An Initial State Of Design And Development Of Intelligent Knowledge Discovery System For Stock Exchange Database

Zamzarina Che Mat @ Mohd Shukor, Rashid Hafeez Khokhar and Mohd Noor Md Sap

Information System Department
Faculty of Computer Science and Information System
University Technology of Malaysia, K.B. 791
81310 Skudai, Johor, Malaysia
Tel. 60-07-5532419, Fax. 60-07-5565044,
mc023013@siswa.utm.my rashid@siswa.utm.my , mohdnoor@fsksm.utm.my

ABSTRACT

Data mining is a challenging matter in research field for the last few years. Researchers are using different techniques in data mining. This paper discussed the initial state of Design and Development Intelligent Knowledge Discovery System for Stock Exchange (SE) Databases. We divide our problem in two modules. In first module we define Fuzzy Rule Base System to determined vague information in stock exchange databases. After normalizing massive amount of data we will apply our proposed approach, Mining Frequent Patterns with Neural Networks. Future prediction (e.g., political condition, corporation factors, macro economy factors, and psychological factors of investors) perform an important rule in Stock Exchange, so in our prediction model we will be able to predict results more precisely. In second module we will generate clustering algorithm. Generally our clustering algorithm consists of two steps including training and running steps. The training step is conducted for generating the neural network knowledge based on clustering. In running step, neural network knowledge based is used for supporting the Module in order to generate learned complete data, transformed data and interesting clusters that will help to generate interesting rules.

Keywords

Data Mining, Fuzzy Rules, Prediction Model, Frequent Patterns, Clustering, Neural Networks, Knowledge Base System.

1.0 INTRODUCTION

Data mining techniques have been studied and developed. There are three categories of data mining prototype, namely research prototype, commercial prototype and public domain and shareware prototype. We can use one of these prototypes to extract the knowledge from databases. For example, Jiawei, H., et al. (1996) developed DBMiner data mining system to extract the knowledge from the relational database and Zaiane et al. (1998) developed WebLogMiner to extract the knowledge from World Wide Web (WWW) database. Although many prototypes have been developed, it does not cater fully for the complex problems in the databases, especially in extracting knowledge from databases. In the area of economic-related information, economic forecasting is a complex and challenging task as noted by Chengyi, S., et al., (1996) due to some factors as follow: i) there is no economic model that carries conviction, ii) economic time series are intrinsically very unreliable and generally having poor signal to produce accurate result, and iii) non-stationary and non-linearity.

We divide our whole system in two modules; I and II. At starting, data selector will divide our database in two modules, in first module we have dealt such data which normally gives vague information, mean some records are missing or some crisp sets need to divide more degree of membership. In first module we don't want to miss any tiny result that can effect to our whole system. Our aim is to provide such intelligent system that gives more precise results. Here is a short introduction of module I.

We propose to use a new data mining technique to discover fuzzy rules for stock exchange databases. The proposed approach utilizes an objective measure to distinguish

interesting associations from uninteresting ones. Furthermore, it allows the ranking of discovered rules according to an inferred by the discovered fuzzy rules. The domain expert from the organization is interested at finding how we normalized our set of attributes. For the average monthly report of any organization, we applied the proposed approach to the SE databases in order to mine a set of fuzzy rules.

To allow human users to better understand the associations, we propose to use a new data mining technique to discover fuzzy rules employ linguistic terms, which are natural for human users to understand because of the affinity with the human knowledge representations, to represent the revealed association relationships. The use of linguistic terms, which are in turn defined by fuzzy sets, also allows the proposed approaches to be resilient to noises in the data. After fuzzification when we get normalized data we go for next step, in this step we apply our mining frequent pattern with neural networks approach. This is the extended form of frequent pattern tree (Agarwal, R. et al., 2000). For future results we will stimulate prediction model.

In second module we introduce two step clustering methods with neural network. The *training step* is conducted for generating the neural network knowledge based on clustering. In *running step*, neural network knowledge based is used for supporting the Module in order to generate learned complete data, transformed data and interesting clusters that will help to generate interesting rules.

Based on the above mentioned problems, at the present stage we investigate ways to make use of data mining techniques, neural networks and other intelligent techniques, i.e. fuzzy logics, knowledge base and case base reasoning in economic forecasting. The rest of the paper is organized as follows. The related works about classification will discussed in section 2, section 3 system capabilities, section 4 module I, and section 5 for module II. The discussion and conclusion are given in section 6 and 7 respectively.

2.0 RELATED WORK

In the previous research, a multitude of promising forecasting methods for predicting stock price from numeric data have been developed. These methods include statistics, ARIMA (Auto Regression Integrated Moving Average), Box-Jenkins, stochastic and neural networks (Clarence, N.W., 1993). For example, Duffie (Chengyi, S. et al., 1996) built stochastic models of stock market based on stochastic differential equation, but they have some disadvantages as follow: (i) there is a function used in the models which represent the influence on stock prices of various factors including corporation factors, macro economy factors, political factors and psychological factors of investors. The function is very difficult to decide or even cannot be decided at all; (ii) the models cannot be used for prediction.

One of the most popular tools to extract meaning and knowledge from database is data mining. Predictive data mining, one type of data mining method, is a major task in data mining and has wide applications, including credit evaluation, sales promotion, financial forecasting and market trend analysis (Shan, C., 1998). Many predictive data mining algorithms have been developed in the past research. For example, Shan, C., (1998) proposed predictive data mining algorithm which consists of three steps, namely data generalization, relevance analysis and statistical regression model. The problem of this algorithm is an inaccurate, irrelevant or missing data that is contained in the large amount of data in databases. This algorithm did not apply any special data preprocessing techniques to identify the inaccurate, irrelevant or missing data and thus the prediction model will not be reliable and accurate. In the middle of 1999, Wei, W., (1999) proposed two predictive data mining algorithms. The first algorithm is a classification-based method which integrates AOI (Attribute Oriented Induction) with the ID3 decision tree method. The second algorithm is a pattern matching based method which integrates statistical analysis with AOI to predict data values of the attribute of interest based on similar groups of data in databases. Both

predictive data mining methods provide high prediction quality and it leads to efficient and interactive prediction in large databases. However, both methods still carry several weaknesses and need to have further improvement.

Association rule mining is originally define in Agrawal, R. et al. (1993) over Boolean attribute in market basket data and has been extended to handle categorical and quantitative attributes (Srikant, R. and Agrawal, R., 1996). In this most general form, an association rule is defined over attributes of a database universal relation, T. An association rule is interesting if its support and confidence are greater than or equal to the user specified minimum support and minimum confidence respectively. A weakness of such approach is that many users do not have any idea what the thresholds should be. If it is set too low, the user miss some useful rules but if it is set too high, the user may be overwhelmed by many irrelevant ones (Han, J. and Kamber, M., 2001). Due to the following reasons first we refine SE databases. After completion our module 1 and module 2 we will use association rules for final results.

Regardless of how the values of quantitative attributes are discredized, the intervals might not be concise and meaningful enough for human users to easily obtain nontrivial knowledge from association rules discovered. Linguistic summaries introduced in Yager, R.R. (1991) express knowledge in linguistic representation, which is natural for human users to comprehend. In addition to linguistic summaries, an interactive top-down summary discovery process, which utilizes fuzzy is-a hierarchies as domain knowledge, has been described in Chan, K.C.C. and Au, W.H. (2000). This technique aims at discovering a set of generalized tuples. Unlike association rules, which involve implications between different attributes, linguistic summaries and generalized tuples provide summarization on different attributes only. The idea of implication has not been taken into consideration and hence these techniques are not developed for the task of rule discovery.

A well known technique for large itemsets generation is the *Apriori Approach* (Agrawal, R. and Srikant, R., 1994). In the *Apriori Approach* a level wise algorithm is used in order to generate itemsets. An *Apriori* algorithm may still suffer from the following two costs. (i) It is costly to handle a huge number of candidates' sets. (ii) It is tedious to repeatedly scan the database and check a large set of candidates by pattern matching, which is especially true for mining long patterns.

The *Tree-Projection* algorithm proposed by Agarwal et al. (2000) recently is an interesting algorithm, which constructs a lexicographical tree and projects a large database into a set of reduced, item-based sub-databases based on the frequent patterns mined so far. Tree-Projection may still encounter difficulties at computing matrices when the database is huge. The study in *Tree-Projection* (Agarwal, R. et al., 2000) has developed some smart memory caching methods to overcome this problem. However, it could be wise not to generate such huge matrices at all instead of finding some smart caching techniques to reduce the cost. Moreover, even if the matrix can be cached efficiently, its computation still involves some nontrivial overhead. And also when there are many long transactions containing numerous frequent items, transaction projection become a nontrivial cost of *Tree-Projection*.

Recently another category of methods, pattern-growth methods, such as *FP-tree* (Han, J. et al., 2000) have been proposed. A pattern growth method uses the Apriori property. However instead of generating candidate sets, it recursively partitions the database into sub-databases according to the frequent patterns found and search for longer frequent patterns to assemble longer global ones. FP-tree has three drawbacks (i) Huge space is required to serve the mining, (ii) Some databases (e.g. Telecommunications) needs real databases contain all the cases and (iii) Large applications need more scalability. In the above three approaches, *Apriori Approach* (Agrawal, R. and Srikant, R.,1994), *Tree-Projection* (Agarwal, R. et al., 2000), and *FP-tree* (Han, J. et al., 2000),

the most suitable approach is *FP-growth* (Han, J. et al., 2000) so we intend to use *FP-tree* for our stock exchange databases problem. In our propose approach we use neural network instead of *FP-growth.* By using neural network we can overcome some drawbacks of *FP-tree.*

Clustering is an exploratory tool for analyzing large datasets, and has been used extensively in numerous application areas. Clustering in data mining is a discovery process that groups a set of data such that the intra-cluster similarity is maximized and inter-cluster similarity is minimizes. The key element of clustering is the notion that the discovered groups are (Zhao, Y. and Karypis, G., 2002). Clustering is put in historical perspective by data modeling that rooted in mathematics, statistics, and numerical analysis. From machine learning perspective, clusters correspond to hidden patterns, the search of clusters is unsupervised learning, and the resulting system represents a data concept. Therefore, clustering is unsupervised learning of a hidden data concept (Berkhin, P. and Becher, J., 2002). A review of clustering applications is discussed in the following paragraphs.

Guha et al. (1998) introduced the hierarchical agglomerative clustering algorithm CURE. CURE is capable of finding clusters of arbitrary shapes and sizes, as it represents each cluster via multiple representative points and it is insensitive to outliers. Shrinking the representative points towards the centroid helps CURE in avoiding the problem of noise and outliers present in the single link method. While the algorithm CURE works with attributes (particularly low dimensional spatial data), the algorithm ROCK developed by the same researchers (Guha, S. et al., 1999) targets hierarchical agglomerative clustering for categorical attributes that employs links and not distance for merging cluster. Compared to traditional clustering algorithm that used distance between points of clustering, ROCK appropriate concept of links to measure the similarity/proximity between a pair of data points with categorical attributes. This method is naturally extended to non-matrix similarity measures. ROCK generates better quality clusters than traditional algorithms and exhibits good properties. However, CURE and ROCK fail to take into account special characteristics of individual clusters. These schemes also can make incorrect merging decisions when the underlying data does not follow the assumed model, or when noise is present.

Knowledge based system has been one of the most successful application of Artificial Intelligence (Hudli, A.V. and Palakal, M., 1991). Knowledge based system are represented by researchers in various way. Pi-Shen, D. (1994) proposed a computational case based reasoning model and investigated its feasibility to decision support. The goal of this model is to draw inferences for problem solving by recognizing similar circumstances in the past. In order to solve the problem, this model requires a set of past cases as the input, and this data set is represented as a relational data file. Generally, this model offers great advantages. Knowledge based system are represented by researchers in various way. A review of knowledge based applications is discussed in the following paragraphs.

Pi-Shen, D. (1994) proposed a computational case based reasoning model and investigated its feasibility to decision support. The goal of this model is to draw inferences for problem solving by recognizing similar circumstances in the past. In order to solve the problem, this model requires a set of past cases as the input, and this data set is represented as a relational data file. Generally, this model offers some of the following advantages (i) it has the ability to handle both quantitative and nominal data, however, a rule based inductive inference model is mainly used to deal with nominal data, and (ii) it is more suitable for unstructured oriented decisions, such as loan risk evaluation. However it still have the following limitations (i) it is a feasible decision support tool for every type of data applications for which data collection is very difficult, (ii) it requires the availability of a set of historical data along with result of evaluation for the induction knowledge, and (iii) it is only suitable for supervised learning. Hayashi, Y. and Imura, A.(1990) proposed a fuzzy neural expert system (FNES)

with automated extraction of fuzzy If..Then rules. Generally, fuzzy neural expert systems offer some of the following advantages (i) it provides a substantially higher diagnostic accuracy than that of linear discriminate analysis, and (ii) it can extract automatically practical knowledge in the form of fuzzy "If..Then" rules from neural network knowledge bases which are generated by learning process and training data accumulated in large scale databases.

3.0 SYSTEM CAPABILITIES

Looking at the above-mentioned problems, we are interested to improve some of the weaknesses, particularly on how to get the valuable knowledge and predicted values for large databases by using special data preprocessing techniques. The main objective of this research is to investigate the existing data mining method and propose an intelligent method for mining rules from large inconsistent stock exchange databases.

Figure 1 shows the general architecture for our proposed system. This system must capable to handle complex data mining problem, i.e. data cleaning, data transformation and rule generation. To do this, we will further study the available algorithms.

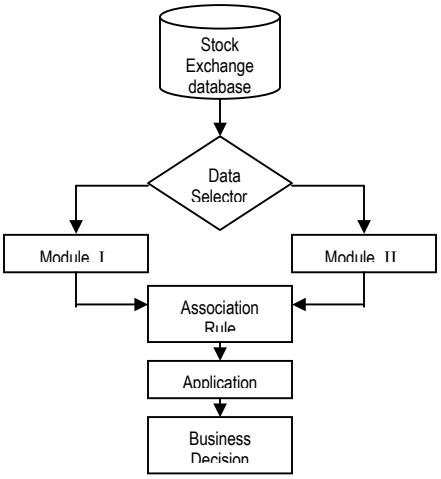

Figure 1: The Intelligent Knowledge Discovery System for Stock Exchange Database, General Architecture.

At the present stage, we are trying to combine two statistical techniques, namely classification and clustering. In this algorithm we also try to:

i. Apply special data preprocessing techniques, such as visualization tools and exploratory data analysis method. It will be used to identify the inaccurate, irrelevant or missing value of data in database and to make the prediction model reliable and accurate.
ii. Test two or (more) less relevant attributes.
iii. Remove inaccurate, irrelevant or missing value of data in database and group the similar or dissimilar attributes into a cluster. Analyze which of the attributes in databases that is highly relevant to do prediction.
iv. Generate interesting rules intelligently, reduce the number of generated rules and improve the accuracy of generating interesting rules.
v. Improve the performance of generated interesting rules and increase the speed of process by combining data mining method with other artificial intelligent application, i.e. knowledge based, case base reasoning fuzzy logic and neural network.
vi. Construct a new predictive modeling that can be used to extract the valuable knowledge and predict the next values.

4.0 Module I

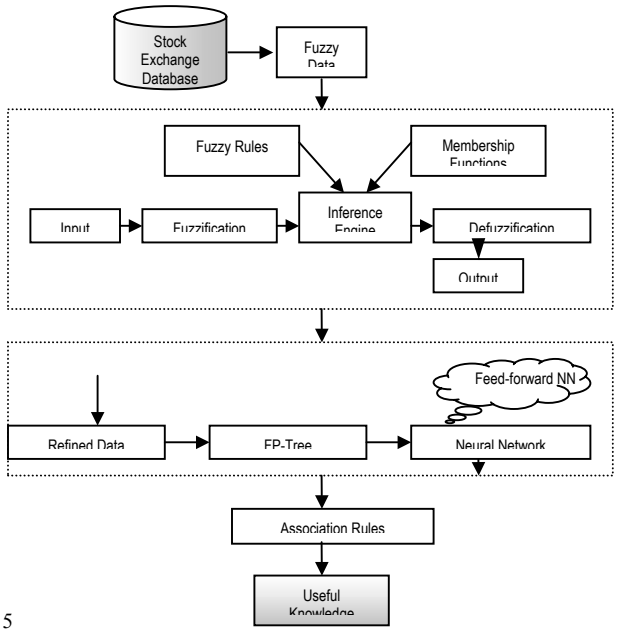

Figure 2: Structural Design of ModuleI

Figure 2 represented the general architecture of Module I, which is consists fuzzification as a main process.

4.1 Fuzzy Rule Base System

A linguistic approach for data mining is capable of mining fuzzy rules in large databases without any need for user-specified thresholds or mapping of quantitative into binary attributes (Keith C.C. Chan et al., 2001). A fuzzy rule describes an intersecting relationship between two or more linguistic variables. The definition of linguistic approach and short introduction of our fuzzy rule base system is presented in the following section.

4.1.1 Linguistic Approach

The fuzzy sets L_{ij} i = 1,……,sj are defined as (Keith C.C. Chan et al., 2001).

$$L_{ij} = \begin{cases} \sum_{dom(Iv)} \frac{m_{Lvr}(i_v)}{i_v} & \text{if Iv is discrete} \\ \int_{dom(Iv)} \frac{m_{Lvr}(i_v)}{i_v} & \text{if Iv is continuous} \end{cases} \quad (1)$$

for all iv dom(Iv). The degree of membership of some value iv dom(Iv) with some linguistic term is given by m_{Lij} .

Before applying this approach we should arrange our data, and categorize our universe of discourse (X). By this way we can categorized our universe of discourse. Also our proposed linguistic approach represents some preprocessors for defining linguistic to fuzzy sets as shown in figure 1.

Suppose D be a set of records, each of which consist of a set of attributes $\tilde{A}$ = {A_1, A_2,……..,A_n}, where A_i, i = 1,……,n, can be quantitative or categorical. For any records d D, d[A_i] denotes the value v_i in d for attribute Ai. For any quantitative attribute $A_i \in \tilde{A}$, let Lim(A_i) = [L_i, U_i] , denote the limit of the set attribute, where L_i is a lower limit and U_i is upper limit of quantitative attribute with in every class. max(Ai) d[Ai], min(Ai) d[Ai] be the maximum and minimum functions of set of attributes.

$$\Psi(d[Ai]) = \frac{max(d[Ai]) - min(d[Ai])}{\theta \in Z} \quad (2)$$

be the class interval that we use in making of every class, where q =10 be the threshold value for making of simple classes and it will be the same for every case because we always assign [0.1 to 1.0] fuzzy sets.

$$\breve{g}(Ai) = \min(Ai) + \Upsilon(Ai) \quad (3)$$

be the groups for set of attributes, in our problem we make 10 groups for universe of discourse (X).

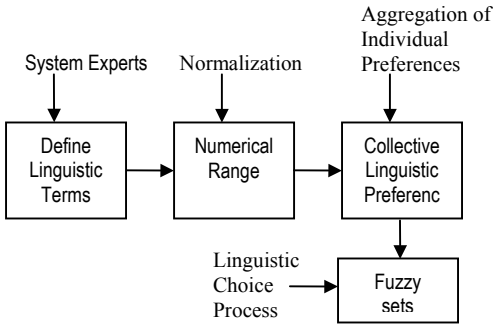

Figure 3: Formation of Linguistic Variables

Based on the fuzzy set theory, a set of linguistic labels can be defined over the domain of each quantitative attribute. Let us therefore denote the linguistic labels associated with some quantitative attribute, Ai $\tilde{A}$ as l_{ij}, j = 1,…..s_i, so that a corresponding fuzzy set, L_{ij} can be defined for each l_{ij}. The membership function of the fuzzy set is denoted as m_{Lij} and is defined as:

$$m_{Lij} : dom(Ai) \to [0, 1]$$
$$\text{Also } \mathbf{0} < m_{Lij}(d[Ai]) < \mathbf{1}, d \quad (4)$$

is partially characterized by the term 1_{ij}.

We intend to normalize every set of attributes before applying our model *Classification with Neural Network* for SE databases; consequently we present our structural design of fuzzy rule base system.

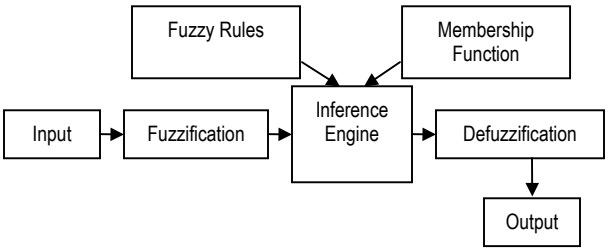

Figure 4: Structural Design of Fuzzy rules base system

4.1.2 Consistency Test

Let N be the total number of items, N Є Z, Z = 1,……..,n. (A), (B), (C) Є N be the occurrence events turn out to be positive and (α), (β), (γ) be the occurrence events turn out to be negative. (AB), (AC), (BC) are the possible pairs of (A), (B), and (C). (ABC) be the whole set of events.

The test of consistence is that no ultimate class-event should be negative.

$$\begin{aligned} Ŧ = (\alpha\beta\gamma) &= \alpha\beta\gamma.N \\ &= (1 - A)\,(1 - B)\,(1 - C).N \\ &= N - (A) - (B) - (C) + (AB) + (AC) \\ &\quad + (BC) - (ABC) \end{aligned} \tag{5}$$

If the result is negative then information as it stands, is not correct. However, if the data returned are alleged to be the result of an actual enquiry in a definite population, there must have been some misprint or miscount or miss-reporting.

Here we take only three events but we can use this consistency test for more than three events just adding positive and negative events respectively.

4.2 Mining Frequent Patterns with Neural Networks

4.2.1 Prediction models

Strictly a model is a simplification of the real world (Michel de Ruiter, 1999). Because reality is too complex to describe and reason with in a feasible manner, at least some simplifications and approximations are necessary. To reduce complexity, we can make assumptions about the kind of data variables can contain, the relation between variables, the amount of similarity between different samples, etc. The resulting world is the imposed model.

Often the term "prediction model" also comprises the procedure to reason with the imposed model. A prediction model often used in machine leaning is that of decision trees. All possible situations are split into a number of classes. These classes are split again into (sub) classes, etc, building a tree structure. For the construction of and searching in the tree several algorithms have been introduced. The tree can be read as defining several rules for the data.

Another model frequently used in practice is the neural network model. The model is named after similar structures in the brain. A model is imposed, that can be described with a network structure. The network contains a large number of weights, the parameters of the model. Those weights are adapted to the network has been trained; it can be applied to new data. Of the new data is similar to the training data and the network generalizes enough, the network will reliable prediction the output.

4.2.2 Frequent Pattern Tree (FP-Tree)

Like most traditional studies in association mining, the frequent pattern mining problem as follows (Han, J. et al., 2000).

Definition 1:- Let $I = \{a_1, a_2, \ldots, a_m\}$ be a set of items, and a transaction database $DB = \langle T_1, T_2, \ldots, T_n \rangle$, where T_i (i I [1…n]) is a transaction which contains a set of items in I. The support I (or occurrence frequency) of a pattern A, which is a set of items, is the number of transactions containing A in DB. A, is a frequent pattern if A's support is no less than a predefined minimum support threshold, x (Han, J. et al., 2000).

Given a transaction database DB and a minimum support threshold, x, the problem of finding the complete set of frequent patterns is called the frequent pattern mining problem.

Definition 2:- A frequent pattern tree (or FP-tree in short) is a tree structure defined below (Han, J. et al., 2000).

1. It consists of one root labeled as "null", a set of item prefix subtrees as the children of the root, and a frequent-item header table.
2. Each node in the item prefix subtree consists of three fields: item-name, count, and node-link, where item-name registers which item this node represents, count registers the number of transactions represented by the portion of the path reaching this node, and node-link links to the next node in the FP-tree carrying the same item-name, or null if there is none.
3. Each entry in the frequent-item header table consists of two fields, (i) item-name and (ii) head of node-link, which points to the first node in the FP-tree carrying the item-name.

A novel frequent pattern tree (FP-tree) (Han, J. et al., 2000) structure, which is an extended prefix-tree structure for storing compressed, crucial information about frequent patterns, and develop an efficient FP-tree-based mining method, FP-growth, for mining the complete set of frequent patterns by pattern fragment growth. Efficiency of mining is achieved with three techniques: (i) a large database is compressed into a highly condensed, much smaller data structure, which avoids costly, repeated database scans, (ii) our FP-tree-based mining adopts a pattern fragment growth method to avoid the costly generation of a large number of candidate sets, and (iii) a partitioning-based divide-and-conquer method is used to dramatically reduce the search space. Our performance study shows that the FP-growth method is efficient and scalable for mining both long and short frequent patterns, and is about an order of magnitude faster than the Apriori algorithm and also faster than some recently reported new frequent pattern mining methods.

4.2.3 Neural Networks

A neural network can be defined as a model of reasoning base on the human brain. The brain consists of a densely interconnected set of nerve cells, or basic information-processing units, called neurons. The human brain incorporates nearly 10 billions neurons and 60 trillion connections, synapses, between them (Shepherd and Koch, 1990). By using multiple neurons simultaneously, brain can perform its functions much faster computer in existence today.

The model is named after similar structures in the brain. A model is imposed, that can be described with a network structure. The network contains a large number of weights, the parameters of the model. Those weights are adapted to the network has been trained; it can be applied to new data. Of the new data is similar to the training data and the network generalizes enough, the network will reliable prediction the output.

4.2.4 Feed-forward Neural Network

The most frequently used artificial neural network is the feed-forward network. It consists of a layer of input nodes, a number of layers of hidden nodes, and a layer of output nodes (Figure 5). Each input node sends the value of an input variable into the network. All input nodes are connected with all nodes in the first hidden layer; the connections have values called weights. Every layer is fully connected with the next layer, until the last layer which represents the output variables. The non-input nodes also have a parameter called their bias.

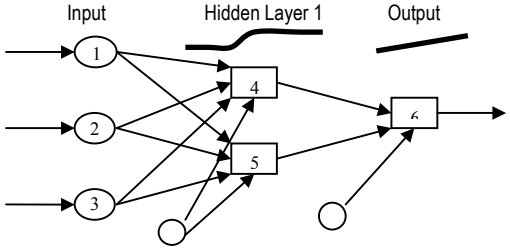

Figure 5: A feed-forward neural network with one hidden layer.

The nodes in the hidden layers and the output layer all have an activation function. The values sent to a node are multiplied with the associated weights, summed, and added to the node's bias. The activation function associated with the node is applied to this total value and the result is sent from the node to the next layer. Figure 5 shows a graphical representation of feed-forward neural network.

5.0 MODULE II

The general architecture of Module II is represented in figure 6. The main process of this Module is clusters generation. Generally it consists of two steps including training and running steps. The training step is conducted for generating the neural network knowledge based of clustering, i.e. the basic structure of neural networks knowledge based while the running step is used for creating the target output, i.e. the action part, of interesting clusters. In running steps, neural network knowledge based is used for supporting the module in order to generate learned complete data and interesting clusters that then will help to generate interesting rules. The generation of clustering in this module follows these steps (i) generate clusters of item sets in databases. The item sets are clustered into one cluster when the support and interestingness of pair item sets are greater than or equal to minimum support and minimum interestingness, (ii) evaluate the generated clusters in order to identify whether the generated clusters interesting or not, and (iii) generate rules from each cluster.

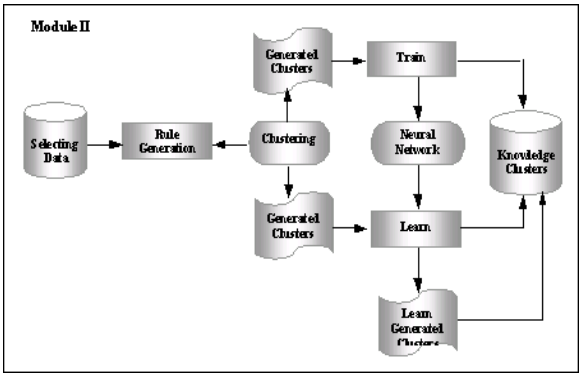

Figure 6 : The General Architecture of Module II

5.1 Agglomerative Clustering

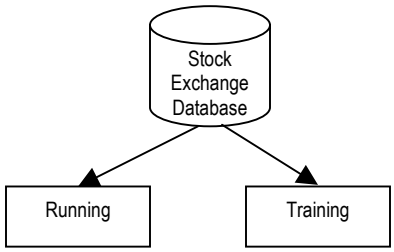

Figure 7: General Steps for Clustering Algorithms

An *agglomerative* clustering starts with singleton clusters and recursively merges two or more most appropriate clusters. A *divisive* clustering starts with one cluster of all data points and recursively splits the most appropriate cluster. The process continues until a stopping criterion is achieved. Advantages of hierarchical clustering include: (i) Embedded flexibility regarding the level of granularity (ii) Ease of handling of any forms of similarity or distance (iii) Consequently, applicability to any attribute types. Disadvantages of hierarchical clustering are related to: (i) Vagueness of termination criteria (ii) The fact that most hierarchical algorithms do not revisit once constructed (intermediate) clusters with the purpose of their improvement

Hierarchical clustering initializes a cluster system as a set of singleton clusters (agglomerative case) or a single cluster of all points (divisive case). It proceeds iteratively with merging or splitting of the most appropriate clusters until the stopping criterion is achieved. The appropriateness of clusters for merging/splitting depends on the similarity/dissimilarity of clusters elements. This reflects a general presumption that clusters consist of similar points. Example of dissimilarity between two points is the distance between them.

To merge or split subsets of points rather than individual points, the distance between individual points has to be generalized to the distance between subsets. Such derived proximity measure is called a *linkage metric*. The type of the linkage metric used significantly

affects hierarchical algorithms, since it reflects the particular concept of *closeness* and *connectivity*. Major inter-cluster linkage metrics (Olson, C., 1995) include *single link*, *average link*, and *complete link*. The underlying dissimilarity measure (usually, distance) is computed for every pair of points with one point in the first set and another point in the second set. A specific operation such as single link, average link, or complete link is applied to pair-wise dissimilarity measures:

$$d(C_1, C_2) = operation\ \{ d(x, y) \mid x \in C_1, y \in C_2 \}. \quad (6)$$

where C is the simplest attribute space subset which is a direct Cartesian product of subranges called segment. These subranges consist of a single category value, or of a numerical bin.

All of the above linkage metrics can be derived as instances of the Lance-Williams updating formula (Lance, G. and Williams W., 1999).

$$d(C_i \cup C_j, C_k) = a(i)d(C_i, C_k) + a(k)d(C_j, C_k) + bd(C_i, C_j) + c \mid d(C_i, C_k) - d(C_j, C_k) \mid \quad (7)$$

The similarity between two clusters is measured by the similarity of the closest pair of data points belonging to different clusters. Unlike the centroid/medoid based methods, this method can find clusters of arbitrary shape and different sizes. The single link method measures the similarity of two clusters by the maximum similarity between any pair of objects from each cluster, whereas the complete-link uses the minimum similarity. That is, the similarity between two clusters S_i and S_j is given by

$$sim_{single\text{-}link}(S_i, S_j) = \max_{d_i \in S_i,\, d_j \in S_j} \{\cos(d_i, d_j)\}. \quad (8)$$

The complete link scheme uses minimum similarity to measure the same similarity. That is,

$$sim_{single\text{-}link}(S_i, S_j) = \min_{d_i \in S_i,\, d_j \in S_j} \{\cos(d_i, d_j)\}. \quad (9)$$

The group average scheme (UPGMA) overcomes these problems by measuring the similarity of two clusters as the average of the pairwise similarity from each cluster. That is,

$$sim_{UPGMA}(S_i, S_j) = \frac{1}{n_i n_j} \sum_{d_i \in S_i, d_j \in S_j} \cos(d_i, d_j) = \frac{D_i^t D_j}{n_i n_j} \quad (10)$$

5.2 Neural Network Knowledge Based

In this module, two intelligent techniques, i.e. neural network and knowledge based are combined. It is used in both training and running steps. In the training steps it used for creating the neural network knowledge based of clustering. In the running step, it is used for supporting the module in order to generate learned complete cluster and interesting rules.

The basic steps of these approach are (i) compile/encode the available theoretical knowledge (domain theory) into an adequate Artificial Neural Network (ANN), (ii) use empirical data; sets of examples to train the network, hence introduce additional knowledge into ANN, (iii) extract the refined theory under symbolic form; to be reinserted into the ANN, the cycle being repeated, until some stopping criteria are satisfied.

The simplest way to implement discrete rules is in the form of binary logical functions. An n-variable binary logical function

$$f: \mathbf{B}^n \rightarrow \mathbf{B},\ n \in \mathbf{N},\ \mathbf{B} = \{F,T\};\ y = f(x_1, x_2, \ldots, x_n) \quad (11)$$

attaches one of the truth values F-false or T-true to its output (dependent, consequent) variable y, for each combination of the truth values of the input (independent, antecedent) variables $x_1, x_2, \ldots, x_n$.Usually, numeric coding is used for the binary logical values true (T) and false (F): 0 and 1 (unipolar representation), or –1 and 1 (bipolar representation), respectively.

6.0 DISCUSSION

The most suitable, effective and efficient method are identified and observed based on the result of mining the knowledge pattern. From this point, the enhancement will be held on the chosen method to deliver better performance result in term of the accuracy of detected knowledge pattern. After getting the most effective and efficient data mining algorithm and its, the implementation of knowledge discovery is held relying on the application of those methods. The development

and the improvement of the prototype system are conducted based on the evaluation of each part composing the system. The evaluation on the whole system will also be held to measure its performance in handling stock exchange data. After the system passing through many stages of development and evaluation, now it comes to the implementation of Knowledge Discovery System in the real world stock exchange environment as a pilot system. Continuous evaluation will be held to measure its performance by considering output accuracy in handling user request and effective time consumption for each operation.

7.0 CONCLUSION

Today, the mining of above rules is still one of the most popular pattern discovery methods in knowledge discovery in databases. It is clear that data mining has become more and more important due to the fast growth of data stored in databases. Consequently, new predictive data mining need to be developed to assist the humans in getting valuable knowledge and the predicted value from such a large amount of data. In our project which is related to SE databases, first data selector differentiates between normal and fuzzy data. Normal data has been sent to the module II where we use clustering techniques with neural networks and knowledge base systems. For the fuzzy data in module I we have used fuzzy rule base system.

In the first module we have been dealt special kind of data in stock exchange databases. Here we point out some general problems that seller/buyer has faced, records are missing, result gives unclear vision, lose prediction, crisp sets (0 or 1) affect the overall results. Consequently our proposed first model will be able to solve all of these mentioned problems easily. In our module we have been used fuzz linguistic approach. In this approach first defines some attributes as input or target attributes in a given relational database and then constructs a connectionist network to evaluate the reliability of values of target attributes in every record as a fuzzy measure. Unreliable values of target attributes can be removed from the database or corrected to the values predicted by the network. In addition to evaluating data reliability, the highest connection weights in the network can also be translated into a set of production rules. Although the information theoretic fuzzy approach is able to evaluate the reliability degree of data as a fuzzy measure, it requires the domains of quantitative attributes to be discretized into crisp intervals. After normalizing the data we apply our frequent pattern with neural networks. In this model we use neural network with frequent classification and in this way our proposed approach learn through experience by using feed-forward neural networks.

In the second module we apply combination of neural knowledge based with clustering agglomerative algorithm in the processing phase to generate interesting rules. Existing clustering algorithms for sequence and structure datasets operate on the object's similarity space. Algorithm such as feature- and similarity- based clustering algorithm are quite limiting as they cannot scale to very large datasets, cannot be used to provide a description as to why a set of objects was assigned to the same cluster that is native to the object's features and cannot be used to find clusters that have conserved features. Furthermore it also cannot provide a description as to why the objects assigned to the same cluster that is native to the object's feature. The only way to overcome this shortcoming is to : (i) Develop scalable and computationally algorithms for large sequence and structure datasets that operates directly on the object's native representation. (ii) Develop clustering algorithms that can provide concise explanation on the characteristics of the objects that were assigned to each cluster. The advantages of applying neural network knowledge based system with clustering algorithm in this module are: (i) it can eliminate superfluous attribute in the system formed by example cases and (ii) it can obtain reduced training examples for the neural network with the significant input.

After acquiring both modules we will apply association rules for better results. In our previous paper we already evaluate better approach in association rules. So in this paper

we collaborate with our previous work and finally conceived super system for stock exchange databases.

REFERENCES

Agarwal, R., Aggarwal, C. and Prasad, V. V. V. (2000). *A tree projection algorithm for generation of frequent itemsets. In Journal of Parallel and Distributed Computing* (Special Issue on High Performance Data Mining), (to appear).

Agrawal, R. and Srikant, R. (1994). *Fast algorithms for mining association rules*. In Proc. 1994 Int. Conf. Very Large Data Bases, Santiago, Chile : pp 487-499.

Agrawal, R., Imielinski, T. and Swami, A. (1993). *Mining Association Rules between sets of items in large in Databases.* In proceedings of the ACM SIGMOD Int'l Conf on management of data, Washington D.C. : pp, 207-216.

Alexandra, I.C. and Toshio, O. (1998). *Energy function construction and implementation for stock exchange prediction NNs*. Second International Conference Based Intelligent Electronic System.

Berkhin, P. and Becher, J. (2002). *Learning Simple Relations: Theory and Applications*. In Proceedings of the 2nd SIAM ICDM, 420-436, Arlington, VA.

Chan, K.C.C. and Au, W.H. (2000) *Mining Fuzzy Association Rules in a Database Containing Relational and Transactional Data*. In A. Kandel, M. Last, and H. Bunke.

Chengyi, S. et al. (1996). *Adaptive clustering of stock prices data using cascaded competitive learning neural networks*.

Clarence, N.W. (1993). *Trading a NYSE stock with a simple artificial neural network based financial trading system*. IEEE.

Eidhammer, I., Jonassen, I. and William R. Taylor. (2000). *Structure comparison and structure patterns.* Jorunal of computational Biology.

Guha, S. et al. (1998). *CURE: An efficient clustering algorithm for large databases.* In Proc. Of 1998 ACM-SIGMOD Int.Conf. on Management of Data.

Guha, S. et al. (1999). *ROCK: a robust clustering algorithm for categorical attributes.* In Proc. Of the 15th Int'l Conf. On Data Engineering.

Han, J. and Kamber, M. (2001). *Data Mining: Concepts and techniques*. Scan Francisco, CA: Morgan Kaufmann.

Han, J. et al. (1996). *DBMiner: A system for data mining in relational databases and data warehouse*. Proc 1996 International Conference on Data Mining and Knowledge Discovery, Portland, Oregon.

Han, J. et al. (2000). *Mining FrequentPatterns without Candidate Generation.* SIGMOD'2000 Paper ID: 196 : School of Computing Science Simon Fraser University

Hand, D., Mannila, H. and Smyth, P. (2001). *Principles of data Mining*. Cambridge MA: the MIT press.

Hudli, A.V. and Palakal, M. (1991). *A Neural Network Based Expert System Model.* Proceedings of the 1991 IEEE International Conference on Tool For AI, San Jose.

Kazuhiro, K. (1995). *Neural multivariate prediction using event knowledge and selective presentation learning*. IEEE.

Keith C.C. Chan et al. (2001) *Mining Fuzzy Rules in A Donor Database for Direct Marketing by A Charitable Organization*. The Hong Kong Polytechnic University Hung Hom, Kowloon, Hong Kong.

Lance, G. and Williams W. (1999). *A general theory of classification sorting strategies*. *Computer Journal*, 9 : pp 373-386.

Michel de Ruiter (1999). *Bayesian classification in data mining.* Theory and practice BWI stageverslag.

Olson, C. (1995). *Parallel algorithms for hierarchical clustering. Parallel Computing*, 21 : pp. 1313-1325.

Pi-Shen, D. (1994). *Using Case-Based reasoning For Decision Support*. IEEE.

Shan, C. (1998). *Statistical approach to predictive modeling in large databases*. Simon Fraser University: MSc Thesis.

Shatsky, M., Zipora Y. Fligelman, Ruth Nussinov, and Haim J. Wolfson. (2000). *Alignment of flexible protein structures.* In proceedings of international conference on intelligent systems for Molecular Biology.

Sheng-Chai, C. et al. (1999). *A forecasting approach for stock index future using Grey theory and neural networks*. IEEE.

Srikant, R. and Agrawal, R. (1996). *Mining Quantitative the ACM SIGMOD.* International Conf on Management of data, Montreal, Canada : pp.1-12".

Wei, W. (1999). *Predictive modeling based on classification and pattern matching method*. Simon Fraser University: MSc Thesis.

Wuthrich, B. et al. (1998). *Daily stock market forecast from textual web data*. IEEE.

Yager, R.R. (1991). *On Linguistic Summaries of data*. In G. Piatetsky-Shapiro and W.J. Frawley (Eds.), Knowledge Discovery in Databases, Menlo Park CA: AAAI/MIT press : pq 347-363.

Zaiane, R. (1999) *Introduction to data mining*. CMPUT690 Principles of Knowledge Discovery in Databases.

Zhao, Y. and Karypis, G. (2002). *Comparison of agglomerative and partitional document clustering algorithms.* In SIAM(2002) workshop on Clustering High-dimensional Data and Its Applications.

Zhao, Y. and Karypis, G. (2000) *Clustering in life sciences.* Department of computer of science, university of Minnesota, Minneapolis, MN 55455.