td>Reduce the transaction database size</td>
<td>Just a variation of APRIORI</td>
</tr>
<tr>
<td>Partition</td>
<td>Ashok Savasere et al., 1995</td>
<td>Non-overlapping Partitions of database</td>
<td>Depth first search</td>
<td>Only two scans of database is made</td>
<td>Sometimes partition is too many</td>
</tr>
<tr>
<td>ECLAT</td>
<td>Mohammed JaveedZaki et al., 1997</td>
<td>Uses vertical data lay out</td>
<td>Depth-first search</td>
<td>Very efficient for large itemsets</td>
<td>Less efficient for small ones</td>
</tr>
<tr>
<td>FP-GROWTH</td>
<td>Jiawei Han, Jian Pei, Yiwen Yin 2000</td>
<td>Adopts divide-and-conquer strategy</td>
<td>Depth-first search</td>
<td>Only two scans of database</td>
<td>Tree size can be large and expensive</td>
</tr>
</tbody>
</table>
2.4 Algorithms for Mining Maximal frequent Itemsets

An itemset is maximal frequent if it has no superset that is frequent. A discussion of some of the algorithms for maximal frequent is made in this section.

**MAX-MINER algorithm**

The algorithm MAX-MINER proposed by R.J. Bayardo[11] extracts only the maximal frequent itemsets by extracting the maximal frequent itemsets. Since any frequent itemset is a subset of a maximal frequent itemset, Max-Miner generates all the frequent itemsets. The algorithm combines a level wise bottom-up traversal with a top-down traversal in order to quickly find the maximal frequent patterns. Then, all frequent patterns are derived from these ones and the last database scan is carried on to count their support.

**Depth Project algorithm**

Agrawal’s Depth Project [75] algorithm finds frequent itemsets by using depth first search on a lexicographic tree of itemsets. The authors claimed and proved that this algorithm is faster than MaxMiner.

**MAFIA algorithm**

MAFIA proposed by Burdick [28] is an algorithm for mining maximal frequent itemsets from a transactional database. It is especially efficient when the itemsets in the database are very long. The search strategy integrates a depth-first traversal of the itemset lattice.

**GenMax algorithm**

Karam Gouda et al., [48] nominated an algorithm GenMax. It is a backtrack search based algorithm for mining maximal frequent itemsets.
It uses progressive focusing to perform maximality checking, and diffset propagation to perform fast frequency computation.

Hidber [36] has presented a novel algorithm named CARMA (Continuous Association Rule Mining Algorithm), which is used to compute large itemsets online. Wei Wang, et al. [91], identified a new class of interesting problem called weighted association rule (WAR). They have proposed an approach, which mines Weighted Association Rules by first ignoring the weight and finding the frequent itemsets, and it was followed by introducing the weight during the rule generation.

Hua-Fu Li et al. [39] proposed a new single-pass algorithm, called DSM-FI (Data Stream Mining for Frequent Itemsets), which mines all frequent itemsets over the entire history of data streams. Yu-Chiang Li et al. [100] have evaluated the significance of itemsets for the mining of association rules from databases. They have proposed an algorithm, Enhanced FSM (EFSM), which efficiently reduces the time complexity of the join step.

CTU-PRO algorithm was proposed by Alva Erwin et al. [5] to mine the complete set of high utility itemsets from both sparse and relatively dense datasets with short or long high utility patterns. Chun-Jung Chu et al. [23] have proposed a novel method, namely THUI (Temporal High Utility Itemsets)-Mine, used for mining temporal high utility itemsets from data streams efficiently and effectively.
Table 2.2 Comparison of maximal frequent itemsets algorithms

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Authors and year</th>
<th>Methodology</th>
<th>Technique used</th>
<th>Merits</th>
<th>Demerits</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAX-MINER</td>
<td>R.J. Bayardo 1998</td>
<td>Generates all the frequent itemsets</td>
<td>Combined bottom up and top down traversal</td>
<td>Limited scans of database</td>
<td>Efficient only for longest frequent itemsets</td>
</tr>
<tr>
<td>DepthProject</td>
<td>Ramesh Agrawal et al., 2000</td>
<td>Makes a lexicographic tree of itemsets</td>
<td>Depth first search</td>
<td>Faster than Maxminer</td>
<td>More database scans</td>
</tr>
<tr>
<td>MAFIA</td>
<td>Doug Burdick, Manuel Calimlim et al., 2001</td>
<td>Counting and projection mechanism</td>
<td>Depth-first search with pruning</td>
<td>Efficient when the itemsets in the database are very long</td>
<td>Less efficient for small databases</td>
</tr>
<tr>
<td>GenMax</td>
<td>Karam Gouda and Mohammed J. Zaki 2001</td>
<td>Perform maximality checking, and diffset propagation</td>
<td>Backtrack search</td>
<td>Enumerating the exact set of maximal patterns</td>
<td>Not suited for all type of datasets</td>
</tr>
</tbody>
</table>

2.5 Data mining and E-commerce – integrated Architectures

Kohavi et al [7] proposed an architecture that successfully integrates data mining with an e-commerce system. The architecture shown in figure 2.1 consists of three main components: Business Data Definition, Customer Interaction and Analysis, which are connected using data transfer bridges. Authors claim that this integration effectively solves several major problems associated with horizontal data mining tools.
including the enormous effort required in preprocessing of the data before it can be used for mining, and making the results of mining actionable. The tight integration between the three components of the architecture allows for automated construction of a data warehouse within the Analysis component.

The shared metadata across the three components further simplifies this construction, and, coupled with the rich set of mining algorithms and analysis tools also increases the efficiency of the knowledge discovery process. The tight integration and shared metadata also make it easy to deploy results, effectively closing the loop. Finally, authors presented several challenging problems that need to be addressed for further enhancement of this architecture.

Frank et al [32] developed a bare bones version of an Online Department Store that integrates interactive mining operations. In this e-commerce system, the data mining algorithms have been encapsulated into Microsoft SQL Server stored procedures. What the authors have achieved through this project is to utilize the techniques that go behind

![Figure 2.1 High-level system architecture [7]](image)

Frank et al [32] developed a bare bones version of an Online Department Store that integrates interactive mining operations. In this e-commerce system, the data mining algorithms have been encapsulated into Microsoft SQL Server stored procedures. What the authors have achieved through this project is to utilize the techniques that go behind
data mining for specific supplier hunting, sales statistics, promotional campaigns and e-mail marketing. The tight integration between the three components of the architecture allows for automated construction of a data warehouse within the Analysis component. The shared metadata across the three components further simplifies this construction, and, coupled with the rich set of mining algorithms and analysis tools also increases the efficiency of the knowledge discovery process. This work provides a good start in the direction of integrating data mining into e-commerce systems. A multi-tier application model is divided into several modular tiers, each of which may be located on a different physical computer.

The data tier maintains all of the information needed for an application. Most often, this information is stored in a database. The middle tier of a multi-tier application acts as a sort of “mediator” between the data in the data tier and users of the application. All user requests for data (e.g., a request to view the catalog of products) go through the middle tier before reaching the database. Likewise, responses to requests for data travel back through the middle tier before reaching the user. The middle tier implements business logic and presentation logic to control interactions between users and data. Business logic enforces business rule (e.g., a promotion policy) and is used to ensure data is reliable before it is updated in the database or retrieved for the user.

The third tier is the client tier, which provides a user interface for the application. Users and/or system administrators interact directly with the client tier through the user interface. For the online shopping system, the client is a HTML or WML-enabled web browsers. The user or
administrator also makes requests through the user interface in the client tier.

Xiaofen Zhang et al [94], in their paper presents distributed data mining from the perspective of varied communication and designs a data mining model based on SOAP in e-commerce. The proposed model has a good performance of transplantation. It can cross platforms and heterogeneous data structures, as well as cross firewalls and proxy servers to communicate. It is flexible, transparent and scalable. The model will provide an effective, credible and viable data mining solution for the users of e-commerce.

![Distributed Data mining based on SOAP](image)

**Figure 2.2 Distributed Data mining based on SOAP [94]**

As shown in Figure 2.2, the entire system consists of four portions, i.e. various heterogeneous data sources, Distributed Data Mining Component, Integrated Mining Module and System Interface. Where, Integrated Mining Module and System Interface are at the local computer, whereas various heterogeneous data sources and Distributed Data Mining Component can be located on the remote computer. Various portions work together like this: First, the user inputs mining requests through the
system interface. For example, some enterprise wants to find out the inherent relationship between various commodities through web site A and web site B. The request is passed to Integrated Mining Module. After analyzing, the module knows that it needs the intermediate mining results of site A and site B. Then the module calls Distributed Data Mining Component to mine the data located on site A and site B. Finally, Integrated Mining Module will mine the intermediate results synthetically and transfer the results to the user through System Interface.

Hong Yu et al [39] propose a general architecture for collecting and mining data out of an e-commerce platform. In this proposed architecture, there are four main modules, Data Collection, Data Pre-processing, Pattern Discovery, and Knowledge Analysis. The integrated architecture can effectively serve knowledge management in e-commerce.

2.6 Discussion

A survey of frequent itemset algorithms and a comparative study of them are made in this chapter. Various applications of association rule mining in the e-commerce environment are discussed. In addition, a survey is made on architectures that integrated data mining and e-commerce.