# CHAPTER 2

# LITERATURE SURVEY

## 

## 2.1 Data Mining

Data mining, or knowledge discovery, is the computer-assisted process of digging through and analyzing enormous sets of data and then extracting the meaning of the data. Data mining tools predict behaviors and future trends, allowing businesses to make proactive, knowledge-driven decisions. Data mining tools can answer business questions that traditionally were too time consuming to resolve.

Generally, data mining (sometimes called data or knowledge discovery) is the process of analyzing data from different perspectives and summarizing it into useful information - information that can be used to increase revenue, cuts costs, or both. Data mining software is one of a number of analytical tools for analyzing data. It allows users to analyze data from many different dimensions or angles, categorize it, and summarize the relationships identified. Technically, data mining is the process of finding correlations or patterns among dozens of fields in large relational databases. They scour databases for hidden patterns, finding predictive information that experts may miss because it lies outside their expectations.

Data mining derives its name from the similarities between searching for valuable information in a large database and mining a mountain for a vein of valuable ore. Both processes require either sifting through an immense amount of material, or intelligently probing it to find where the value resides.

**2.2 Data Mining Techniques**

There are several major data mining technique have been developing and using in data mining projects recently including association, classification, clustering, prediction, sequential pattern and decision tree.

**2.2.1 Data Preprocessing**

Raw data is highly susceptible to noise, missing values, and inconsistency. The quality of

Data affects the data mining results. In order to help improve the quality of the data and

Consequently, of the mining results raw data is preprocessed so as to improve the efficiency and ease of the mining process. Data preprocessing is one of the most critical steps in a data mining process which deals with the preparation and transformation of the initial dataset.

Data preprocessing methods are divided into following categories:

1. Data cleaning

2. Data integration

3. Data transformation

4. Data reduction

**2.2.1.1 Data Cleaning**

Data cleaning routines work to “clean” the data by filling in missing values, smoothing noisy data, identifying or removing outliers, and resolving inconsistencies. If users believe the data are dirty, they are unlikely to trust the results of any data mining that has been applied to it. Furthermore, dirty data can cause confusion for the mining procedure, resulting in unreliable output. Although most mining routines have some procedures for dealing with incomplete or noisy data, they are not always robust. Instead, they may concentrate on avoiding over fitting the data to the function being modeled.

**2.2.1.2 Data Integration**

It is likely that your data analysis task will involve data integration, which combines data from multiple sources into a coherent data store, as in data warehousing. These sources may include multiple databases, data cubes, or flat files. There are a number of issues to consider during data integration. Schema integration can be tricky. How can real world entities from multiple data sources be 'matched up'? This is referred to as the entity identification problem. For example, how can the data analyst or the computer be sure that customer id in one database, and customer\_number in another refer to the same entity? Databases and data warehouses typically have metadata that is, data about the data. Such metadata can be used to help avoid errors in schema integration. Redundancy is another important issue. An attribute may be redundant if it can be derived from another table, such as annual revenue. Inconsistencies in attribute or dimension naming can also cause redundancies in the resulting data set.

**2.2.1.3 Data Transformation**

In data transformation, the data are transformed or consolidated into forms appropriate for mining. Data transformation can involve the following:

1. Normalization, where the attribute data are scaled so as to fall within a small specified range, such as – 1.0 to 1.0, or 0 to 1.0.

2. Smoothing works to remove the noise from data. Such techniques include binning, clustering, and regression.

3. Aggregation, where summary or aggregation operations are applied to the data. For example, the daily sales data may be aggregated so as to compute monthly and annual total amounts. This step is typically used in constructing a data cube for analysis of the data at multiple granularities.

4. Generalization of the data, where low level or 'primitive' (raw) data are replaced by higher level concepts through the use of concept hierarchies. For example, categorical attributes, like street, can be generalized to higher level concepts, like city or county. Similarly, values for numeric attributes, like age, may be mapped to higher level concepts, like young, middle-aged, and senior.

**2.2.1.4 Data Reduction**

Complex data analysis and mining on huge amounts of data may take a very long time, making such analysis impractical or infeasible. Data reduction techniques have been helpful in analyzing reduced representation of the dataset without compromising the integrity of the original data and yet producing the quality knowledge. The concept of data reduction is commonly understood as either reducing the volume or reducing the dimensions (number of attributes). There are a number of methods that have facilitated in analyzing a reduced volume or dimension of data and yet yield useful knowledge. Certain partition based methods work on partition of data tuples. That is, mining on the reduced data set should be more efficient yet produce the same (or almost the same) analytical results.

Strategies for data reduction include the following:

1. Data cube aggregation, where aggregation operations are applied to the data in the construction of a data cube.

2. Dimension reduction, where irrelevant, weakly relevant, or redundant attributes or dimensions may be detected and removed.

3. Data compression, where encoding mechanisms are used to reduce the data set size. The methods used for data compression are wavelet transform and Principal Component Analysis.

4. Numerosity reduction, where the data are replaced or estimated by alternative, smaller data representations such as parametric models (which need store only the model parameters instead of the actual data e.g. regression and log-linear models), or nonparametric methods such as clustering, sampling, and the use of histograms.

5. Discretization and concept hierarchy generation, where raw data values for attributes are replaced by ranges or higher conceptual levels. Concept hierarchies allow the mining of data at multiple levels of abstraction, and are a powerful tool for data mining.

**2.2.2 Association**

Association is one of the best known data mining technique. In association, a pattern is discovered based on a relationship between items in the same transaction. That’s is the reason why association technique is also known as relation technique*.* The association technique is used in market basket analysisto identify a set of products that customers frequently purchase together.

Retailers are using association technique to research customer’s buying habits. Based on historical sale data, retailers might find out that customers always buy crisps when they buy beers, and therefore they can put beers and crisps next to each other to save time for customer and increase sales.

**2.2.3 Classification**

Classification is a classic data mining technique based on machine learning. Basically classification is used to classify each item in a set of data into one of predefined set of classes or groups. Classification method makes use of mathematical techniques such as decision trees, linear programming, neural network and statistics. In classification, we develop the software that can learn how to classify the data items into groups.

Classification is a task in Data mining. Data mining, as indicated before, is a machine learning discipline, and is inspired by pattern recognitions, which is a branch of science, of which one of its goals is to classify objects into a number of categories referred to as classes. Objects refer to compact data units specific to a particular problem, which is in general, known as patterns.

Classification prediction encompasses two levels: classifier construction and the usage of the classifier constructed. The former is concerned with the building of a classification model by describing a set of predetermined classes from a training set as a result of learning from that dataset. Each sample in the training set is assumed to belong to a predefined class, as determined by the class attribute label. The model is represented as classification rules, decision trees, or mathematical formula. The later involves the use of a classifier built to predict or classify unknown objects based on the patterns observed in the training set.

The entire process begins with collection of evidence acquired from various data sources or warehouses. In the ideal situation, the data should be of low-dimensionality, independent and

discriminative so that its values are very similar to characteristics in the same class but very different in features from different classes. Raw data hardly satisfies these conditions and therefore a set of procedures called feature generation, extraction and selection is required to provide a relevant input for classification system.

For example, we can apply classification in application that “given all records of employees who left the company, predict who will probably leave the company in a future period.” In this case, we divide the records of employees into two groups that named “leave” and “stay”. And then we can ask our data mining software to classify the employees into separate groups.

**2.2.4 Clustering**

Clustering is a data mining technique that makes meaningful or useful cluster of objects which have similar characteristics using automatic technique. The clustering technique defines the classes and puts objects in each class, while in the classification techniques, objects are assigned into predefined classes[3].

Clustering is a division of data into groups of similar objects. Representing the data by fewer clusters necessarily loses certain fine details, but achieves simplification. It models data by its clusters. Data modeling puts clustering in a historical perspective rooted in mathematics, statics, and numerical analysis. From a machine learning perspective clusters correspond to hidden patterns, the search for clusters is unsupervised learning, and the resulting system represents a data mining applications such as scientific data exploration, information retrieval and text mining, spatial database applications, web analysis, CRM, marketing, medical diagnostics, computational biology, and many others.

To make the concept clearer, we can take book management in library as an example. In a library, there is a wide range of books in various topics available. The challenge is how to keep those books in a way that readers can take several books in a particular topic without hassle. By using clustering technique, we can keep books that have some kinds of similarities in one cluster or one shelf and label it with a meaningful name. If readers want to grab books in that topic, they would only have to go to that shelf instead of looking for entire library.

**2.2.5 Prediction**

The prediction, as it name implied, is one of a data mining techniques that discovers relationship between independent variables and relationship between dependent and independent variables. For instance, the prediction analysis technique can be used in sale to predict profit for the future if we consider sale is an independent variable, profit could be a dependent variable. Then based on the historical sale and profit data, we can draw a fitted regression curve that is used for profit prediction.

**2.2.6 Sequential Pattern**

Sequential patterns analysis is one of data mining technique that seeks to discover or identify similar patterns, regular events or trends in transaction data over a business period.

In sales, with historical transaction data, businesses can identify a set of items that customers buy together a different times in a year. Then businesses can use this information to recommend customers buy it with better deals based on their purchasing frequency in the past.

**2.2.7 Decision Trees**

Decision tree is one of the most used data mining techniques because its model is easy to understand for users. In decision tree technique, the root of the decision tree is a simple question or condition that has multiple answers. Each answer then leads to a set of questions or conditions that help us determine the data so that we can make the final decision based on it.

**2.3 Clustering Techniques**

Clustering methods can be divided into two basic types: hierarchical and partitional clustering. Within each of the types there exists a wealth of subtypes and different algorithms for finding the clusters.

Hierarchical clustering proceeds successively by either merging smaller clusters into larger ones, or by splitting larger clusters. The clustering methods differ in the rule by which it is decided which two small clusters are merged or which large cluster is split. The end result of the algorithm is a tree of clusters called a dendrogram, which shows how the clusters are related. By cutting the dendrogram at a desired level a clustering of the data items into disjoint groups is obtained.

Partitional clustering, on the other hand, attempts to directly decompose the data set into a set of disjoint clusters. The criterion function that the clustering algorithm tries to minimize may emphasize the local structure of the data, as by assigning clusters to peaks in the probability density function, or the global structure. Typically the global criteria involve minimizing some measure of dissimilarity in the samples within each cluster, while maximizing the dissimilarity of different clusters.

**2.3.1 k-means Clustering**

The k-means algorithm, one of the mostly used clustering algorithms, is classified as a partitioned or nonhierarchical clustering method. Given a set of numeric objects X and an integer number k (≤n), the k-means algorithm searches for a partition of X into k clusters that minimizes the within groups sum of squared errors (WGSS). This process is often formulated as the following mathematical program problem P [4].

Minimize P(W, Q) = X k l=1 Xn i=1 wi,l d(Xi, Ql) (2.1)

Subject to X k l=1 wi,l = 1, 1 ≤ i ≤ n wi,l ∈ {0, 1}, 1 ≤ i ≤ n, 1 ≤ l ≤ k (2.2)

Where W is an n ×k partition matrix, Q = {Q1, Q2,..., Qk } is a set of objects in the same object domain, and d(·, ·) is the squared Euclidean distance between two objects.

Problem P can be solved by iteratively solving the following two problems:

1. Problem P1: Fix Q = Qˆ and solve the reduced problem P(W, Qˆ ).

2. Problem P2: Fix W = Wˆ and solve the reduced problem P(Wˆ , Q).

The k-means algorithm has the following important properties:

1. It is efficient in processing large data sets.

2. It often terminates at a local optimum (MacQueen, 1967; Selim and Ismail, 1984).

3. It works only on numeric values.

4. The clusters have convex shapes

**2.2.8.2 k-mode Clustering**

In principle the formulation of problem P in Section 3 is also valid for categorical and mixed type objects. The cause that the k-means algorithm cannot cluster categorical objects is its dissimilarity measure and the method used to solve problem P2. These barriers can be removed by making the following modifications to the k-means algorithm:

1. Using a simple matching dissimilarity measure for categorical objects,

2. Replacing means of clusters by modes, and

3. Using a frequency-based method to find the modes to solve problem P2

**2.2.8.2.1 Dissimilarity Measure**

Let X, Y be two categorical objects described by m categorical attributes. The dissimilarity measure between X and Y can be defined by the total mismatches of the corresponding attribute categories of the two objects. The smaller the number of mismatches is, the more similar the two objects. This measure is often referred to as simple matching (Kaufman and Rousseeuw, 1990). Formally,

d2(X,Y) = (2.3)

Where

δ(xj,yj) = (2.4)

**2.2.8.2.2 Mode of a set**

Let X be a set of categorical objects described by categorical attributes, A1, A2,..., Am. Definition 1. A mode of X = {X1, X2,..., Xn} is a vector Q = [q1, q2,..., qm] that minimizes

D(X, Q) = (2.5)

2.2.7. k-mode algorithm

When (5) is used as the dissimilarity measure for categorical objects, the cost function (1) becomes

P(W, Q) = (2.6)

where wi,l ∈ W and Ql = [ql,1, ql,2,..., ql,m] ∈ Q. To minimize the cost function the basic k-means algorithm can be modified by using the simple matching dissimilarity measure to solve P1, using modes for clusters instead of means and selecting modes according to Theorem 1 to solve P2. In the basic algorithm we need to calculate the total cost P against the whole data set each time when a new Q or W is obtained. To make the computation more efficient we use the following algorithm instead in practice.

1. Select k initial modes, one for each cluster.

2. Allocate an object to the cluster whose mode is the nearest to it. Update the mode of the cluster after each allocation.

3. After all objects have been allocated to clusters, retest the dissimilarity of objects against the current modes. If an object is found such that its nearest mode belongs to another cluster rather than its current one, reallocate the object to that cluster and update the modes of both clusters.

4. Repeat 3 until no object has changed clusters after a full cycle test of the whole data set.

Like the k-means algorithm the k-modes algorithm also produces locally optimal solutions that are dependent on the initial modes and the order of objects in the data set. In our current implementation of the k-modes algorithm we include two initial mode selection methods. The first method selects the first k distinct records from the data set as the initial k modes.

**2.2.8.3 k-prototype Clustering**

The algorithm is based on the k-means paradigm but removes the numeric data only limitation whilst preserving its efficiency. The algorithm clusters objects with numeric and categorical attributes in a way similar to k-means. Because objects are clustered against k prototypes instead of k means of clusters, called as k-prototype algorithm.

K-prototype is used to dynamically update the k prototypes in order to maximize the intra cluster similarity of objects. The object similarity measure is derived from both numeric and categorical attributes. When applied to numeric data the algorithm is identical to k-means.

**2.2.8.3.1 Mathematical Preliminaries**

Let X = {X 1 ,X 2 ,...,X n } denote a set of n objects and X i = [x i1 ,x i2 ,...,x im ] be an object represented by m attribute values. Let k be a positive integer. The objective of clustering X is to find a partition which divides objects in X into k disjoint clusters. For a given n, the number of possible partitions is definite but extremely large 1 . It is impractical to investigate every partition in order to find a better one for a classification problem. A common solution is to choose a clustering criterion to guide the search for a partition. A clustering criterion is called a cost function below.

**2.2.8.3.1.1 Cost Function**

The widely used cost function is the trace of the within cluster dispersion matrix .

One way to define this cost function is

E = (2.7)

Here, Q l = [ q i1 , qi2 , ... , qim ] is the representative vector or prototype for cluster l, and yil is an element of a partition matrix . d is a similarity measure often defined as the square Euclidean distance.

Y has the following two properties, (1) 0 ≤ yil ≤ 1 and (2) ∑ yil = 1 . Y is called a hard partition if yil ∊{0,1} . Otherwise, it is a fuzzy partition 2 . In a hard partition, ytl = 1 indicates that object Xi is assigned to cluster l by Y.

The inner term E l = ∑ yil d ( X i , Q l ) in Eq. (2.7) is the total cost of assigning X to cluster l, i.e., the total dispersion of objects in cluster l from its prototype Q l . E l is minimized if

qij = for j=1,….,m (2.8)

where n l = ∑ yil is the number of objects in cluster l.

When X has categorical attributes, we can introduce a similarity measure as

d(X1,Q1) = + (2.9)

where δ (p,q)=0 for p=q and δ (p,q)=1 for p≠q. and are values of numeric attributes, whereas and are values of categorical attributes for object i and the prototype of cluster l. m r and m c are the numbers of numeric and categorical attributes. γ 1 is a weight for categorical attributes for cluster l.

We can rewrite E l as

E1 = +γ1  (2.10)

where is the total cost on all numeric attributes of objects in cluster l. is minimized if is calculated by Eq. (2.2). Let Cj be the set containing all unique values in the categorical attribute j and p( cj ∊ Cj | l ) the probability of value c j occurring in cluster l. in Eq. (2.10) can be rewritten as

= (2.11)

where n1 is the number of objects in cluster 1.

**2.2.8.3.2 Similarity Measure**

The cost function Eq. (2.11) is defined on Eq. (2.9), which is a combined similarity measure on both numeric and categorical attributes between objects and cluster prototypes. The similarity measure on numeric attributes is the square Euclidean distance whereas the similarity measure on categorical attributes is the number of mismatches between objects and cluster prototypes. Weight γ l is introduced to avoid favoring either type of attribute.

The influence of weight γ l in clustering can be illustrated by Figure 1. Assume the triangles and diamonds represent a set of objects described by a categorical and two numeric attributes. Triangle and diamond represent two values of the categorical attribute whereas numeric attribute values are reflected by locations of the objects. These objects are partitioned into two clusters.

### If γ l =0, clustering only depends on numeric attributes, i.e., locations of the objects. The result will be the two clusters separated by the vertical dashed line. If γ l > 0, then object c may change to the right cluster because it is close to that cluster and its categorical value is the same as that of the majority of objects in that cluster. Similarly, object d may change to the left cluster. However, object a may still stay in the left cluster because it is too far to the right, even though it has a categorical value as same as that of the majority objects in that cluster. Similarly, object e may still be in the right cluster. Object b becomes uncertain, depending on whether γ l is biased towards numeric or categorical attributes. If γ l is biased to the categorical attribute, object b may change to the right cluster. Otherwise, it can stay in the left one. The choice of γ l is dependent on distributions of numeric attributes. Generally speaking, γ l is related to σ l, the average standard deviation of numeric attributes in cluster l. In practice, σ l can be used as a guidance to determine γ l. However, since σ l is unknown before clustering, the overall average standard deviation σ of numeric attributes can be used for all σ l. In an iterative algorithm, σ l can be calculated from a preceding clustering result. In Section 4.2 we will present some

### Simulations on γ l.