From Knowledge to Linguistic Expression – A Novel Stock Market Expert System using Extended Super-Function

Qiong Liu, Fuji Ren, Xin Lu and Shingo Kuroiwa
Faculty of Engineering, The University of Tokushima, Tokushima, Japan 770-8506
liuqionghit@hotmail.com

Abstract

Recently, neural networks have been very successfully applied in time series to improve multivariate prediction ability. Several neural network models have already been developed for the market prediction of future price changes. But it is very difficultly to understand for the normal investor. They just want to understand the analytic results by some expressed language. So this paper proposed a new natural language generating system for TOPIX including a technical index processing unit and a natural language generation unit. The technical index processing unit is applied to calculate the technical index of stock market and produce the corresponding technical pattern. And the natural language generation unit is utilized for describing the technical analysis in natural language. So far, this paper proposed the Extended Super-function method that contributed to improving technical analysis accuracy, completeness and flexibility in natural language. The ESF method is also an attempt in artificial intelligence, which help the normal investor to understand complicated forecasting model of stock market by using comprehensible natural language expression. The results of nature language generation are easy for non-expert users to understand.

1 Introduction

A time series is a set of numbers that measures the status of some activity over time. Time-series forecasting is an important research area in several domains. Traditionally, forecasting research and practice has been dominated by statistical methods. In the last few years, neural network and other advanced methods on prediction have been used in financial domains [1]. The research has focused on understanding the nature of financial markets before applying methods of forecasting in domains including stock markets, financial indices, bonds, currencies and varying types of investments.

Neural networks have been very successful in a number of signal processing applications. Neural networks have been applied in time series to improve multivariate prediction ability [2, 3]. Neural networks map an input layer (present and past values of the time series) onto an output layer (future value). Future values of variables often depend on their past values, the past va1ues of other correlated variables, and other un- certain factors. For example, the future price of a stock depends on political and international events (event-knowledge) as well as various economic indictors. Most forecasts rely on historical trends or past experiences to predict the future [4]. Much success above is based on such characteristics as parallel processing, nonlinear processing, and non-parametric and distributed representation. This characteristics permits neural networks can flexibly and accurately approximate almost any function and makes neural networks an ideal modelling tool far studies in which there exists very little a priori knowledge about the appropriate functional representation of the relationship under investigation [5]. Interest in neural networks has led to a considerable surge in research activities in the past decade. However, most studies on time series analysis using neural networks address the use of neural networks as a forecasting tool, but limited research focuses on the application of neural networks for event extraction to acquire the dynamic relation between endogenous and exogenous variables. The measurement of past trends may be interrupted by changes in policy or data availability. When the past cannot be used to predict the future due to major unexpected shifts in economic, political, or social conditions, the future can be more uncertain due to the occurrence of major events. Evaluating the impact of these events (exogenous variables) on stock markets are interesting to economists, because the markets are affected by less measurable but important farces in economic analysis and financial prediction.

In stock market prediction, many methods for technical analysis have been developed and are being used. In technical analysis, technical indexes calculated from price sequence are used to predict the trend of future price changes. Many statistical methods have been proposed, but the results are insufficient in prediction accuracy. So in this paper, a neural network model was proposed for technical analysis of stock market, and its training method for improving the prediction capability. This paper utilizes a multilayer feedforward neural network as a prediction model to forecast the trend of stock market. The prediction of stock market trend can refer the investor to buy or sell the stocks before his investing action. The system involving this neural network model is specially applied in TOPIX (Tokyo Stock Exchange Prices Index). TOPIX is a weighted average of prices of all stocks listed on the First Section of Tokyo Stock Exchange.

This paper also describes a natural language generation system to express prediction information of TOPIX in natural language for non-expert users. By gradually integrating complementary aspects of various linguistic theories within the computational framework of functional unification, this system has evolved to be one of the most comprehensive grammars of English for prediction expressions using ESF we proposed.

2 Stock Market Forecasting System

Figure 1 is the overview of the proposed forecasting system for TOPIX. The prediction system classifies the input pattern that consists of several technical data of TOPIX, and generates a buying, selling or holding signal for notifying users. As shown in the Figure 1, the system consists of a neural network, a preprocessing unit, a postprocessing unit and a natural language generation unit. The preprocessing unit standardizes each technical data to form an input pattern into the neural network. Then the network recognizes the turning point of the TOPIX price curve from the input pattern. Whereafter, the postprocessing unit converts the result of recognition into a buying, selling and holding signals in boolean value (0 or 1). Finally, depending on the prediction signals and the prepared special lexicon relative to stock market, the natural language generation unit creates the prediction expression in English.

2.1 Selecting Technical Data

An index is a series of data points that are derived by applying a formula to the price data of a security. Price data includes any combination of the open, high, low or close over a period of time. Some indexes may use only the closing prices, while others incorporate volume and open interest into their formulas. The price data is entered into the formula and a data point is produced. Data items to form the input pattern to this system are technical indexes of TOPIX. Typical technical indexes are:
1. **Moving Average**;
2. **Moving Average Convergence Divergence**;
3. **Turnover Moving Average**;
4. **Relative Strength Index**;
5. **Bollinger Bands**;
6. **Parabolic SAR**;
7. **Stochastic Oscillator**.

2.2 Constructing Network Model

As shown in Figure 1, the neural network as prediction model is a hierarchical network that consists of three layers: the input layer, the hidden layer, and the output layer. Each unit in the network is connected to all units in the adjacent layers. Each unit receives outputs of the units in the lower layer and calculates the weighted sum to determine total input. Then the output is determined by applying the logistic function [6] to the total input. As a result, the output ranges in 0 to 1.

2.3 Defining Prediction Formality

In the neural network model, the output layer has three units. As output patterns of the network, we define three patterns as shown in Figure 2. Each corresponds to specific TOPIX curve patterns: buying signal (i.e. current price is at bottom), selling signal (i.e. current price is at top), and no-change (i.e. otherwise), respectively. Bottoms and tops in the price curve are closely related to buying and selling timings. When teaching the neural network model, each correct output pattern of training sample is calculated from three TOPIX data as described in Figure 2: current price, price at five weeks before, and price at five weeks later.

TOPIX curve			neural network correct output pattern		
past	current	future	Selling Prediction	Buying Prediction	Holding Prediction
↗		↘	1	0	0
↘		↗	0	1	0
↗		↗	0	0	1
↘		↘			

Figure 2: Relation between TOPIX curve and correct output pattern of neural network

On the TOPIX graph the correct signals are calculated from three TOPIX data (i.e. current price, price at five days before, and price at five days later) as described in Figure 2, and are not recognized by human experts. However, an expert analyst comments that these correct signals are almost satisfactory as long as the investment period is supposed to be of three months.

Because the output of the units in the output layer ranges in analog of 0 to 1 in a neural network, an actual output pattern may not match with any of the three patterns. In this case, the postprocessing unit converts the analog value to 0 or 1 by using two thresholds. In the experimental simulation further described , 0.2 and 0.8 are used as thresholds. When the output is beneath 0.2, then 0 is selected. In the same way, when the output is more than 0.8, then 1 is selected. If the output is between 0.2 and 0.8, or if the converted pattern still does not match any of the three categories, the system notifies that prediction has failed.

2.4 Generating Natural Language Expression

A ESF is a function that shows the correspondence between original symbol pattern and target symbol patterns. The conception of symbol pattern is most necessary to ESF and will be described firstly. Pattern structures have four basic ingredients, not commonly found elsewhere:

1. Every significant symbol set has attributes and

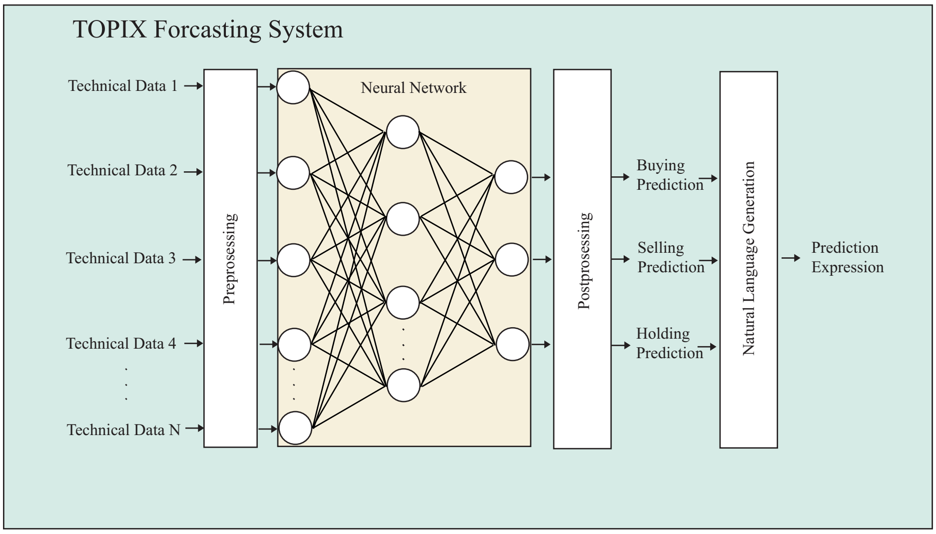

Figure 1: TOPIX Forecasting System

corresponding values.
2. The pairing of a token, attributes and values can form a symbol pattern.
3. The function can be regarded as the relationship of patterns, in which the patterns act as their roles.
4. The new value of attribute can be generated by the function influenced by patterns.

These ingredients form the essence of pattern structures. Therefore, the core pattern structure can be defined as following.

1. The vocabulary of core language consists of:
(a) A finite set of attribute symbols;
(b) A finite set of atomic value symbols;
(c) A finite set of token symbols;
(d) A infinite set of function structures;
(e) The auxiliary symbols [] () <>.

2. The structure of core language consists of:
(a) If t is an token, A is an attribute constant and v is a atomic value constant, then $p_t ::= p\ [t,\ A_:v]$ is a atomic pattern expression.
(b) If $p_1, ..., p_k$ are pattern expressions, then $f(p_1, ..., p_k)$ is function structure and also regarded as complex value.
(c) If t is an token, A is an attribute constant and f is a complex value, then $p_t ::= p\ [t,\ A : f]$ is a complex pattern expression.

The pattern structure is not the linear notation where the patterns are ordered in two dimensions or more. The core of semantic definition is atomic pattern specification. Any complex pattern expression is composed of some atomic pattern expressions ordered in function structures.

The grammar of Extended Super-function is a five-tuple

$$< V_S,\ V_T,\ F_S,\ F_T,\ R > \tag{1}$$

where
$-$ V_S is a finite set of patterns of source symbol set;
$-$ V_T is a finite set of patterns of target symbol set;
$-$ F_S is a infinite set of functions of source symbol set;
$-$ F_T is a infinite set of functions of target symbol set;
$-$ R is a infinite set of functions which means the relationship of patterns between source symbol set and target symbol set. if the relationship is assumed as equivalence, the set of functions means the translation between two symbol sets.

The following examples illustrate the Extended Super-function grammar with this semantics.

Example: This is a ESF from the Relative Strength Index (RSI) expression in stock technical analysis to its specification in natural language (English). (RSI is a momentum oscillator that compares the magnitude of gains against the magnitude of losses.)

$$ESF : \begin{cases} SELL_i = 1,\ BUY_i = 0, \\ if\ RSI_{i-1} \geq 70,\ RSI_i < 70 \\ SELL_i = 0,\ BUY_i = 1, \\ if\ RSI_{i-1} \leq 30,\ RSI_i > 30 \end{cases}$$

$$\rightleftharpoons \begin{cases} The\ RSI\ falls\ below\ 70,\ the\ stock\ is \\ overbought,\ and\ it\ is\ considered\ bearish\ signal. \\ The\ RSI\ rises\ above\ 30,\ the\ stock\ is \\ oversold,\ and\ it\ is\ considered\ bullish\ signal. \end{cases}$$

$When$
$\mathbf{RSI} = \{RSI_i \mid 0 \leq RSI_i \leq 100,\ i = 1, 2, 3, ..., n\}$
$\mathbf{SELL} = \{SELL_i \mid SELL_i \in \{0, 1\},\ i = 1, 2, 3..., n\}$
$\mathbf{BUY} = \{BUY_i \mid BUY_i \in \{0, 1\},\ i = 1, 2, 3..., n\}$

Definition:$ESF =< V_S,\ V_T,\ F_S,\ F_T,\ R >$

$V_S = \{p\ [t_1,\ Variable : RSI_{t-1}],$
$p\ [t_2,\ Variable : RSI_t],$
$p\ [t_3,\ Expression : f\ (p_1^S,\ p_2^S)],$
$p\ [t_4,\ Expression : g\ (p_1^S,\ p_2^S)],$
$p\ [t_{root},\ Expression : f\ (p_3^S,\ p_4^S)]\}$

$V_T = \{p\ [t_1,\ String : The\ RSI\ falls\ below\ 70],$
$p\ [t_2,\ String : The\ RSI\ rises\ above\ 30],$
$p\ [t_3,\ String : f\ (p_1^T)],\ p\ [t_4,\ String : f\ (p_2^T)],$
$p\ [t_{root},\ String : f\ (p_3^T,\ p_4^T)]\}$

$F_S = \{f\ (p_1^S,\ p_2^S) = \ (p_1^S \geq 70,\ p_2^S < 70),$
$g\ (p_1^S,\ p_2^S) = \ (p_1^S \leq 30,\ p_2^S > 30),$
$f\ (p_3^S,\ p_4^S) = Or\ (p_3^S,\ p_4^S)\}$

$F_T = \{f\ (p_1^T) = (p_1^T,\ the\ stock\ is\ overbought,$
$and\ it\ is\ considered\ bearish\ signal),$
$f\ (p_2^T) = (p_2^T,\ the\ stock\ is\ oversold,$
$and\ it\ is\ considered\ bullish\ signal),$
$f\ (p_3^T,\ p_4^T) = Or\ (p_3^T,\ p_4^T)\}$

$R = \{\lambda_3 < p_3^S,\ p_3^T >,\ \lambda_4 < p_4^S,\ p_4^T >,$
$\lambda_{root} < p_{root}^S,\ p_{root}^T >\}$

In order to make the generated sentences more comprehensible, we need to modify the lexical chooser and syntactic generator of ESF to increase intelligibility of the output. Our current work involves modifying the ESF language generation package so that it can produce generic language sentence for the stock market forecasting.

3 Experimental Simulation

In order to verify the accuracy of the prediction system, we have carried out an experimental simulation applied to actual TOPIX data. 300 data from August 19, 2002 to November 6, 2003 have been used as training samples for making prediction models. Following 30 data from November 7, 2003 to December 19, 2003 are used as prediction samples for evaluation. For sample, correct output patterns are calculated according to the definition described in Figure 2. Seven technical indexes of TOPIX described in Section 2 are selected to form input patterns into the system as listed.

The neural network model with normal training has learned the training samples in no-change category completely. The Sum-Squared Error and the Average Error is 18.7632 and 0.2672 separately. They denote that the accuracy of neural network model is sufficient. The performance of the statistical model is much lower than that of the normal neural network model in every aspect. To verify the usefulness of the proposed prediction system, buying and selling were simulated. For comparison of performance reason, two other cases of simulation have also been carried out using a single technical index: relative strength index (RSI). This index has been used everyday in stock market analysis. One more case of buy-and-hold strategy was also taken as a basis. In buy-and-hold, buying is done once at the beginning of the simulation and selling is done at the end of it, that means no active buying and selling. Prediction systems are required to achieve higher performance than this strategy's. From the comparison of the three cases of simulation, the results show that the proposed neural network model scored highest performance.Then the results are translated into English using ESF.

4 Conclusions

This paper proposed a new forecasting system for TOPIX including a neural network model unit and a natural language generation unit. The neural network model is applied to the technical analysis of stock market prediction, and the natural language generation unit is utilized for describing the technical analysis in English. So far, This paper proposed a training method that contributed to improving prediction accuracy of the stock market. The proposed method is also an attempt in Artificial Intelligence, which help non-expert users to understand complicated forecasting model of stock market by using comprehensible natural language expression.

The experimental simulation applied to practical data has demonstrated that the prediction system firstly generates buying and selling signals at more proper timings on the whole, which made higher profit compared with that yielded by a single use of each technical index. And the prediction system use the buying and selling signals to generate nature language expression in English. The results of nature language generation are easy for non-expert users to understand. As future work, we need to study several problems which will help improve the model. First, on analyzing the network, it is necessary to know the effectiveness of each technical index used as input. Second, In order to make the generated sentences more comprehensible, we need to modify the lexical chooser and syntactic generator of ESF to produce more complex constitutions to increase intelligibility of the prediction output.

References

[1] AN. Refenes, N. Burgess, and Y. Bentz, "Neural Networks in Financial Engineering: A study in methodology," in *IEEE Trans on Neural Networks*, pp. 1222-1267, 8(6), 1997.

[2] Apostolos-Paul N. Refhes, A. Neil Burgess and Yves Bentz, "Neural Networks in Financial Engineering: A Study in Methodology," in *IEEE Trans on Neural Networks*, pp. 1222-1267, 8-6, 1997.

[3] A. F. Atiya, S. M. E1-Shoura, S. I. Shaheen, M. S. El-Sherif, "A comparison between neura1-network forecasting techniques case study: river flow forecasting," in *IEEE Trans on Neural Networks* pp. 402-409, 10-2, 1999.

[4] K. Kohara, "Stock Price Prediction Using Prior Knowledge and Neural Network," in *Intelligent Systems in Accounting and Management*, pp. 11-22, 6, 1997.

[5] Min Qi, "Predicting US recessions with leading indicators via neural network models," in *International Journal of Forecasting*, pp. 383-401, 17, 2001.

[6] D. E. Rumelhart, , G.E. Hinton, and R.J. Williams, "Learning Internal Representation by Error Propagation" in *D.E. Rumelhart et al [Eds.] Parallel Distributed Processing*, MIT PRESS, pp. 1/318-1/362, 1986.