REFINEMENT OF GENERATED WEIGHTED FUZZY PRODUCTION RULES BY USING FUZZY NEURAL NETWORKS FOR STOCK MARKET PREDICTION

[1] Mohd Noor Md Sap, [2] Rashid Hafeez Khokhar
Faculty of Computer Science and Information Systems
University Technology Malaysia,
81310 Skudai, Johor, Malaysia
Tel: (607)-5532419, Fax: (607) 5565044
[1] mohdnoor@fsksm.utm.my, [2] rashid@siswa.utm.my

ABSTRACT

One of the most important problems in the modern finance is finding efficient ways of summarizing the stock market data that would allow one to obtain useful information about the behavior of the market. The trader's expectations to predict stock markets are seriously affected by some uncertain factors including political situation, oil price, overall world situation, local stock markets etc. Therefore, predicting stock price movements is quite difficult. In this paper, the new technique to predict stock market is presented for the refinement of generated Weighted Fuzzy Production Rules (WFPR's) by using fuzzy neural networks. The existing techniques to generate WFPR's are suffered from the problem of low accuracy of classifying or recognizing unseen examples. The reasons for having these problems are 1) the WFPRs generated are not powerful enough to represent the domain knowledge, 2) the techniques used to generate WFPRs are pre-matured, ad-hoc or may not be suitable for the prediction problem, and 3) further refinement of the extracted rules has not been done. In this paper, we look into the solutions of the above problems by 1) enhancing the representation power of WFPRs by including local and global weights, 2) developing a fuzzy neural network (FNN) with enhanced learning algorithm, and 3) using this FNN to refine the local and global weights of WFPRs for stock market prediction. By experiment our method with some stock markets examples has found a better accuracy in classifying unseen samples without increasing the number of extracted WFPRs.

Keywords: Data Mining, Stock Market Prediction, Time Series, Classification, Clustering, Fuzzy Decision Tree, Fuzzy Logics.

1. Introduction

The stock market is a rather complicated system, and good predictions for its developments are the key to successful trading. Traders must predict stock price movements in order to sell at top range and to buy at bottom range. As stock trading is a very risky business (Torben and Lund, 1997), it is necessary to evaluate the risks and benefits before entering into any trading. The key to realize high profits in stock trading is to determine the suitable trading time when the risk of trading should be minimum. Many attempts have been made for meaningful prediction of stock market by using data mining and statistical techniques like Support Vector Machine (Alan Fan et al., 2001; Haiqin, 2002), Neural Networks (Xiaohua et al., 2003; Raymond, 2004), Linear and Non-linear models (Weiss, E. 2000; Chinn et al., 2001) and Classification (Agrawal R. et al., 2000; Han, J and Pei, 2000). However, these techniques to predict stock market real time data are yet to be achieved good classifiers (model).

In this paper, the methods for refinement of generated weighted fuzzy production rules are presented. These WFPR,s are extracted from our proposed predictive Fuzzy Decision Tree (FDT) algorithm (Khokhar and Noor 2004). In particular, the proposed predictive FDT algorithm is based on the concept of degree of importance of attributes contributing to the classification. This concept was firstly proposed by Pawlak, (1991) while investigating the reduction of knowledge. It was used to extract the minimum indispensable part of equivalent relations. Later on Wang et al., (2001) extended this concept to a fuzzy case and then used it to select the expanded attribute at a considered node while generating fuzzy decision trees. The same idea was extended in (Khokhar and Noor 2004) for the construction of fuzzy decision trees and applied to stock market analysis.

In predictive FDT, every path from root node to leaf node presents Weighted Fuzzy Production Rules (WFPR's) and the strength of the rule can be measured by a parameter referred as Certainty Factor (CF). In this research, weights are assigned to every arc from root to leaf node according to the degree of importance of the attribute contributing to classification. WFPR's can be extracted from the predictive FDT. These rules are large in numbers. Therefore, in order to mine and enhance the representation power of FPRs, the knowledge representation parameters (KRPs) such as local and global weights are included in these FPRs. These local and global weights had been proposed by Yeung et al. (1997 and 1998). A fuzzy neural network (FNN) is proposed to refine or tune the local and global weights of FPRs. A set of weighted FPRs (WFPRs) obtained will be more optimal and accurate in recognizing and classifying unseen samples. It is because those weights with values more or less equal to zero could be deleted so that smaller number of propositions in the antecedent of WFPRs (so-called simple WFPRs) is generated.

Furthermore, the extracted WFPRs with local and global weights capturing more domain experts' knowledge will have higher accuracy in solving recognition and classification problems. A FNN offers advantages of allowing us to map these KRPs (local and global weights) of FPRs into the connection weights of a FNN and with a modified back-propagation (BP) learning algorithm, we are able to tune, refine and even acquire these parameters. In (Hong and Chen, 1999) eleven categories of FNNs have been identified. The FNN used in this paper is similar to the fuzzy-like neuro model where a neural network is used to represent fuzzy rules. The difference is that our FNN is used to represent WFPRs which could be refined or tuned so that approximately optimal rules and higher testing accuracy could be obtained. In (Hiraga, 1998; Jang, 1993) two FNN models are proposed to solve parameters tuning of fuzzy membership functions. The problem settings of ANFIS in (Jang, 1993) is that it is used to represent three types of fuzzy inference systems used in fuzzy controlled systems whose rules are parallel in nature, whereas our proposed method could handle multi-level WFPRs and extends the traditional method to a more general one.

The following sections explore most of the steps in the process. Section 2 discusses the existing standard fuzzy decision tree techniques. Predictive Fuzzy Decision Tree is the subject of section 3 and weighted fuzzy production rules of section 4. Section 5 presents the back-propagation and the convergence of the fuzzy learning rules. Discussion will be on section 6 and finally, the conclusion is presented in section 7.

2. Existing Fuzzy Decision Trees

There have been many methods for constructing decision trees from collection of crisp examples (Quinlan, 1993; Agarwal, 2000; Han, J et al., (2000). The decision trees generated by these methods are useful in building knowledge-based expert systems. Due to the rapid growth of uncertainty in the knowledge-based systems, it is found that using crisp decision trees alone to acquire imprecise knowledge is not enough. Uncertainty such as fuzziness and ambiguity should be incorporated into the process of learning from examples such as decision tree induction. These decision tree induction techniques introduce fuzzy decision tree generation suggested by many authors (Umanol, 1994; Dong and Kothari, (2001); Quinlan, 1993). The fuzzy decision tree with minimal number of leaf-nodes is usually thought to be optimal. However, the optimal (fuzzy) decision tree generation has been proved to be NP-hard. Therefore, the research on heuristic algorithms is necessary to mine knowledge from hidden pattern from huge databases. The heuristic information used in constructing fuzzy decision trees can be various and each heuristic may be better than the other in some aspects. Mainly three heuristics are popular for generating fuzzy decision trees among the existing one. These heuristics are based on

1) classification information-entropy to select expanded attributes (Umanol, 1994; Dong and Kothari, 2001)
2) classification ambiguity to select expanded attributes (Yuan and Shaw, 1995)
3) degree of importance of the attribute contributing to the classification to select the expanded attributes (Wang at al., 2001; Yeung, 1999; 2002)

One powerful technique for generating crisp decision trees called ID3. Quinlan proposed the earlier version of ID3, which is based on minimum classification information-entropy to select expanded attributes, in (Quinlan, 1986). As the increasing uncertainty incorporated into the knowledge-based system, the fuzzy version of ID3 has been suggestion by several authors e.g. (Umanol, 1994; Dong and Kothari, 2001) Classification information-entropy is based on probabilistic models i.e., Shannon Entropy. It is a well known concept to describing probabilistic distribution's uncertainty. Subsequently, this concept was extending to describe the possibilistic distribution uncertainty, called fuzzy entropy. The typical extension was given in (Yuan and Shaw, 1995). Yuan and Shaw, (1995) refers a possibilistic distribution to a vector whose components are in [0, 1] while a probabilistic distribution is possibility distribution with the property that the sum of all components is equal to 1.

For a probabilistic distribution, each component considered as a probability with which the corresponding event occurs. A possibilistic distribution usually considered as a fuzzy set vector, and each component of the vector, i.e., the membership degree regarded as the possibility with which the corresponding event occurs. For the difference and consistency between probability and possibility, one can refer to Zadeh's, (1999). The difference between the uncertainties described by the entropy of a probabilistic distribution and described by the fuzzy entropy of a possibilistic distribution is that the former attains its maximum at all components being 0.5 but the latter does not. Fuzzy ID3 uses the fuzzy the entropy of a possibilistic distribution.

Another existing powerful heuristic algorithm to generate fuzzy decision tree was introduced by Yuan and Shaw's (1995). Instead of using minimum fuzzy entropy, this heuristic (Yuan and Shaw's, 1995) used the minimum classification ambiguity to select expanded attributes. The classification ambiguity is called non-specificity (also called U-uncertainty). Recently, Wang et al., (2001) proposed another heuristic, which uses the maximum classification importance of attribute contributing to its consequents to select the expanded attributes. This concept firstly proposed by Pawlak, (1991), while investigating the reduction of knowledge. It was used to extract the minimum indispensable part of equivalent relations. In Wang et al., (2001) extended this concept to a fuzzy case and then used it to select the expanded attribute at a considered node while generating fuzzy decision trees. In Wang et al., (2001) method, aims to search for an attribute that its average degree of importance contributing to the classification attains maximum, i.e., selecting such an integer k_o (the k_o th attribute) that $P_{k_o} = \max_{1\le k\le n} P_k$.

Proposition 1: For fixed k and i, consider two functions

$$Entr_i^{(k)} = -\sum_{j=1}^{m} p_{ij}^{(k)} \log_2 p_{ij}^{(k)} \quad (1)$$

$$Ambig_i^{(k)} = \sum_{j=1}^{m} (\pi_{ij}^{(k)} - \pi_{i,j+1}^{(k)}) \text{ In } j. \quad (2)$$

within the area $\{0 \le p_{ij}^{(k)} \le 1 \mid j = 1,2,...m\}$, the first function attains its minimum at a vector of which each component is either 0 or 1, and the second attains its minimum at a vector in which one component is 1 but the other components are 0. Here make the appointment $0\log_2 0 = \lim_{x\to 0}(x\log_2 x) = 0$

In which $(\pi_{i1}^{(k)}, \pi_{i2}^{(k)}, \ldots\ldots \pi_{im}^{(k)})$ with descending order $\pi_{i,m+1}^{(k)} = 0$ is a permutation of $(\tau_{i1}^{(k)}, \tau_{i2}^{(k)}, \ldots\ldots, \tau_{im}^{(k)})$ which is a normalization of $(p_{i1}^{(k)}, p_{i2}^{(k)}, \ldots\ldots, p_{im}^{(k)})$, i.e., $\tau_{ij}^{(k)} = p_{ij}^{(k)} / \max_j p_{ij}^{(k)}$. Where $Entr_i^{(k)}$ and $Ambig_i^{(k)}$ show the classification information-entropy and classification ambiguity for each (k) respectively. $p_{ij}^{(k)}$ shows the probability of classification attributes.
(for proof see Wang et al., 2001).

This proposition indicates that fuzzy ID3 aims averagely to search the expanded attribute with relative frequencies as close to 0 or 1 as possible while Yuan and Shaw's method aims averagely to search one with relative frequencies as close to 0 (except for the maximum frequency) as possible.

It is easy to see from Proposition 1 that the minimum of the function $Ambig_i^{(k)}$ implies the minimum of the function $Entr_i^{(k)}$ and the inverse is invalid. Particularly, if $\{0 \le p_{ij}^{(k)} \le 1 \mid j = 1,2,...m\}$ is a probabilistic distribution then the two minima are equivalent.

Proposition 2: From proposition 1, function $Entr_i^{(k)}$ and $Ambig_i^{(k)}$ attains their maxima at $p_{i1}^{(k)} = ... = p_{im}^{(k)} = e^{-1}$ and $p_{i1}^{(k)} = ... = p_{im}^{(k)} = 1$ respectively (for proof see Wang et al., 2001) .

Proposition 1 implies that when all frequencies are 1 the fuzzy entropy is 0 but the non-specificity attains maximum. That indicates such a situation in which using fuzzy ID3 techniques, Yuan, and Shaw's select different expanded attributes. However, proposition 1 indicates that if Yuan and Shaw's technique selects an expanded attributes with very small value of $Ambig_i^{(k)}$, then fuzzy ID3 select the same expanded attribute at the same non-leaf node. Moreover, for the frequency distribution the smaller the non-specificity, the closer it is to a probabilistic distribution. Thus proposition 1 intuitively indicates that the two techniques are likely to select the same expanded attribute while the non-specificity is small. Particularly, if the two techniques select the same expanded attribute at the root with small fuzzy entropy and non-specificity, then the two techniques for selecting expanded attributes are gradually consistent. That is the expanded attribute selection of fuzzy ID3 techniques, to some extent is identical to the one of Yuan and Shaw's technique. These two techniques show the expanded attributes of the two heuristics are the same at most non-leaf nodes.

In Wang et al., (2001) fuzzy decision tree, which is based on the maximum degree of importance of attribute contributing to the fuzzy classification. It aims, on the considered node with several attributes to be chosen to select an attribute whose contribution to classification is maximal.

Proposition 3: Under an assumption of uniform distribution, either maximum or minimum degree of importance implies maximum fuzzy entropy when the classification is crisp and implies maximum non-specificity when the classification is fuzzy. The uniform distribution assumption is formulated in the proof (for proof see Wang et al., 2001)

Proposition 3 indicated that the relation between (Wang et al., 2001) FDT and the other is very complicated. It implicitly proposes that there exists such an attribute at which the maximum (minimum respectively) degree of importance and maximum (minimum respectively) entropy can be achieved simultaneously at a node.

These three heuristic has some strength and weakness, Table 1 presents the summary of comparative results in terms of complexity, applicability, comprehensibility, learning accuracy, handling of classification ambiguity and robustness. With regard to the complexity of the fuzzy decision tree, the relation among the three heuristics is non-deterministic, dependent mainly on the expanded attribute selection. First consider the computation effort while expanding a non-leaf node and then consider the size of trees. The number of leaves is an important index to measure the size of a tree. Obviously bigger numbers of leaves are creating more complexity in construction of FDT. While expanding a non-leaf node and then consider the size of trees. The following assertion is valid:

Table 1: Summary of analytic comparison of fuzzy

	Fuzzy ID3 (Umanol, 1994)	Yuan & Shaw's (1995)	Wang et al., (2001)
Heuristic Information	Fuzzy entropy	Non-specificity of possibility distribution	Importance of attributes contributing to classification
Criterion of expanded attribute	Minimum fuzzy entropy	Minimum non-specificity	Maximum importance degree
Expanded attribute used in the tree	Partially same as Yuan and Shaw's heuristic	Partially same as fuzzy ID3 heuristic	Not same as others
Reasoning mechanism	Mini-maxi operation of memberships	Multiplication-addition operation of memberships	Weighted average of similarity
Comprehensibility of tree	Lower	Quite Higher	Higher
Reasoning accuracy	Medium	Less	Greater
Complexity (time, space)	Same as Yuan and Shaw's	Same as Fuzzy ID3	Greater than both
Robustness	Medium	Greater	Less
Scalability	Less	Medium	Greater
Prediction	No	No	No

CE (fuzzy ID3) $\approx$ CE (Yuan and Shaw's method) $\leq$ CE (Yeung and Wang method)

Where CE represents the term Computation-Efforts that refers mainly to the number of times of operations such as addition, multiplication, max, min, etc.

The number of leaves is an important index to measure the size of a tree. The generic standard of leaf-node is a frequency-threshold, which is node (fuzzy set) regarded as a leaf if the relative frequency of some class at the node exceeds a given threshold. Fuzzy rules extracted from Yuan and Shaw's tree includes only one parameter CF, one can see that the comprehensibility of Yuan and Shaw's tree is better than that of Wang et al., tree which is turn better than that of fuzzy ID3, that is

Comprehensibility (fuzzy ID3) $\leq$ Comprehensibility (Yeung and Wang, tree) $\leq$ Comprehensibility (Yuan and Shaw's)

The last row in Table 1 is presenting the parameter predictions that have not been considered by any researchers. Therefore, in this research parameter prediction is also considered for the construction of FDT.

3. Predictive Fuzzy Decision Tree (FDT)

Decision trees are a well-k wn and widely used method for classification problems. For handling numerical attributes or even for numerical prediction. The traditional decision trees based on crisp predicates are not suitable. Through the usage of fuzzy predicates for different types of attributes not only the expressive power of decision trees can extend but it also allows creating models for numerical attributes in a very natural manner. For this purpose, some attempts are being made for the last decade and introduce fuzzy decision tree for numerical prediction but the existing fuzzy decision tree is still suffering s me problems like complexity, comprehensibility of tree, over-fitting, robustness. scalability, and mining useful fuzzy rules from numerical attributes.

The predictive FDT is based on the maximum degree of attribute contributing to the fuzzy classification (Wang et al., 2001). It aims, on the considered node with several attributes to be chosen to select an attribute whose contribution to classification is maximal. Before starting the construction of predictive FDT it is important to understand some attributes information that is used through out this paper. In this research, the three basic attribute P_O =price of open, P_C =

price of close, V = volume are considered from stock market data. In addition, some other factors are also considered that can affect stock market after 5 minutes of period. These factors are as follows:

P_{OP} = Oil Price, P_{ND} =Natural Disaster, P_{PS} =Political Situation, P_{NP} =New Policies, P_{NB} =New Budget, P_{OC} =Other Companies, P_{LSM} =Local Stock Markets, P_{OWS} =Overall World Situation.

These factors are calculated for different associated stocks from KLSE, NYSE and LSE after analyzing the daily stock prices in different periods of times. Table 2 presents an example of real time data sample for every 5 minutes of periods during 9:30 to 11:05. The last column in table 2 is showing the target classification attribute with name primary signals. The primary signals are defined after evaluating the final trends of every stock price with in 5 minutes of periods. Particularly, the five basic attributes i.e., price of open, price of low, price of high, and price of close are used to calculate primary signals.

Predictive FDT consists of three steps including kernel K-means clustering, fuzzification of numerical numbers, and finally fuzzy decision tree algorithm. The kernel K-means clustering algorithm (Noor and Awan, 2004) takes into account the neighborhood and thus also gives smoothing effect. It is used to compresses the data set and to find the center concerning the data set. For the fuzzification of numerical numbers usually a fuzzy attribute can take many values if the representation of the fuzzy value is given directly by a membership degree.

Like numerical attributes, the range of such a fuzzy attribute can be described as the interval $[1,0]^M$ where M is the dimension. For a given set of membership functions, the intention in this research is to find several new fuzzy sets, which are regarded as clustering result to reasonably describe this set of membership functions. Finally, proposed predictive FDT algorithm is presented. In the following subsections several basic concepts involved in predictive FDT are presented.

When an attribute is categorical, the fuzzification is quite straightforward. Each of the possible values of the attribute is just treated as fuzzy subset. In this case, the membership value in a fuzzy subset is either 0 or 1. For numerical attributes, the kernel k-means clustering algorithm (Noor and Awan, 2004) to cluster the attribute values into 2 or 3 clusters representing two T_1, T_2 or three T_1, T_2, T_3 linguistic terms respectively. The choice of the number of clusters is arbitrary though; the notation guided us that a value can typically be thought of as being low, average and high. Memberships have been generated based on a triangular membership function and the 2 or 3 cluster centers (a_1, a_2 with $a_1 < a_2$) or (a_1, a_2, a_3 with $a_1 < a_2 < a_3$) respectively obtained through kernel-kmeans clustering. The following sections present the explanation of these clusters with example and how these clusters have been used in triangular membership functions.

Table 2: 20 real time (5-Minute Bars) examples of stock market data

No	Data and Time	P_O	P_C	V	P_{OP}	P_{ND}	P_{PS}	P_{NP}	P_{NB}	P_{OC}	P_{LSM}	P_{OWS}	Primary Signal
1	08/02/04 9:30	39.8	39.84	79,100	0	-0.03	0.03	0.08	0	0.75	-0.06	-0.36	0.0090
2	08/02/04 9:35	39.84	39.76	19,000	0	-0.35	-0.07	0	0	0.58	-0.26	0.02	0.0150
3	08/02/04 9:40	39.74	39.8	30,800	0	0.24	-0.09	-0.01	0	-0.32	-0.25	0.26	-0.0006
4	08/02/04 9:45	39.8	39.85	64,200	0	-0.45	-0.03	0.09	0	0.85	-0.65	0.32	0.0292
5	08/02/04 9:50	39.85	39.97	31,900	0	-0.12	-0.04	-0.04	0	-0.54	0.09	-0.32	-0.0222
6	08/02/04 9:55	39.97	39.88	39,300	0	-0.34	-0.25	-0.49	0	-0.04	-0.62	0.25	-0.0238
7	08/02/04 10:00	39.87	39.9	43,800	0	0.48	0.18	-0.08	0	-0.39	-0.85	-0.69	-0.0806
8	08/02/04 10:05	39.9	39.86	27,100	0	0	0.25	0	0	-0.07	0.22	0.32	-0.0237
9	08/02/04 10:10	39.86	39.78	8,800	0	0	0.19	0	0	0.26	-0.02	0.01	0.0245
10	08/02/04 10:15	39.77	39.68	13,500	0	0	-0.78	0	0	-0.24	-0.55	-0.33	0.0886
11	08/02/04 10:20	39.67	39.71	14,800	0	-0.26	-0.19	0	0	0.02	0.02	0.08	0.0706
12	08/02/04 10:25	39.72	39.77	17,400	0	-0.24	0.18	0	0	-0.25	0.38	-0.26	0.0598
13	08/02/04 10:30	39.78	39.81	26,100	0	-0.19	0.08	0.05	0	0.02	-0.65	0.38	-0.0451
14	08/02/04 10:35	39.83	39.85	26,200	0	-0.04	-0.08	-0.25	0	-0.05	-0.58	0.28	-0.0949
15	08/02/04 10:40	39.85	39.89	34,100	0	-0.07	-0.09	-0.48	0	0.07	-0.23	0.58	-0.0750
16	08/02/04 10:45	39.92	39.9	23,900	0	-0.09	-0.47	-0.15	0	-0.08	-0.28	-0.38	-0.0425
17	08/02/04 10:50	39.91	39.93	30,800	0	-0.08	0.19	0.35	0	-0.87	-0.26	0.85	-0.0028
18	08/02/04 10:55	39.93	39.94	14,300	0	0.04	0.78	-0.18	0	0.26	-0.36	0.11	-0.0083
19	08/02/04 11:00	39.96	39.92	14,400	0	0.15	-0.32	0.34	0	0.98	-0.47	-0.23	-0.0068
20	08/02/04 11:05	39.92	39.91	7,900	0	-0.04	-0.48	-0.19	0	0.27	-0.15	-0.52	-0.0168

3.1 Fuzzification of Numerical Number

Fuzzification is a process of fuzzifying numerical numbers into linguistic terms, which is often used to reduce information overload in human decision-making process. Linguistic terms are simple forms of fuzzy values but generally their membership functions are unknown and need to be determined.

Given a fuzzy set of records, D, each of which consists of a fuzzy set of attributes $I = \{a_1, a_2, ...a_n\}$, where $a_r, r = 1, ...n$, can be quantitative or categorical. Consider a set of examples $\{e_1, e_2, ...e_N\}$, which is defined as the universe of discourse X (in short X is denoted by $\{1,2, ..., N\}$). Let $T^{(1)}, ..., T^{(n)}$ and $T^{(n+1)}$ be a set of fuzzy attributes where $T^{(n+1)}$

denotes a classification attribute. Each fuzzy attribute $T^{(j)}$ consists of a set of linguistic terms $L(T^{(j)}) = \{L_1^{(j)},......., L_{mj}^{(j)}\}(j = 1,2,....., n+1)$. All linguistic terms are defined on the same universe of discourse X. The value of the ith example e_i with respect to the jth attribute, denoted by μ_{ij}, is a fuzzy set defined on $L(T^{(j)})(i = 1,......N, j = 1,2,....n+1)$. In other words, fuzzy set μ_{ij} has a form of $\mu_{ij}^{(1)} / L_1^{(j)} + \mu_{ij}^{(2)} / L_2^{(j)} + ... + \mu_{ij}^{(mj)} / L_{mj}^{(j)}$ where $\mu_{ij}^{(k)}$ denote the corresponding membership degree $k = 1,2,.......m_j$. The fuzzy sets and its linguistic terms are as follows:

$$C_{P_O} = \text{change in price of open} = \{Low, Average, High\}$$
$$C_{P_C} = \text{change in price of close} = \{Low, Average, High\}$$
$$C_V = \text{change in volume} = \{Small, Medium, Large\}$$
$$P_{OP} = \text{Oil Price} = \{Decrease, Stable, Increase\}$$
$$P_{ND} = \text{Natural Disaster} = \{No, Yes\}$$
$$P_{PS} = \text{Political Situation} = \{Clear, NotClear\}$$
$$P_{NP} = \text{New Policies} = \{LowAffect, NoAffect, HighAffect\}$$
$$P_{NB} = \text{New Budget} = \{Fair, NotFair\}$$
$$P_{OC} = \text{Other Companies} = \{Low, Medium, High\}$$
$$P_{LSM} = \text{Local Stock Markets} = \{LowAffect, NoAffect, HighAffect\}$$
$$P_{OWS} = \text{Overall World Situation} = \{Bad, Good\}$$

Fuzzy sets can have a variety of shapes. However, a triangle or a trapezoid can often provide an adequate representation of the expert knowledge and at the same time significantly simplifies the process of computation. Figure 1(a), Figure 1(b), Figure 1(c), and Figure 1(d) are representing an example of set of 4 linguistic terms (oil price, change in close, local stock market and primary signals) for fuzzy attribute. Horizontal direction presents universe of discourse and vertical direction presents degree of membership $[0,1]$.

Let X be a given data set, which is clustered into k linguistic terms $T_j, j = 1,2,......, k$. For simplicity, it is assumed that the type of membership to be triangular as follows:

$$T_1(x) = \begin{cases} 1 & x \le a_1 \\ \dfrac{a_2 - x}{a_2 - a_1} & a_1 < x < a_2 \\ 0 & x \ge a_2 \end{cases} \qquad (3)$$

$$T_k(x) = \begin{cases} 1 & x \ge a_k \\ \dfrac{(x - a_{k-1})}{(a_k - a_{k=1})} & a_{k-1} < x < a_k \\ 0 & x \le a_{k-1} \end{cases} \qquad (4)$$

$$T_j(x) = \begin{cases} 0 & x \ge a_{j+1} \\ (a_{j+1} - x)/(a_{j+1} - a_j) & a_j \le x < a_{j+1} \\ (x - a_{j-1})/(a_j - a_{j-1}) & a_{j-1} \le x < a_j \\ 0 & x < a_{j-1} \end{cases} \quad 1 < j < k \qquad (5)$$

The only parameters to be determined are the k centers $\{a_1, a_2,......a_n\}$. A simple method to determine these centers is fuzzy clustering based on KMeans. For example consider the numerical attribute "Change in open" of the group of examples as shown in Table 1 by choosing k =3 and suppose the learning rate $\alpha = 0.02$, three values of center a_1-0.02, $a_2 = -0.13$, a_3=0.01 after 135 time of iteration. The membership functions for attribute "change in open" with linguistic terms $\{Low, Average, High\}$ are described as follows

$$Low = T_1(x) = \begin{cases} 1 & x \le -0.02 \\ \dfrac{-0.13 - x}{-0.13 - (-0.02)} & -0.02 < x < -0.13 \\ 0 & x \ge -0.13 \end{cases}$$

$$Average =$$

$$T_2(x) = \begin{cases} 0 & x \ge 0.01 \\ (0.01 - x)/(0.01 - (-0.13)) & -0.13 \le x < 0.01 \\ (x - (-0.02))/((-0.13) - (-0.02)) & -0.02 \le x < -0.13 \\ 0 & x < -0.02 \end{cases} \qquad 1 < j < k$$

$$High = T_3(x) = \begin{cases} 1 & x \ge 0.01 \\ \dfrac{(x - (-0.13))}{(0.01 - (-0.13))} & -0.13 < x < 0.01 \\ 0 & x \le -0.13 \end{cases}$$

It is obvious that the three linguistic terms can be described as Low, Average and High. The second column of Table 3 shows the membership degrees of the attribute Change in open belonging to the three membership functions. When the values of an attribute are fuzzy, the values can be written as one of the two forms: simple linguistic terms and membership functions. The information provided in a membership function is more concrete than those provided in linguistic terms but the meaning of membership function is not clear. Linguistic terms are simple forms of fuzzy values but their degree of membership is unknown and need to be determined. Degree of membership for selective 5 attributes by using the Table 2 is determined as in Table 3.

Table 3: After training real time examples of stock market with fuzzy representation

No	C_{P_O}			P_{ND}		P_{NB}		P_{OWS}		P_{LSM}			Primary Signal		
	Low	Med	High	No	Yes	Fair	NotFair	Bad	Good	Low Affect	Med Affect	High Affect	Down	Hold	Up
1	0.10	0.55	0.45	0.45	0.55	0.30	0.70	0.94	0.06	0.00	0.14	0.86	0.00	0.53	0.47
2	0.03	0.10	0.87	0.85	0.15	0.67	0.33	0.88	0.22	0.58	0.25	0.17	0.00	0.67	0.33
3	0.71	0.10	0.19	1.00	0.00	0.00	1.00	0.70	0.30	0.81	0.19	0.00	0.49	0.33	0.18
4	0.09	0.70	0.21	1.00	0.00	0.20	0.80	0.27	0.73	0.00	1.00	0.00	0.25	0.07	0.68
5	0.42	0.30	0.48	0.92	0.08	0.46	0.54	0.09	0.91	0.73	0.27	0.00	0.30	0.20	0.50
6	0.70	0.00	0.30	0.42	0.58	0.70	0.30	0.94	0.06	0.12	0.00	0.88	0.30	0.70	0.00
7	0.00	0.00	1.00	0.54	0.46	0.85	0.15	0.82	0.18	0.00	0.00	1.00	0.41	0.59	0.00
8	1.00	0.10	0.00	0.60	0.40	0.28	0.72	0.00	1.00	0.46	0.33	0.23	0.16	0.40	0.46
9	0.43	0.11	0.46	0.67	0.33	0.05	0.95	1.00	0.00	0.10	0.00	0.90	0.95	0.00	0.05
10	0.45	0.55	0.00	0.91	0.09	0.32	0.68	0.50	0.50	0.00	0.20	0.80	0.69	0.20	0.11
11	0.00	0.69	0.31	0.20	0.80	0.17	0.83	0.44	0.56	0.00	0.67	0.33	0.80	0.20	0.00
12	0.00	1.00	0.00	0.80	0.20	0.48	0.52	1.00	0.00	0.00	1.00	0.00	0.00	1.00	0.00
13	0.20	0.80	0.00	0.51	0.49	0.24	0.76	0.20	0.80	0.00	0.25	0.75	0.49	0.00	0.51
14	0.41	0.38	0.21	0.78	0.22	1.00	0.00	0.82	0.18	0.74	0.20	0.06	0.00	0.82	0.18
15	0.50	0.30	0.20	0.54	0.46	0.18	0.82	0.19	0.81	0.00	0.64	0.36	0.00	0.54	0.46
16	0.50	0.10	0.40	0.41	0.59	0.80	0.20	0.98	0.00	0.00	0.21	0.79	0.00	0.80	0.20
17	0.05	0.30	0.65	0.50	0.50	0.60	0.40	0.05	0.95	0.25	0.00	0.75	0.65	0.00	0.35
18	0.34	0.35	0.31	0.38	0.62	0.40	0.60	0.82	0.18	0.74	0.18	0.08	0.20	0.52	0.20
19	0.50	0.44	0.01	0.94	0.06	0.15	0.85	0.23	0.77	0.00	0.64	0.36	0.34	0.10	0.56
20	0.10	0.01	0.89	0.10	0.90	0.30	0.70	0.98	0.02	0.21	0.10	0.79	0.10	0.30	0.60

3.2 Predictive Fuzzy Decision Tree Algorithm

Let X represent a discrete universe of discourse, $F(X)$ denote the set of all fuzzy subsets defined on X. For $X = \{e_1, e_2, ...e_N\}$ and $T \in F(X)$, T can be represented as $T = T(e_1)/e_1 + ... + T(e_N)/e_N$, and $M(T) = \sum_{i=1}^{N} T(e_i)$ denote the cardinality of T (Wang et al., 2001).

By using the same notations in section 3.1, consider a test node S having n attributes $T^{(1)}, T^{(2)}, ..., T^{(n)}$ to be selected. For each $k (1 \le k \le n)$, the attribute $T^{(k)}$ takes m_k fuzzy subsets (linguistic terms), $L_1^{(k)}, L_2^{(k)}, ..., L_{mk}^{(k)}$. $T^{(n+1)}$ denotes the classification attribute, taking values $L_1^{(n+1)}, L_2^{(n+1)}, ..., L_m^{(n+1)}$. For each attribute value (fuzzy subset), $L_i^{(k)} (1 \le k \le n, 1 \le i \le m_k)$, its relative frequencies concerning the j^{th} fuzzy class $L_i^{(n+1)} (1 \le j \le m)$ at the considered nonleaf node S is defined as

$$p_{ij}^{(k)} = M(L_i^{(k)} \cap L_j^{(n+1)} \cap S) / M(L_i^{(k)} \cap S) \quad (6)$$

The weight of the *ith* value $L_i^{(k)}$ is defined as

$$w_i = M(S \cap L_i^{(k)}) / \sum_{j=1}^{mk} M(S \cap L_j^{(k)}) \quad (7)$$

Definition 1:

Let $\mu_{ik} = \mu_{ik}^{(1)},......\mu_{ik}^{(j)},......\mu_{ik}^{(mj)}$ be the value of the *ith* example with respect to the *kth* attribute, $w_{ik} = \mu_{ik}^{(1)},......\mu_{ik}^{(j)},......\mu_{ik}^{(mj)}$, v_i be the value of the *ith* example with respect to the classification $(1 \le i \le N, 1 \le k \le n)$, i.e., v_i is a fuzzy set define on $L_1^{(n+1)}, L_2^{(n+1)},....., L_m^{(n+1)}$. (the set of linguistic terms of the classification attribute $T^{(n+1)}$), SM be a selected similarity measure, $\lambda_{ip}^{(j)} = \wedge_{q \ne k}(SM(\mu_{iq}, \mu_{pq})) \wedge SM(w_{ik}, w_{pk})$, (where $\wedge$ denotes minimum, $i \ne p$), and $\sigma_{ip} = SM(v_i, v_i)(i \ne p)$. Then, for the *kth* attribute $T^{(k)}$ $(1 \le k \le n)$, the degree of importance of its *jth* linguistic term $L_j^{(k)}(1 \le j \le m_k)$ contributing to the classification is defined as

$$\theta_j^{(k)} = \frac{1}{N(N-1)} \sum_i \sum_{p \ne i} g^+(\lambda_{ip}^{(k)} - \sigma_{ip}), \qquad (8)$$

where $g^+(x) = \begin{cases} x & x > 0 \\ 0 & x \le 0 \end{cases}$.

Definition 2:

The averaged degree of importance of the *kth* attribute $T^{(k)}$ is defined as $P_k = \sum_{j=1}^{mk} w_j \theta_j^{(k)}$, in which w_j is defined by equation (7).

Proposed approach aims to search for an attribute such that its average degree of importance contributing to the classification attains a maximum, i.e., selecting an integer $k_\circ$ (the $k_\circ th$ attribute) so that $P_{k\circ} = Max_{1 \le k \le n} P_k$ where P_k is given by definition 2.

According to the above heuristic, a FDT can be generated by using training a set of data. Before training the initial data, the α cut is usually used for the initial data (Wang et al., 2001). The purpose of using α cut is to reduce the fuzziness. The α cut of a fuzzy set L is defined as

$$L_\alpha(x) = \begin{cases} L(x) & L(x) \ge \alpha \\ 0 & L(x) < \alpha \end{cases} \qquad (9)$$

When α is set in the interval $(0, 0.5]$, procedure for generating a predictive FDT is described as follows:

Step 1.
Given the cut-standard $\alpha(\alpha \in (0,1))$ and the leaf-standard $\beta(\beta \in (0,1))$.

Step 2.
Use α to cut the initial data set. More specifically, each membership degree less than α is changed to 0 and all others remains unchanged.

Step 3.
Consider the root node (1,1,........,1) as the first candidate node.

Step 4.
Randomly select a non-leaf candidate node. For any attribute which has not been used in forefather nodes of this node, compute P_k.

Step 5.
Select an attribute with average maximum importance to the classification (given by definition 2) as the expanded attribute. According to the expanded attribute, generate son-nodes of the non-leaf candidate node. These son-nodes are considered as new candidate nodes.

Step 6.

For each of these son-nodes, if the relative frequency $P_{ij}^{(k)}$ (given by equation 6) of a certain class exceeds β or the membership sum of the considered son-node is less than a small positive number, then this son-node is labeled leaf.

Step 7.

If all nodes are leaves then stop, else go to Step 4.

4. Weighted Fuzzy Production Rules (WFPRs)

The WFPRs generation from fuzzy decision tree is an extended form of fuzzy production rules (FPR) proposed by (Yaung et al., 1994). WFPRs defined here is similar to the conventional production rules with the exception that fuzzy values such as "fat" or "small" are allowed in the propositions. A weight is assigned to each proposition in the antecedent part, and a certainty factor is calculated for each rule.

A WFPR is defined as: R: IF a THEN c ($CF = \mu$), Th, w, where $a = \langle a_1, a_2, ...a_n \rangle$ is the antecedent portion which comprises of one or more propositions connected by either "AND" or "OR". Each proposition $a_i (1 \le i \le n)$ can have the format "x is f_{ai}", where f_{ai} is an element of a set of fuzzy sets F = $\{ f_1, f_2, ...f_n \}$. The consequent of the rule c can be expressed, as "x is f_c", where f_c is also an element of F. The parameter μ is the certainty factor of the rule R and it represents the strength of belief of the rule. The symbol $Th = \langle \lambda_1, \lambda_2, ...\lambda_n \rangle$ represents a set of threshold values specified for the proposition in the antecedent a. The set of weights assigned to the propositions $\langle a_1, a_2, ...a_n \rangle$ is given by $w = \langle w_1, w_2, ...w_n \rangle$. The weight w_i of a proposition a_i shows the degree of importance of a_i contributing to the consequent c when comparing to other proposition a_j, for $j \ne i$. It is obvious that when there is only one proposition in the antecedent of fuzzy production rules, the weight w_i is meaningless. The set of weight w assigned to each proposition in the antecedent is referred as local weights. Another important concept called global weight, which could be assigned to each rule in an inference path, is fully explored in (Yeung and Tsang, 1995).

In general WFPR's are categorized into three types, which are defined as follows:

Type 1: A Simple Fuzzy Production Rule
R: IF a THEN c ($CF = \mu$), λ, w, For this type of rule, since there is only one proposition 'a' in the antecedent, the weight w is meaningless.
Type 2: A Composite Fuzzy Conjunction Rule
R: IF a_1 AND a_2 THEN c ($CF = \mu$), $\lambda_1, \lambda_2, w_1, w_2$,

Type 3: A Composite Fuzzy Disjunction Rule
R: IF a_1 OR a_2 THEN c ($CF = \mu$), $\lambda_1, \lambda_2, w_1, w_2$,

For both types 2 and 3, λ_i is the threshold value for a_i and w_i is the weight assigned to a_i. Some authors do not assign a certainty factor to a FPR while others ignore the weight and the threshold value assigned to each proposition in the antecedent. We considered that the capturing of fuzzy knowledge using fuzzy production rule with weights and threshold values plays an important role in real world applications. Hence the weight (degree of importance), the threshold value as well as the certainty factor have been taken into account.

4.1 Weighted Fuzzy Production Rules with Single Antecedent

R: IF A THEN C, ($CF = \mu$), Th, W e.g., if a_1 then C, ($CF = \mu$), $Th = \{\lambda_{a_1}\}$, $W = \{w_1\}$. C is represented as a "concluded disorder" in Chen's Diagnosis problem (Chen, 1988; Chen, 1994) or a consequent in other problems.

Given four cases of facts:

Case 1: $A' = A$
Case 2: $A' =$ very A
Case 3: $A' =$ more or less A
Case 4: $A' =$ not A

What conclusion C' can be drawn? In order to draw the conclusion C', similarity-based fuzzy reasoning algorithm is analyzed all of these possible cases and select the most accurate case which is best suited for classification of stock market prediction.

4.2 Weighted Fuzzy Production Rules with Multiple Antecedents

If the antecedent portion or consequence portion of fuzzy production rule contains "AND" or "OR" connectors, then it is called a composite fuzzy production rule. According to (Looney, 1987), the composite fuzzy production rule can be distinguished into the following rule-types:

Type 1: IF a_{j1} AND, a_{j2} AND...AND a_{jn} THEN a_k ($CF = \mu_i$)
Type 2: IF a_j THEN a_{k1} AND ,a_{k2} AND.....AND a_{kn} ($CF = \mu_i$)
Type 3: IF a_{j1} OR, a_{j2} OR...OR a_{jn} THEN a_k ($CF = \mu_i$)
Type 4: IF a_j THEN a_{k1} OR ,a_{k2} OR.....OR a_{kn} ($CF = \mu_i$)

In proposed algorithm, multiple propositions connected by "AND" are used,
R: IF A THEN C, ($CF = \mu$), Th, W e.g., if a_1 AND a_2 THEN C, ($CF = \mu$), $Th = \{\lambda_{a1}, \lambda_{a2}\}$, $W = \{w_1, w_2\}$ $A = \langle a_1, a_2 \rangle$, a_1 AND a_2 are connected by "AND" consider the following four cases:

Case 1: $A' = \langle a_1', a_2' \rangle = A = \langle a_1, a_2 \rangle$
Case 2: $A' =$ very $A = \langle$ very a_1, very $a_2 \rangle$ A $\langle a_1', a_2' \rangle$ $\langle a_1', a_2' \rangle$ $\langle a_1', a_2' \rangle$
Case 3: $A' = \langle a_1', a_2' \rangle =$ more or less $A = \langle$ more or less a_1, more or less $a_2 \rangle$
Case 4: $A' = \langle a_1', a_2' \rangle =$ not $A = \langle$ not a_1, not $a_2 \rangle$

What conclusion can be drawn?

4.3 Mapping a WFPR and Its Reasoning Algorithm to FNN

A set of WFPRs and the proposed weighted fuzzy reasoning algorithm can exactly be mapped into a three-layer FNN. These three layers are called Term layer, Rule layer, and Classification layer. We describe the structure of the mapped FNN as follows.

Term layer: This is the input layer (layer i). Each node in this layer represents a linguistic term of an attribute. Since each linguistic term corresponds to an attribute value, the input of each node is regarded as the similarity degree between the observed attribute value and the corresponding term (proposition) of the antecedent in a WFPR. The similarity degree can also be the membership value that indicates to what degree the observed fact belongs to the linguistic term.

Rule layer: This is the only hidden layer (layer j). Each node in this layer represents a given antecedent part of a rule. According to linguistic terms (propositions) appeared in the antecedent part of a rule, the connections between the term layer and the rule layer are determined.

Classification layer: This is the output layer (layer k). Each node in this layer represents a fuzzy cluster. Since the inferred result of a WFPR has generally the form of vector (discrete fuzzy set defined on the space of cluster labels), the output of the network has more than one value. The meaning of each output value after normalization is the membership value that indicates to what degree the training object belongs to the cluster corresponding to the node.

Connection weights: The local weights (shown as L_{wij}) of a set of WFPRs are regarded as the connection weights between the term layer and the rule layer. The global weights (shown as G_{wjk}) of the set of WFPRs are regarded as the connection weights between the rule layer and the classification layer.
Fig. 1 presents a generic conjunctive WFPRs mapped to a FNN which could be used to refine and tune local and global weights.

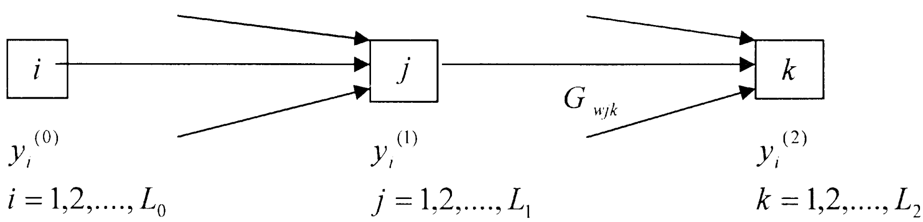

Figure 1: Generic FNN for a conjunctive WFPRs

5. Back-propagation Algorithm and the Convergence of the Fuzz Learning Rule

A. The Convergence of the Fuzzy Learning Rule

Let us consider a two-layer feed forward FNN as shown in Fig. 2 in which the neuron unit is a fuzzy neuron with fuzzy operators $(\wedge, \bullet)$. The training method is presented as follows.

Step 1) Setting the initial connection weights

Setting $W_{ij} = 1, \quad i = 1,2,\ldots\ldots,n; \quad j = 1,2,\ldots\ldots,m$

Step 2) Calculating the actual output

$$(b_i^{\,j})' = \mathop{\wedge}_{k=1}^{n} (W_{kj} . a_i^{\,k}) \qquad j = 1,2,\ldots\ldots,m \qquad (10)$$

where $A_i = (a_i^1, a_i^2, \ldots\ldots\ldots, a_i^n)$ $i = 1,2,\ldots\ldots\ldots,n$ is the vector of pattern's inputs
$B_i = (b_i^1, b_i^2, \ldots\ldots\ldots, b_i^m)$ .is the vector of pattern's outputs.
$(b_i^1)', (b_i^2)', \ldots\ldots\ldots, (b_i^m)'$ is the actual response for the input pattern A_i. W_{ij} stands for the connection weight from node i in F_1 to node j in F_2.

Step 3) Adjusting the connection weight

Let $\delta_{ij} = (b_i^j)' - b_i^j$

$$W_{kj}^{new} = \begin{cases} W_{kj}^{old} - \eta.\delta_{ij} & \text{. } if \quad W_{kj}^{old}.a_i^k < b_i^j \\ W_{kj}^{old} & otherwise \end{cases} \tag{11}$$

where $\eta \in (0,1]$ denotes the learning rate.

Step 4) Go to Step 3 until $W_{kj}^{new} = W_{kj}^{old}$ hold for all k and j

Step 5) Repeat Step 2 for the new input and output pattern.
This algorithm is called the fuzzy learning rule. To the fuzzy learning rule, we have the following theorems.

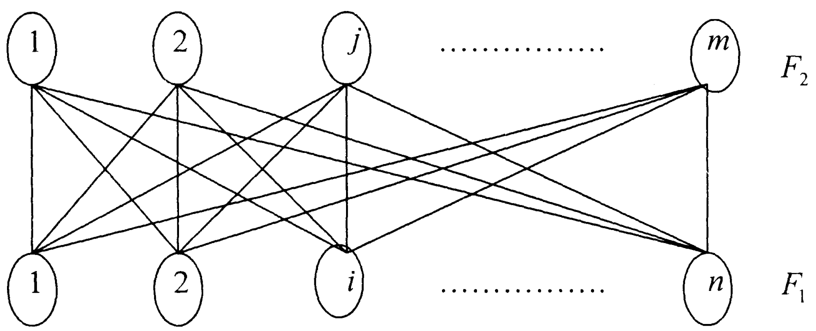

Figure 2: Operator network

Theorem 1: The fuzzy learning rule is convergent.
Proof: This theorem is an extension of δ-rule found in (Minsky and Papert, 1988). From the steps 1 to 5, it is easy to see that this learning rule converges.

Theorem 2: If a solution to the following equation-group exists:

$$\overset{n}{\underset{k=1}{\wedge}} (W_{kj}.a_i^k) = b_i^j \qquad i = 1,2,\ldots\ldots,n; \; j = 1,2,\ldots\ldots\ldots,m \tag{12}$$

then the fuzzy learning rule algorithm can converge to the W° (W° is an $m \times n$ matrix) such that W° satisfy the above equation-group.
Proof: From the theory of fuzzy relation equation (He, 1985) one may notice that each iteration when this neural network learns, it searches for a matrix of weights so that $\overset{n}{\underset{k=1}{\wedge}} (W_{kj}.a_i^k) = b_i^j$ $i = 1,2,\ldots\ldots,n; \; j = 1,2,\ldots\ldots\ldots,m$, i.e., it tries to find a solution for the fuzzy relation equation-group. If W° (an $m \times n$ matrix) exists and is the solution of this fuzzy relation equation group, then it is true that the fuzzy learning rule converges to this W°. This completes the proof.

Theorem 2 shows that a two-layer FNN with $(\wedge, \bullet)$ operators can produce the fuzzy relation: $A_i = (a_i^1, a_i^2, \ldots\ldots\ldots, a_i^n) \rightarrow B_i = (b_i^1, b_i^2, \ldots\ldots\ldots, b_i^m)$ $i = 1,2,\ldots\ldots\ldots,n$ by learning.

B. Generic Example of a FNN

To formulate the back-propagation algorithm, let us consider a generic case of our proposed FNN as shown in Fig. 1, where there are L_0 Term nodes, L_1 Rule nodes and L_2 Classification nodes. For a given input vector, e.g. the nth input vector, the feed forward propagation process is described as follows:
The initial layer (Term layer): $\{y_i^{(0)}[n] \mid i = 1,2,......., L_0\}$ (the given input vector);
The first layer (Rule layer):

$$y_j^{(1)}[n] = \wedge_{i=1}^{L_0} \times (L_{wij} \cdot y_i^{(0)}[n]) \quad j = 1,2,........, L_1 \tag{13}$$

The second layer (Class layer):

$$y_k^{(2)}[n] = \sum_{j=1}^{L1} G_{wjk} \cdot y_j^{(1)}[n], \; k = 1,2,...., L2. \tag{14}$$

Let there be N training sample data. Then, the total error function is usually defined as

$$E = \frac{1}{2}\sum_{n=1}^{N}\sum_{k=1}^{L_2}(d_k[n] - y_k[n])^2$$

$$E = \sum_{n=1}^{N}\left(\frac{1}{2}\sum_{k=1}^{L_2}(d_k[n] - y_k[n])^2\right) = \sum_{n=1}^{N} E_n \tag{15}$$

Where $d_k[n] = y_k^{(2)}[n] / \max_{1\le k\le L_2}\{y_k^{(2)}[n]\}$ is a normalization value of the $k-th$ actual output of the $n-th$ training sample $(1 \le k \le L_2)$. It is easy to see from (1), (2), and (3) that the error E is a function with respect to the local weight L_{wij} and the global weight G_{wjk} $(i = 1,....., L_0)$; $j = 1,.....L_1$; $k = 1,......, L_2$. The main objective of learning is to adjust these weights so that the error function reaches minimum or is less than a given small value ε.

C. Enhanced Back-Propagation Algorithm for the FNN

A back-propagation, one of the most popular and powerful learning algorithms, has been proposed for years to learn a multilayer neural network with three or more layers. In our proposed FNN, we establish an enhanced back-propagation algorithm by modifying the smooth derivative introduced in (Blanco, 1995) which is briefly described as follows.

The usual derivatives

$$\partial(y \wedge p)/\partial y = \begin{cases} 1 & if \quad y \le p \\ 0 & if \quad y < p \end{cases} \quad \text{and}$$

$$\partial(y \vee p)/\partial y = \begin{cases} 1 & if \quad y \ge p \\ 0 & if \quad y < p \end{cases} \tag{17}$$

are regarded as the crisp truth degree of the proposition " y is less than or equal to p " and the crisp truth degree of the proposition " y is greater than or equal to p " respectively. To improve the performance of training, these crisp behaviors will be replaced by fuzzy behaviors which are able to capture the real meaning of $(y \leq p)$ and $(y \geq p)$ in a vague context (Blanco, 1995). Since the relative position of y with respect to p is softened, the relative position could be regarded as the minority degree of p with respect to y, denoted by $\| p \leq y \|$. Noting that when $p \leq y$ then $\| p \leq y \| = 1$, whereas when it is reasonable to consider the minority degree of $\| p \leq y \|$ to be equal to y. The Godel implication is the most suitable one. Consequently, two enhanced derivatives are defined as follows:

$$\frac{\partial(x \vee c)}{\partial x} = \begin{cases} 1 & if \quad x \geq c \\ x & if \quad x < c \end{cases} \text{ and } \frac{\partial(x \wedge c)}{\partial x} = \begin{cases} 1 & if \quad x \leq c \\ c & if \quad x > c \end{cases} \tag{18}$$

Let us now derive the standard back-propagation equations.
According to the principle of gradient descent, the back-propagation equations for the FNN as shown in Fig. 1 can be written as

$$L_{wij} = L_{wij} - \alpha \frac{\partial E_n}{\partial L_{wij}} \text{ and } G_{wjk} = G_{wjk} - \beta \frac{\partial E_n}{\partial G_{wjk}} \tag{19}$$

where α and β are the learning rate. Therefore, the problem of derivation is how to evaluate the two partial derivatives appeared in (Jang, 1993).
The detailed derivation can be found in (Eric et al., 2004) and the derived results are shown in the form of equation where the attached $[n]$ has been omitted from each

$y_{\alpha}^{\beta}(\alpha = i, j, k; \beta = 0,1,2)$, and all notations have the same meaning as that in (13) and (14).

Theorem 3: The enhanced Back-Propagation Algorithm for FNN converges.
Proof: From Theorem 1 and the traditional gradient descent learning method of a neural network, it is easy to see that our enhanced Back-Propagation algorithm for FNN converges.

6. Discussion

Fuzzy Neural Network method could be used to solve parameters refinement, a tuning problem, or a parameters acquisition problem. What we need to do is map the FPRs to a FNN and randomly assign the initial weights to real values in [0, 1]. The training data are then fed into the network. After training, the tuned or adjusted weights are obtained, which are the required parameters. Thus, it could be used to solve some of the knowledge acquisition problems. In our experiment we put some constraints on the weight updating. It is meaningless for the knowledge representation parameters: CFs and LWs to exceed a specified range. There is a criterion of keeping the knowledge representation parameters in their allowable range [0, 1]. If the weights are out of range, a weight adjustment method is used. This method has also been discussed in (Lacher, et al., 1992).

In some experiment of FNN by Tsang, E.C.C et al., (2002) used to model a job-seeking expert system, the error tolerance is set to 0.01 and the maximum cycle is set to 500 to avoid an endless loop in the learning process. As we know, that NN has a problem of easily getting stuck to a local minimum. In order to reduce the chance of getting stuck to a local minimum, we could use a momentum term to guide the FNN.

There are many different FNNs proposed and found in literature. A survey paper which summarizes the fusion and union methods of fuzzy with (NN)/(GAs) could be found in (Hiraga et al., 1998). In order to evaluate the pros and cons of our proposed method, we tried to compare our proposed method with other existing methods. To our surprise, there are some papers in the literature that use FNN to tune parameters in multilevel FPRs. Many papers in FNN use NN to tune parameters of membership functions in single-level FPR systems. So it is not appropriate trying to compare our proposed model with those methods mentioned in (Hiraga et al., 1995) and

Jang, J. R. (1993) as they only handle single-level FPRs while our method handles more complex multilevel FPRs.

Tsang, E.C.C et al., (2002) use the numerical data representing the local weights and certainty factors are used for experiments. One may notice that in FPRs Tsang, E.C.C et al., (2002) the conclusion drawn is the same as the original consequent (i.e., $B' = B$) with certainty factor (numerical value) indicating the degree of truth of this conclusion. The method proposed in this paper is still the "computing with words (CW)" concept as proposed in (Yeung, D. S. and Tsang, E. C. C. 1996). The differences between our method with the method proposed in (Yeung, D. S. and Tsang, E. C. C. 1996) is that: 1) we use the degree of similarity method while in (Yeung, D. S. and Tsang, E. C. C. 1996) the CRI method is used instead and 2) our conclusion is the same as the consequent of FPRs and a degree of truth is computed, while in CRI the conclusion drawn may be different from the consequent of FPRs and no degree of truth is provided.

Our proposed method could be used to solve multi-attribute decision making (MADM) problems if decision makers (DMs) are able to represent decision processes in terms of FPRs. As mentioned in (Hashiyama et al., 1993), MADM refers to the problem of selecting among alternatives associated with multiple attributes. The multiple attributes could be represented as propositions in the antecedent and the alternative represented as the proposition in the consequent in our conjunctive FPRs. In (Hashiyama et al., 1993), a FNN is used to identify the changes in the subjects' weights of the attributes.

7. Conclusions

This paper proposes a method to generate and obtain a set of approximately optimal WFPRs by refining and tuning the local and global weights with a FNN. The aim of including local and global weights in FPRs and refinement of these weights is to improve the learning and testing accuracy without increasing the number of rules in the learning problem. When a local weight is found to have small or zero value after refinement, the corresponding proposition in the antecedent of a rule could be deleted. Thus a set of approximately optimal WFPRs could be extracted. We know that the simpler the form of the extracted rules, the stronger the generalization capability of the extracted rules. As the computational complexity of finding an optimal set of fuzzy rules is generally NP-hard, the approach to find approximately optimal set of fuzzy rules becomes very important. Hence, the set of extracted fuzzy rules, with high learning and testing accuracy and with small number of the rules, should be considered to be optimal.

Our future research work on rule refinement will be on determining the trade-off and strike a balance between the number of rules extracted and the testing accuracy of the extracted rules by using large databases.We will look into the problems of how we could achieve an optimal number of rules by deleting those rules with small or zero global weights. We will also develop an algorithm that will allow us to tune, refine and find optimal rules from a set of rough, crude and raw rules. The robustness and statistical property of this algorithm will also be studied.

Reference

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), 2000.

Alan, Fan., Palaniswami, M. (2001). Stock selection using support vector machines *Proceedings. IJCNN '01. International Joint Conference on Neural Networks, 2001,* Volume: 3 , 15-19 July 2001 Pages:1793 - 1798 vol.3

Blanco, A., Delgado, M. , and Requena, I. (1995). Identification of fuzzy relational equations by fuzzy neural networks," *Fuzzy Sets Syst.*, vol. 71, pp. 215–226, 1995.

Chih-Chung, Chang., and Chih-Jen, Lin. (2001). LIBSVM: a Library for Support Vector Machines (Version 2.31), 2001.

Dong, M. and Kothari, R. (2001). Look-ahead based fuzzy decision tree induction. *Journal for IEEE-FS*, volume = 9, June 2001 pages = 461-468 url = citeseer.ist.psu.edu/558808.html}.

Han, J., Pei, J., and Yin, Y. (2000). Mining frequent patterns without candidate generation. *Sigmod' 2000*, paper ID: 196.

He, Z. Y. (1985) *Fuzzy Mathematics and Its Applications*. Tianjing, China: Tianjing Science Technology, 1985, pp. 188–215. artificial intelligence.

Han, J., Pei, J., Yin, Y. (2000). Mining frequent patterns without candidate generation. *Sigmod' 2000*, paper ID: 196.

Haiqin, Yang., King, I., and Laiwan, Chan. (2002). Non-fixed and asymmetrical margin approach to stock market prediction using Support Vector Regression. ICONIP '02. Proceedings of the 9th International Conference on Neural Information Processing, 18-22 Nov 2002, Volume: 3 , . 2002 Pages:1398 – 1402.

Hong, T. P. and Chen, J. B. (1999). Finding relevant attributes and membership functions. *Fuzzy Sets Syst.*, vol. 103, no. 3, pp. 289–404, 1999.

Hashiyama, T., Furuhashi, T., and Uchikawa, Y. (1993). A fuzzy neural network for identifying changes of degree of attention in a multi-attribute decision making process," in *Proc. Int. Joint Conf. Neural Networks*, 1993, pp. 705–708.

Hiraga, I., Uchikawa, Y. and Nakayama, S. (1995). An acquisition of operator's rules for collision avoidance using fuzzy neural networks," *IEEE Trans.Fuzzy Syst.*, vol. 3, pp. 280–287, Aug. 1995.

Hiraga, I. , T. Furuhashi, and L. C. Jain. (1998). Acquisition of fuzzy knowledge by NN and GA—a survey of the fusion and union methods proposed in Japan. in *Proc. IEEE 2nd Int. Conf. on Knowledge-Based Intelligence Electronic Systems (KES'98)*, 1998, pp. 69–78.

Jang, J. R. (1993) "ANFIS: Adpative-Neural-Based Fuzzy Inference System," *IEEE Trans. Syst., Man, Cybern.*, vol. 23, pp. 665–685, May/June 1993. John W. T. Lee (M'88) received the B.Sc. degree

Khokhar, R. H. and Noor, M. M. S. (2004) " Classification of Degree of Importance of Attributes for Stock Market Data Mining" Journal of Information Technology, Jilid 16, Bil.2 21-43.

Lacher, R. C., S. I. Hruska, and C. Kuncicky. (1992). Back-propagation learning in expert networks. *IEEE Trans. Neural Networks*, vol. 3, pp. 62–72, Jan. 1992.

Looney, C. G. (1987). Fuzzy controls via Boolean rule matrix transformations," *IEEE Trans, Syst., Man, Cyber.*, vol. SMC-17, no. 6 , pp. 1077-1082, Nov/Dec, 1987.

Minsky, M. L. and Papert, S. A. (1988). *Perceptrons*. Cambridge, MA: MIT Press, 1988, pp. 161–170.

Noor, M. M. S, A. M. Awan. (2004) Weighted Kernel K-Means Algorithm for Clustering Spatial Data. *Journal of Information Technology, University Technology Malaysia,* Vol. 16 (2), No 10, pp 137-156, Dec 2004

Pawlak, Z. (1991) *Rough Sets, Theoretical Aspects of Reasoning about data*. Dordrecht, : Kluwer, 1991.

Quinlan., J. R. (1986). Induction of Decision Tree. *Machine Learning*, Vol, pp.81-106 1986.

Quinlan., J. R. (1993). C4.5: Programs For Machine Learning. Morgan Kaufmann, Los Altos, 1993.

Raymond, S. T. Lee. (2004), iJADE Stock Advisor: An Intelligent Agent based Stock Prediction System using Hybrid RBF Recurrent Network, *IEEE Transactions on Systems, Man and Cybernetics (Part A)* 34(3) 421-428.

Torben, G. A. and Lund, J. (1997). Estimating continuous-time stochastic volatility models of the short-term interest rate. *Journal of Econometrics* 77 (1997), pp. 343--378.

Tsang, E.C.C., Lee, J.W.T., Yeung, D.S. (2002) Tuning certainty factor and local weight of fuzzy production rules by using fuzzy neural network Systems, Man and Cybernetics, Part B, IEEE Transactions on Volume 32, Issue 1, Feb. 2002 Page(s):91 - 98

Umanol, M., Okamoto, H., Hatono, I., Tamura, H., Kawachi, F., Umedzu, S., Kinoshita, J. (1994). Fuzzy decision trees by fuzzy ID3 algorithm and its application to diagnosis systems Fuzzy Systems, 1994. IEEE World Congress on Computational Intelligence., Proceedings of the Third IEEE Conference on , 26-29 June 1994 Pages:2113 - 2118 vol.3

Wang, X.-Z., Yeung, D.S., and Tsang, E.C.C. (2001). A comparative study on heuristic algorithms for generating fuzzy decision trees Systems. *IEEE Transactions on Man and Cybernetics, Part B,* Volume: 31 , Issue: 2 , April 2001 Pages:215 – 226.

Weiss, E. (2000). Forecasting commodity prices using ARIMA. *Technical Analysis of Stocks & Commodities*, vol. 18, no. 1, pp. 18-19, 2000.

Xiaohua, Wang., Paul, Kang., Hoh, Phua., and Weidong, Lin. (2003). Stock market prediction using neural networks: does trading volume help in short-term prediction. *Proceedings of the International Joint Conference on Neural Networks*, Volume: 4, July 20 - 24, 2003 Pages: 2438 – 2442

Yeung, D. S. and Tsang, E. C. C. (1996). Fuzzy logic = computing with words," *IEEE Trans. Fuzzy Syst.*, vol. 4, pp. 103–111, May 1996.

Yeung, D. S. and Tsang, E. C. C. (1997). A comparative study on similarity-based fuzzy reasoning methods. Systems Man and Cybernetics, Part B, IEEE Transactions on , Volume: 27 , Issue: 2, April 1997 Pages: 216 – 227

Yeung, D. S., X. Z. Wang, E. C. C. Tsang, (1999). Learning weighted fuzzy rules from examples with mixed attributes by fuzzy decision trees Systems, *IEEE International Conference on Man, and Cybernetics, SMC '99* Volume: 3 , 12-15 Oct. 1999 Pages:349 – 354

Yeung, D. S., Wang, X. Z., Tsang, E. C. C. (2002). Fuzzy rule mining by fuzzy decision tree induction based on fuzzy feature subset. IEEE International Conference on Systems, Man and Cybernetics, 2002. Volume: 4 , 6-9 Oct. 2002 Pages:6 pp. vol.4

Yuan, Y. and Shaw, M. J. (1995). Induction of fuzzy decision trees," Fuzzy Sets and Systems vol. 69, 1995, pp. 125-139

Zadeh's, L. A. (1999). Fuzzy Sets as a basis for a theory of possibilistic. *Fuzzy Sets Syst.* vol. 100, pp. 9-34, 1999.