Chapter 1

Introduction to Data Mining

Recently, our capabilities of both generating and collecting data have been increasing rapidly. The widespread use of bar codes for most commercial products, the computerization of many business and government transactions, and the advances in data collection tools have provided us with huge amounts of data. Millions of databases have been used in business management, government administration, scientific and engineering data management, and many other applications. It is noted that the number of such databases keeps growing rapidly because of the availability of powerful and affordable Database Management Systems (DBMS). DBMS manages and maintains databases on physical storage devices and provides the ability to store, access and modify the huge amount of data. It also provides a suite of utilities to manage and monitor the performance on those actions against the data. Examples of a DBMS would be Oracle, DB2, SQL/Server, etc.

With this huge amount of data, the current databases users are demanding more sophisticated information from them. A marketing manager, for example, is no longer satisfied with a simple listing of marketing contacts, but wants detailed information about customers, past purchases as well as predictions of future purchases.

Simple Structured Query Language (SQL) queries are not adequate to support this increased demand for useful information. Data mining steps in to satisfy this need. It is often defined as finding hidden information in a database. Alternatively, it has been called exploratory data analysis.

Is it possible to mine databases using DBMS? Traditional database access is performed using a well-defined query stated in a language such as SQL. The output of the query consists of the data from the database that satisfies the query. That is, the output is usually a subset of the database, but it may also be an extracted view or may contain aggregations. Data mining access of a database differs from this traditional access in several ways: (1) Query: The query might not be well formed or precisely stated. The data miner might not even be exactly sure of what he wants to see. (2) Data: The data accessed is usually a different version from that of the original operational database. The data to be mined have been cleaned and modified to better support the mining process. (3) Output: The output of the data mining query probably is not a subset of the database. Instead it is the output of some analysis of the contents of the database.

Before we proceed on, we should contrast three terms that are often used in the literature. These are data, information and knowledge.

- ***Data:***

  Data are any facts, numbers, or text that can be processed by a computer. Today, organizations are accumulating vast amounts of data in different formats and different databases. This includes:

  1. Operational or transactional data such as sales, cost, inventory, payroll, and accounting.

2. Non-operational data such as industry sales, forecast data, and macro economic data.

3. Meta data - data about the data itself such as logical database design or data dictionary definitions

- ***Information:***

  The patterns, associations, or relationships among data can provide *information*. For example, analysis of retail point of sale transaction data can yield information on which products are selling and when.

- ***Knowledge:***

  Information can be converted into knowledge about historical patterns and future trends. For example, summary information on retail supermarket sales can be analyzed in light of promotional efforts to provide knowledge of consumer buying behavior. Thus, a manufacturer or retailer could determine which items are most susceptible to promotional efforts.

1.1 Data Mining and Knowledge Discovery

The terms Knowledge Discovery in Databases (KDD) and Data Mining (DM) are often used interchangeably. Indeed, there have been many other names given in the literature to this process of discovering hidden patterns in data such as knowledge extraction, information discovery, exploratory data analysis, information harvesting, and unsupervised pattern recognition. Recently, KDD has been used to refer to a process consisting of many steps and is defined as:

Definition 1.1.1 ***Knowledge Discovery from Databases(KDD)***

KDD is the process of discovering valid, novel, useful, and ultimately understandable

patterns in massive databases. ■

Novel means that the knowledge being thought is not obvious or intuitive. *Useful* means that the knowledge has some applicability to solving real world problems and aids in decision making process. *Understandable* means that the knowledge be conveyed in a manner that is suitable for human consumption.

Figure 1.1 outlines the whole KDD process. As the figure shows, the process is interactive and iterative, involving numerous steps with many decisions being made by human experts. The following steps are composing the KDD process [12]:

1. *Learning the application domain*: Deciding relevant prior knowledge and goals of discovery application.

2. *Creating a target data set*: Selecting a data set, or focusing on a subset of variables or data samples on which discovery is to be performed.

3. *Data cleaning and preprocessing*: Basic operations such as the removal of noise. Deciding on strategies for handling missing data fields.

4. *Data reduction and transformation*: Using dimensionality reduction or transformation methods to reduce the effective number of variables under consideration or to find invariant representations for the data.

5. *Choosing functions of data mining*: Deciding whether the goal of the KDD process is classification, regression, association, and so on.

6. *Choosing the data mining algorithm(s).* Selecting method(s) to be used for searching for patterns in the database.

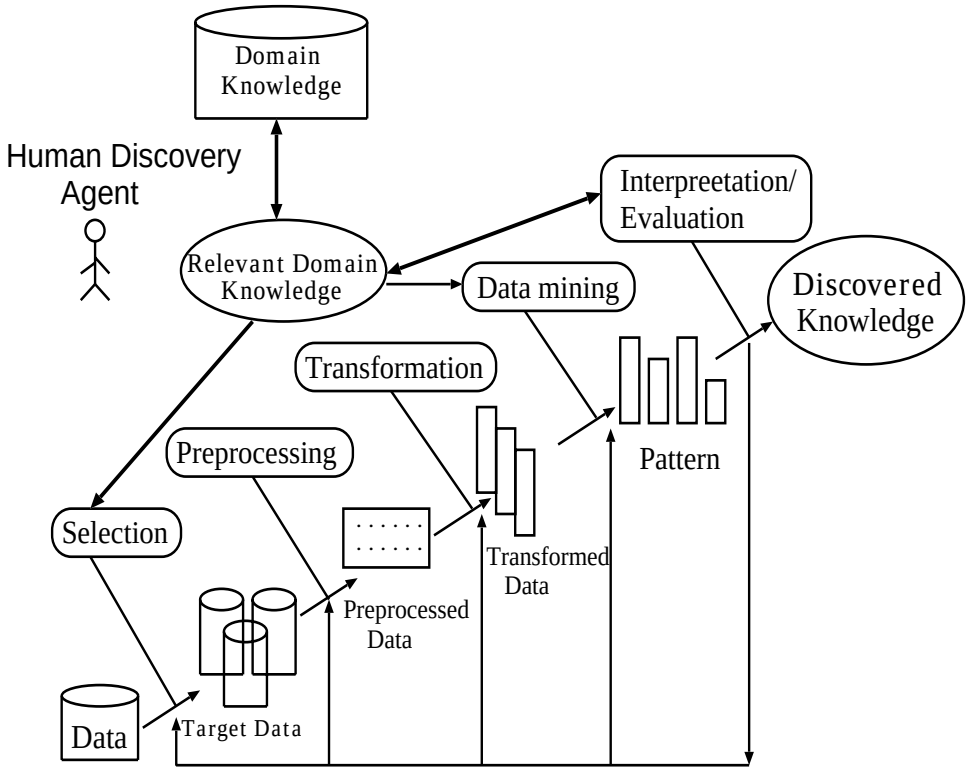

Figure 1.1: KDD Process Model [12, 15]

7. *Data mining*: Search for pattern of interest in a particular representational form or a set of such representations: Classification rules or trees, association, regression and so forth.

8. *Pattern evaluation and knowledge presentation*: Visualization, transformation, removing redundant patterns, etc., possible return to any of the steps 1-7.

9. *Use of discovered knowledge*: Incorporating the discovered knowledge into the performance system.

Figure 1.1 shows that data mining is a step in the whole KDD process. In fact it is the most intrinsic computational one. The figure also demonstrates that data mining is mainly concerned with the problem of patterns enumeration in the database. The current research on data mining can be viewed as addressing instances

of the problem:

Definition 1.1.2 ***Data Mining Problem [12, 15]***

Given a very large database, a representation language, thresholds and/or constraints, and relevant domain knowledge. Find all valid patterns. ■

what is pattern? A *pattern* is a sentence in the representation language describing subset of facts in the databases. *Thresholds* are statements determining the size of this subset. The representation language is used for representing the patterns and relevant domain knowledge. We say that a pattern is *valid* if it satisfies the thresholds and constraints. If we make a particular of the database as transactional databases, the representation language as attribute-value language, and threshold as frequency, we get the most popular Data Mining problem, **Frequent Pattern Mining(FPM)**.

1.1.1 Data mining - On What Kind of Data?

In principle, data mining should be applicable to any kind of information repository. These are the different types of data that are common in data mining applications.

- *Structured databases* such as relational, transactional, and multimedia databases.
- *Semi-structured databases* such as XML databases.
- *Unstructured databases* such as text databases.

In this thesis we are concerned with mining frequent patterns from transactional databases. In particular, we focus on two popular kinds of transactional databases which are *itemset* and *sequences* databases. They are defined as follows:

- **Itemset databases** A transactional itemset database consists of a table where each table's row represents a transaction. The transaction typically includes

a unique identifiers called transaction-id (tid for short), and a list of items called itemset making up the transaction (such as items purchased in a store). A simple example of transactional itemset database is shown in Table 1.1.

Tid	**List_of_items**
1	$acfh$

Table 1.1: Transactional Itemset Databases

- **Sequence databases** Transactional sequence databases consists of a list of sequences, where each sequence has an identifier, called sid, followed by a list of transactions in that sequence. A simple example of sequence databases is shown in Table 1.2.

Sid	**Sequence**
10	$< abc, bcd, abcd, e >$

Table 1.2: Transactional Sequence Databases

1.2 Data Mining Tasks

Generally speaking, there are two classes of data mining descriptive and prescriptive. Descriptive mining is to summarize or characterize general properties of data in data repository, while prescriptive mining is to perform inference on current data, to make predictions based on the historical data. There are various types of data mining tasks such as association rules, classifications, clustering, and sequential patterns. We will review those different types of mining techniques with examples.

- **Association Rule** Association Rule mining, one of the most important and well researched techniques of data mining, was first introduced in [1]. It aims to extract interesting correlations, frequent patterns, associations or casual

structures among sets of items in the transaction databases or other data repositories.

Example 1.2.1 *In an online book store there are always some tips after you purchase some books, for instance, once you bought the book Data Mining Concepts and Techniques, a list of related books such as: Database System 40%, Data Warehouse 25%, will be presented to you as recommendation for further purchasing.*

In the above example, the association rules are: when the book Data Mining Concepts and Techniques is brought, 40% of the time the book Database System is brought together, and 25% of the time the book Data Warehouse is brought together. Those rules discovered from the transaction database of the book store can be used to rearrange the way of how to place those related books, which can further make those rules more strong. Those rules can also be used to help the store to make his market strategies such as: by promotion of the book Data Mining Concepts and Techniques, it can blows up the sales of the other two books mentioned in the example. Association rules are widely used in various areas such as telecommunication networks, market and risk management, inventory control, etc. [8, 29, 23].

- **Classification** Classification [18] is to build (automatically) a model that can classify a class of objects so as to predict the classification or missing attribute value of future objects (whose class may not be known). It is a two-step process as follows. *In the first step*, based on the collection of training data set, a model is constructed to describe the characteristics of a set of data classes or concepts. Since data classes or concepts are predefined, this step is also known as supervised learning (i.e., which class the training sample belongs

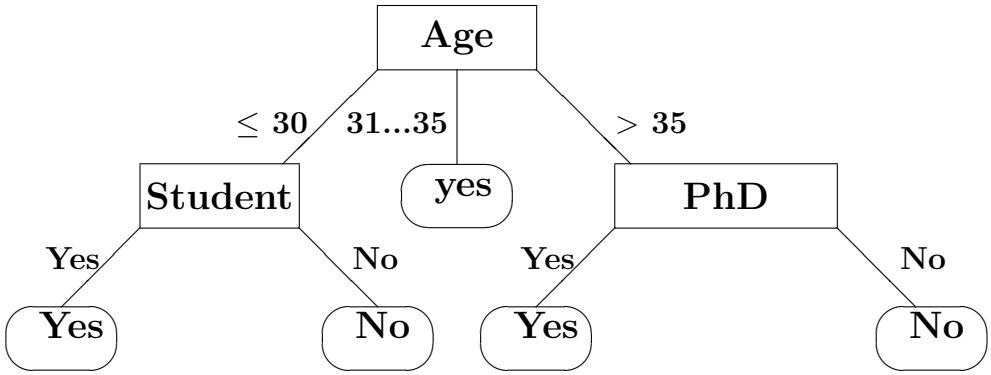

Figure 1.2: An Example of Decision Tree

to is provided). *In the second step*, the model is used to predict the classes of future objects or data. There are handful techniques for classification [18]. Classification by decision tree [34, 10, 35] was well researched and plenty of algorithms have been designed, Murthy did a comprehensive survey on decision tree induction [30]. Bayesian classification is another technique that can be found in Duda and Hart [11]. Nearest neighbor methods are also discussed in many statistical texts on classification, such as Duda and Hart [11] and James [22]. Many other machine learning and neural network techniques [24] are used to help constructing the classification models. A typical example of decision tree is shown in Figure 1.2. A decision tree for the class of buy laptop, indicate whether or not a customer is likely to purchase a laptop. Each internal node represents a decision based on the value of corresponding attribute, also each leaf node represents a class (the value of buy_laptop = Yes or No). After this model of buy_laptop has been built, we can predict the likelihood of buying laptop based on a new customer's attributes such as age, degree and profession. Those information can be used to target customers of

certain products or services, especially widely used in insurance and banking.

- **Clustering** As we mentioned before, classification can be taken as supervised learning process, clustering is another mining technique similar to classification. However clustering is a unsupervised learning process. Clustering is the process of grouping a set of physical or abstract objects into classes of similar objects [18, 9, 13], so that objects within the same cluster must be similar to some extend, also they should be dissimilar to those objects in other clusters. In classification which record belongs which class is predefined, while in clustering there is no predefined classes. In clustering, objects are grouped together based on their similarities. Similarity between objects are defined by similarity functions, usually similarities are quantitatively specified as distance or other measures by corresponding domain experts. Most clustering applications are used in market segmentation. By clustering their customers into different groups, business organizations can provided different personalized services to different group of markets. For example, based on the expense, deposit and draw patterns of the customers, a bank can clustering the market into different groups of people. For different groups of market, the bank can provide different kinds of loans for houses or cars with different budget plans. In this case the bank can provide a better service, and also make sure that all the loans can be reclaimed. A comprehensive survey of current clustering techniques and algorithms is available in [7].

- **Sequential Patterns** Sequential pattern mining, which discovers frequent subsequences as patterns in a sequence database, is an important data mining problem with broad applications, including the analysis of customer purchase patterns or Web access patterns, the analysis of the processes of scientific ex-

periments, natural disasters, disease treatments, DNA analysis, and so on. For example, consider the sales database of a bookstore, where the objects represent customers and the attributes represent authors or books. Let's say that the database records the books bought by each customer over a period of time. The discovered patterns are the sequences of books most frequently bought by the customers. An example could be that, "70% of the people who buy introduction to visual Basic and introduction to C++ also buy introduction to Perl within a month." Stores can use these patterns for promotions, shelf placement, etc. Consider another example of a web access database at a popular site (30% of users who visited /company/products/, had done a search in Yahoo, within the past week on keyword w; or 60% of users who placed an online order in /company/products/product1.html also placed an online order in /company1/products/product4.html within 15 days.), where an object is a web user and an attribute is a web page. The discovered patterns are the sequences of most frequently accessed pages at that site. This kind of information can be used to restructure the web-site, or to dynamically insert relevant links in web pages based on user access patterns.

The sequential pattern mining problem was first introduced by Agrawal and Srikant in [4]: Given a set of sequences, where each sequence consists of a list of elements and each element consists of a set of items, and given a user-specified, minimum support threshold, sequential pattern mining is to find all of the frequent subsequences, i.e., the subsequences whose occurrence frequency in the set of sequences is no less than minimum support. The task of discovering all frequent sequences in large databases is quite challenging. The search space is extremely large. For example, with m attributes there are $O(k^m)$ potentially frequent sequences of length k. With millions of objects in the database the

problem of I/O minimization becomes paramount.

1.3 Requirements and Challenges of Data Mining

In order to conduct effective data mining, one needs to first examine what kind of features an applied knowledge discovery system is expected to have and what kind of challenges one may face at the development of data mining techniques

- ***Handling of Different Types of Data:*** Because there are many types of data used in different applications, one may expect that a knowledge discovery system should be able to perform effective data mining on different types of data. Since most databases are available in relational model, it is crucial that a data mining system ables to perform efficient knowledge discovery on relational data. Moreover, many current applications involves complex data types such as structured and complex data objects, hypertext and multimedia data, spatial and temporal data, transaction data, etc. A powerful system should be able to perform effective data mining on such complex types of data as well. However, the diversity of data types and different goals of data mining make it unrealistic to expect one data mining system to handle all kinds of data. Specific data mining systems should be constructed for knowledge mining on specific kinds of data, such as systems dedicated to knowledge mining in relational databases, transaction databases, spatial databases, multimedia databases, etc.

- ***Efficiency and Scalability of Data Mining Algorithms:*** To effectively extract information from a huge amount of data in databases, the data mining algorithms must be efficient and scalable with large databases. That is, the running time of a data mining algorithm must be predictable and acceptable with large databases. Algorithms with exponential or even medium-order

polynomial complexity will not be of practical use.

- ***Incorporation of Background Knowledge:*** Background knowledge or information regarding the domain under study might be used to guide mining and focus the hypothesis space toward interesting patterns. Furthermore, it allows the patterns to be expressed in concise terms and at different levels of abstraction. Domain knowledge related to databases such as integrity constraints and deduction rules can help focus and speed up a data mining process. Developing new mining algorithms which make full use of relevant domain knowledge pose additional challenges to data mining research.

- ***High dimensionality:*** A conventional database schema may be composed of many different attributes. The problem here is that not all attributes are necessary to solve a given data mining problem. In fact, the use of some attributes may interfere with the correct completion of the mining task. The use of other attributes may simply increase the overall complexity and decrease the efficiency of an algorithm. This problem is sometimes referred to as the *dimensionality curse* meaning that there are many attributes (dimensions) involved and it is difficult to determine which ones should be used. One solution to this problem is to reduce the number of attributes which is known as dimensionality reduction. However, determining which attributes are not needed is not always easy to do.

1.4 Related Work and Thesis Contributions

The frequent patterns mining problem was originally introduced in the context of mining frequent itemsets from a database of sets of items in 1993 by R. Agrawal et al in [1]. Two years later, the problem was defined for mining frequent sequence from

sequence database by Agrawal and Srikant in [4]. The extensive research performed on this problem since its introduction has led to an overwhelming abundance of algorithms. Each algorithm typically consists of two interleaved steps, namely generation of candidate patterns and frequency testing. In most algorithms, generation is done by using one of the two popular tree traversals: depth-first and breadth-first. Since frequency testing plays the major role to the mining algorithm scalability, this thesis focuses on this important step.

Two popular data representations approaches are adopted: vertical and horizontal representation. In vertical-based algorithms, frequency counting is done via joining operations on tidsets or bitmaps [42, 43, 36, 6]. On the other hand, horizontal-based algorithms use database scans for this task. The first generation of horizontal algorithms–candidates generation is based on breadth first traversal–performs full scans of the entire database to evaluate the support of candidate patterns [1, 3, 2, 4, 5]. The second generation of horizontal algorithms–algorithms are based on depth first traversal–utilizes proper database projection to reduce the size of the database to be scanned, thus, accelerated the counting process [19, 20, 32].

Following the success that has been achieved by the vertical approach for FPM problem, we in this thesis introduce a novel vertical data representation called *primeset*. The elements in primeset are not transaction ids as in tidset. In fact, each element is a reference number that replaces a group of these ids whereas it preserves the information that the ids in the group represent. This groups-of-tids transformation into references is done based on the properties of prime numbers, thus, the so-called primeset. The primeset structure has been integrated with Eclat, a well-known vertical tidset-based mining algorithm for mining frequent itemsets. The new version

is called PrimeEclat. Primeset structure has also been extended to be employed in sequence mining. Its new form is referred to as *PS-list*. PS-list has been integrated with Spade, a well-known algorithm for mining sequential patterns, and the new version is called PrimeSpade. The experimental results show that the primeset representation delivers more than five times performance improvement over the normal vertical data representation in mining frequent patterns.

The thesis not only presents a novel data representation but also presents an adaptation of the well-known diffset data representation [46] to be employed in sequential pattern mining. The resulted structure is referred to as $diffseq$. To be tested, the diffseq data representation is integrated with Spade and the new version is called dSpade. Since diffset shows high performance for mining frequent itemsets in dense transactional databases, experimental evaluation shows that dSpade is suitable only for mining dense sequence databases. Finally, a performance comparison study is adopted to show that the role at which each structure, primset or diffset, is suitable for mining frequent patterns.

1.5 Thesis Outline

We begin by presenting FPM problem formulation and popular data representations in Chapter 2. The horizantal approach to the problem is described in Chapter 3. Chapter 4 introduces the vertical approach. Primeset, PS-list, and diffseq structures are introduced in Chapter 5. The chapter also presents PrimeEclat, PrimeSpade, and dSpade algorithms with optimizations. Experimental study of these new algorithms are covered in Chapter 6. Finally, conclusions and future work are presented in Chapter 7.