

NORN Finance Forecaster - A Neural Oscillatory-based Recurrent Network for Finance Prediction

Raymond S. T. Lee and James N. K. Liu

Department of Computing, Hong Kong Polytechnic University

Hung Hom, Hong Kong

csstlee@comp.polyu.edu.hk, csnkliu@comp.polyu.edu

Abstract

Financial prediction is so far the most important applications in contemporary scientific study. In this paper, we present a fully integrated stock prediction system – NORN Finance Forecaster – A Neural Oscillatory-based Recurrent Network for finance prediction system to provide both a) Long-term trend prediction, and b) Short-term stock price prediction. One of the major characteristics of the proposed system is the automation of the conventional financial technical analysis technique such as market pattern analysis via NOEGM (Neural Oscillatory-based Elastic Graph Matching) model and its integration with the Time-difference recurrent neural network model. This will provide a fully integrated and automated tool for analytic and investigation of stock investment. From the implementation point of view, the stock pricing information of 33 major Hong Kong stocks in the period of 1990 to 1999 are being adopted for system training and evaluation. As compared with contemporary neural prediction model, the proposed system has achieved challenging results in terms of efficiency and accuracy.

1. Introduction

Financial prediction such as stock prediction and foreign currency exchange rate forecast so far is one of the most important and hottest topics, both in the scientific and financial fields. In fact, due to the importance of this particular topic, a well-established school of concepts and techniques have been devised in the previous decades, namely the technical [Murphy, 1986] and fundamental analysis [Ritchie, 1996] techniques. However, owing to the fact that these tools are based on totally different approaches of analysis, they always give rise contradictory results. Besides, 'pattern matching' technique is still widely used for most financial analysts and market technicians. Actually, there are numerous market patterns that have found (and believe) to exhibit repeated market moves following their occurrence. So they are especially useful for long-term financial trend prediction. These market patterns can be classified into two major categories: 'reversal patterns' and 'continuation patterns' [Zirilli, 1997]. The correct identification of these patterns is a very useful information to predict the financial trend movement. However, the identification of these patterns so far is highly subjective

and prone to errors. The major obstacles for the automation of this pattern matching process is that these patterns are highly variant and 'elastic' in the sense that they can exist under different appearances.

In this paper, we present an integrated neural network-based financial prediction system which fully automates the process of 1) long-term trend prediction and 2) short-term (e.g. one-day) stock forecast. Unlike the contemporary neural network-based financial prediction model, we have integrated and automated critical technical analysis tools into the proposed model, namely the market pattern identification scheme in trend prediction and oscillatory-based RSI (relative strength index) analysis technique. From the implementation point of view, stock information for 33 major Hong Kong stocks in the period from 1990 to the end of 1999 (ten-year data) have been used. Compared with other contemporary time series neural network prediction models including the feedforward backpropagation model from NeuroForecaster and Genetica, NORN has achieved challenging results in terms of efficiency, accuracy and the ability to integrate critical technical analysis techniques.

2. NORN: system framework

2.1 System overview

The neural oscillatory-based recurrent network model (NORN) proposed in this paper is based on the integration of critical technical analysis techniques 1) Market pattern analysis and 2) Oscillator-based system with hybrid RBF (Radial Basis Function) recurrent network for stock prediction. NORN basically consists of the following modules:

- Automatic market analysis module using NOEGM model;
- Hybrid RBF recurrent network with TDSL (Time-Difference Structural Learning) for stock prediction.

The major objective of NORN on one hand is the provision of a fully automatic stock prediction model (based on an integrated recurrent neural network), and on the other hand, to integrate it with critical technical analysis techniques to increase the degree of accuracy.

2.2 Market pattern analysis via NOEGM

NOEGM – system overview

The Neural Oscillatory Elastic Graph Matching (NOEGM) model consists of three main modules: 1) Multi-frequency bands feature extraction for the stock pattern using Gabor filters; 2) Automatic figure-ground TC pattern segmentation using Neural Oscillatory model; 3) Invariant market pattern matching using Elastic Graph Dynamic Link Model (EGDLM).

Actually NOEGM model has been applied to various invariant pattern recognition applications including face recognition [Lee *et al.*, 1999], scene analysis [Lee and Liu, 1999a] and the latest research on tropical cyclone (TC) pattern recognition from the satellite pictures [Lee and Liu, 1999b; 2000], based on composite neural oscillators as neural framework. A schematic diagram for the NOEGM model is shown in Figure 1.

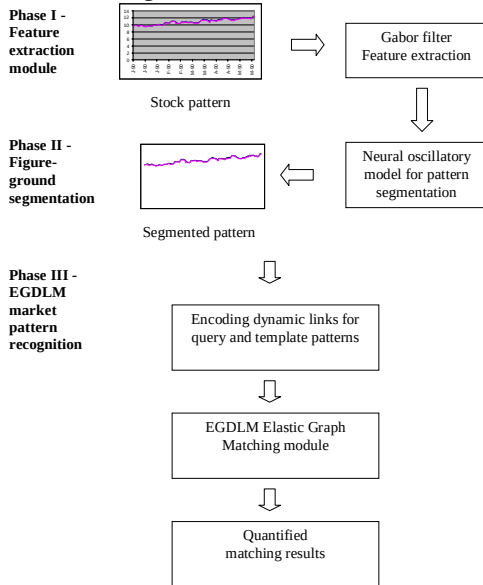


Figure 1 – Schematic diagram for the NOEGM model for market pattern recognition

Feature extraction module

In this module, Gabor filters of 10 different frequency bands (ϕ) and 5 different orientations (θ) are being used, a total of 50 feature vectors of different attributes are extracted from the stock patterns. The filter function is given:

$$g_{\phi,\theta}(x,y) = \frac{1}{\sigma\sqrt{\pi}} e^{-\frac{x^2+y^2}{2\sigma^2}} e^{2\pi i \phi(x\cos\theta+y\sin\theta)} \quad (1)$$

Pattern segmentation – neural oscillatory model

In the neural oscillatory model, neurons in our visual cortex are presented by numerous neural oscillators. A typical neural oscillator consists of an excitatory neuron (u_i) and an inhibitory neuron (v_i), which interacts and oscillates with other neurons according to external stimulus. In the model, each layer represents the whole visual horizon in 2D perspective. In each location, the “column” neural oscillators represent the set of “visual sensors”, which is modeled by an array of feature vectors.

According to the model, with the presence of a subject in an image (eg. stock pattern), neurons that belong to the same segment are oscillated in a phase-locked mode with zero phase shift after sufficient cycles of neural oscillations, while neurons in different segments are oscillated out of phase and uncorrelated [Lee and Liu, 1999a].

In the neural oscillatory model, each stock pattern graph is broken into a 2D mesh of $N = N_1 \times N_2$ composite neural oscillator sites. Each site (column) consists of M layers of neural oscillators which denote the neural oscillation from each local feature response (e.g. using Gabor Filters). In addition to the local excitatory and inhibitory neurons, the neural dynamics of the composite neural oscillators are activated/deactivated by the oscillatory model.

“Vertical & horizontal” excitatory connections

Neural oscillators within the same “column” are mutually activated with strength W_i . In each feature layer, each neural oscillator is activated by the eight “closest” neighboring excitatory composite neurons with strength W_- .

“Vertical & horizontal” inhibitory neurons (v_i & v^q)

To control the unexpected local phase locking, “Vertical” and “Horizontal” inhibitors are introduced, which are governed by the inhibitory strength $T_i v_i$ & $T_- v^q$ respectively.

“Global” inhibitory neurons (v)

A global inhibitory neuron which receives excitation from all excitatory neural oscillators, and it inhibits all excitatory units with the signal T_v . The neural dynamics are shown as follows:

$$\frac{du_i}{dt} = -u_i^q + S_\lambda(u_i^q - \beta v_i^q - \theta_u + I_{ci}^q + W_i^q u_i^q + W_-^q u_i^q - T_i^q v_i - T_-^q v^q - T_v^q v) \quad (2)$$

$$\tau \frac{dv_i^q}{dt} = -v_i^q + S_\lambda(\alpha u_i^q - \gamma v_i^q - \theta_v) \quad (3)$$

$$\tau \frac{dv_i}{dt} = -v_i + S_\lambda(W_i v_i - \theta_v) \quad (4)$$

$$\tau \frac{dv^q}{dt} = -v^q + S_\lambda(W_-^q v^q - \theta_v) \quad (5)$$

$$\tau \frac{dv}{dt} = -v + S_\lambda(W' v - \theta_v) \quad (6)$$

C) Segmentation criteria: correlation function σ_{xy}

The segmentation criteria is governed by the “Correlation Factors”, which are a measurement of the binding strength (phase relationship) between the composite neural oscillators and their “nearest” neighbors, given by:

$$\sigma(\bar{x}, \bar{y}) = \frac{\langle \bar{x} \bar{y} \rangle - \langle \bar{x} \rangle \langle \bar{y} \rangle}{\sqrt{\langle \bar{x}^2 \rangle - \langle \bar{x} \rangle^2} \cdot \sqrt{\langle \bar{y}^2 \rangle - \langle \bar{y} \rangle^2}} \quad (7)$$

Market pattern recognition – elastic graph dynamic link model (EGDLM)

Object recognition based on EGDLM which described the recognition problem as an elastic graph matching mechanism between the attribute graphs of the image vectors (input layer) with the set of “memory” graphs in the object gallery (memory layer). The schematic diagram of

EGDLM network architecture for market pattern matching is shown in Figure 2.

One of the most striking features of the EGDLM is its “invariant” property. In the network model, only the topological relations between the neural oscillators (i.e. dynamic links) are encoded into the network. The pattern matching process is resembled to the ‘Elastic Graph Matching’, which is invariant under various transformations such as translation, rotation, reflection, dilation and occlusion, commonly occurred in pattern recognition.

EGDLM object recognition model consists of two processes: 1) dynamic link initialization and 2) elastic graph matching module. In dynamic link initialization process, dynamic links ($z_{ij,kl}$) between “memory” object graphs (obtained from market pattern templates) and query pattern are initialized according to the following rules:

$$z_{ij,kl} = \varepsilon J_{ij} J_{kl} \quad (8)$$

for $J_{ij} \in A$, $J_{kl} \in B$.

In the elastic graph matching module, attribute graph of the figure pattern (with neural oscillator vectors as nodes and correlation links as edges) is matched with each of the attribute graphs in the object gallery by minimizing energy function $H(z)$:

$$H(z) = - \sum_{i,j \in B, k,l \in A} z_{ij} z_{kl} z_{ik} z_{jl} + \gamma \sum_{i \in B} \left(\sum_{k \in A} z_{ik} - 1 \right)^2 + \gamma \sum_{k \in A} \left(\sum_{i \in B} z_{ik} - 1 \right)^2 \quad (9)$$

where $H(z)$ is minimized using the gradient descent.

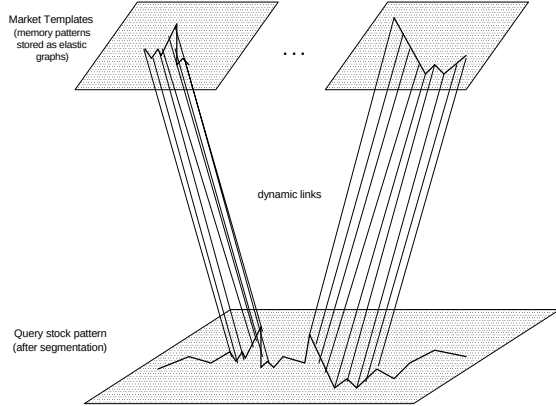


Figure 2 – EGDLM Network architecture for market pattern matching

2.3 Hybrid radial basis function networks for nonlinear time series stock prediction

Hybrid RBF network (HRBFN) – system overview

The proposed Hybrid RBF Network (HRBFN) [Lee and Liu, 2000] incorporates with two main technologies into the conventional RBF network for temporal time series prediction problem: 1) Structural learning technique that integrates the “forgetting” factor into the RBF BP algorithm [Ishikawa; 1996]; 2) A ‘Time Difference with Decay’ (TDD) method is incorporated into the network to strengthen the temporal time series relation of the input data sequence for network training.

The HRBFN consists of three layers. The first layer is the input layer which consists of two portions: 1) Past network outputs that feedback into the network; 2) Major co-relative variables are concerned with the prediction problem. Past network outputs enter into the network by time-delay unit as the first inputs. These outputs are also affected by a decay factor γ that is governed by the following equation.

$$\gamma = \alpha e^{-\lambda k} \quad (10)$$

In general, the time series prediction of the proposed network is to predict the outcome of the sequence x_1^{t+k} at the time of $t+k$ that is based on the past observation sequence of size n , i.e. $x_1^t, x_1^{t-1}, x_1^{t-2}, x_1^{t-3}, \dots, x_1^{t-n+1}$ and the major variables that influence the outcome of the time series at time t . For convenience, the following notations are used throughout the following network description: The numbers of input nodes in the first and second portions are set to n and m respectively. The number of hidden nodes is set to p . The predictive steps are set to k , so the number of output nodes is k . At the time t , the input will be $[x_1^t, x_1^{t-1}, x_1^{t-2}, x_1^{t-3}, \dots, x_1^{t-n+1}]$ and $[x_2^1, x_2^2, \dots, x_2^m]$ respectively. The output is given by x^{t+k} , denoted by p_k^t for simplicity, w_{ij}^t denotes the connection weight between the i -th node and the j -th node at time t .

HRBFN – structural learning algorithms

The main idea of RBF learning algorithm with “forgetting” factor is to introduce a constant decay to connected weights that make the redundancy weight(s) fade out quickly. The cost function of the structural learning algorithm is given by equation (11).

$$E^t = E_1^t + \varepsilon \sum_{i,j} |w_{ij}^t| \quad (11)$$

where E_1^t denotes the error square in traditional RBF learning, the second term is the penalty criteria.

If delta rule is used, the learning rule of the weights is given by:

$$\Delta w_{ij}^{t+1} = -\eta \frac{\partial E^t}{\partial w_{ij}^t} + \alpha \Delta w_{ij}^t \quad (12)$$

HRBFN – TDD method

The structural learning algorithm discussed above does provide a “dynamic” structure building of the neural network, but it cannot adapt the temporal time series relations of the input and output feedback data sequences into the model. In order to code with this problem, a temporal difference method with decay feedback is hybridized into the learning algorithm of the proposed model. The basic concepts are presented as follows.

In a typical time series prediction problem, given a series of past observations of time-step n at time t , i.e. $[x_1^t, x_1^{t-1}, x_1^{t-2}, x_1^{t-3}, \dots, x_1^{t-n+1}]$, with the predictive time-step of k , we not only obtain the predicted output at time $t+k$, i.e. p_k^t , but more importantly is the sequence of future events started from time t , i.e. $[p_1^t, p_2^t, p_3^t, \dots, p_k^t]$. In other words, the network can provide an overlapping and inter-related event

sequence as an additional “hint” for network learning, which can be implemented by using Temporal Difference technique [Sutton, 1983]. Besides, consider a sequence of temporal difference operations from time $t+1$ to $t+k$, the prediction from a “nearer” future normally has a higher level of confidence than a “far” future, so a decay operator is integrated into the learning algorithm in order to reflect the situation. With the integration of TDD methodology, the learning algorithm discussed in equation (12) will be modified into:

$$\Delta w_{ij}^{t+1} = \eta \left(\sum_{h=2}^k \gamma(h) \cdot (p_{h-1}^{t+1} - p_h^t) \sum_{l=1}^t \lambda^{l-1} \frac{\partial p_h^l}{\partial w_{ij}^t} + (p_1^t - p_0^t) \cdot \right. \quad (13)$$

$$\left. \sum_{l=1}^t \left[\gamma(l) \cdot \lambda^{l-1} \frac{\partial p_l^t}{\partial w_{ij}^t} \right] + \alpha \Delta w_{ij}^t - \varepsilon'' \text{sgn}(w_{ij}^t) \right)$$

where ε'' is defined as:

$$\varepsilon'' = \begin{cases} \varepsilon & |w_{ij}^t| < \theta \\ 0 & \text{otherwise} \end{cases} \quad (14)$$

3. Implementation

3.1 NORN for stock prediction

Introduction

In section 2 we have discussed the critical components of NORN for automatic time series prediction and pattern recognition. In this section, we illustrate how these technologies can be integrated to stock prediction in the following ways: 1) long-term stock trend prediction; 2) short-term stock price prediction. A schematic diagram for the integrated NORN model for stock prediction is shown in Figure 3.

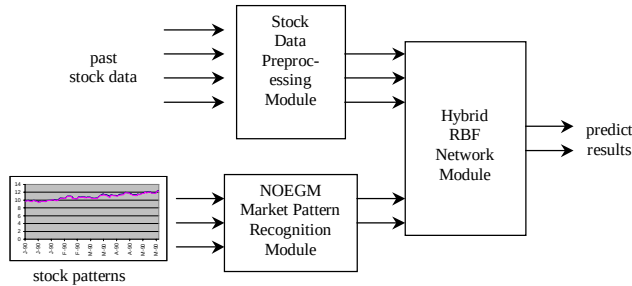


Figure 3 – Schematic diagram of the integrated NORN model for stock prediction

The integrated NORN model consists of the following main modules:

- 1) Stock data pre-processing module;
- 2) NOEGM market pattern recognition module;
- 3) Hybrid RBF network module for stock prediction.

Stock data pre-processing module

From the implementation point of view, ten-year daily stock data (1990-1999) of 33 major Hong Kong stocks are being adopted. Daily information includes daily high, low and closing stock value. The inputs of the HRBF recurrent network is generated by 1) pre-processed time series stock data, and 2) market pattern recognition results.

According to the different requirements and forecasting purpose of NORN for stock prediction, the time series stock

data are processed differently for the long-term trend prediction and short-term price forecasting.

A) Data pre-processing (feature extraction) for long-term trend prediction

For long-term trend prediction, since we are interested in stock pattern variation instead of daily price fluctuation, two types of inputs (pre-processed) are being used: 1) normalized daily closing stock value (x_i); and 2) normalized ‘desired-value’ for days between a buy and a sell signal (d_i) given by:

$$x_j = \frac{c_i - c_{i-j}}{c_i \lambda_j}, j=1 \dots A, -1 \leq x_j \leq 1 \quad (15)$$

$$\text{where } \lambda_j = \frac{2}{P-A} \sum_{i=A+1}^P \frac{|c_i - c_{i-j}|}{c_i}, j=1 \dots A \quad (16)$$

$$d_i = \frac{c_i - c_{low}}{c_{high} - c_{low}} \quad (17)$$

The main purpose of this data pre-processing (feature extraction) method given by equation (15) is to extract the difference between the current day’s closing price (c_i) and the previous A closes (c_{i-j}). This difference is then normalized by dividing it with the closing price and the scaling factor λ .

In order to ‘feed-in’ the critical signal for trend prediction, the normalized ‘desired-value’ for the days between a buy and a sell signal (d_i) is being adopted. Actually, this signal is based on the concept of ‘oscillator-based system’ of the technical financial analysis discussed previously, a kind of ‘neural-network oscillator’ for stock prediction. For example, when this neural net oscillator rises above a sell threshold, it signals an alert that the market is approaching a trend change to the down side. On the other hand, when the neural net indicator falls below a buy threshold, it signals an alert that the market is approaching a trend changes to the up side. So, these oscillatory input nodes can provide an effective indicator for trend prediction.

B) Data pre-processing (feature extraction) for short-term price prediction

Unlike long-term trend prediction, short-term stock price prediction focuses on the fluctuations of the stock value on daily basis. So, the strategy for creating input patterns is to use the daily change in price (Δ values for open, close, high and low) to predict the next day’s high, low and close.

Using similar normalization approach, for each different type of Δ values (open, close, high and low), the volatility factor λ can be calculated as follows:

$$\lambda = \frac{2}{P} \sum_{i=1}^P \frac{|\Delta|}{c_i} \quad (18)$$

and each normalized input value is given by:

$$x_i = \frac{\Delta}{c_i \lambda}, i=1 \dots A, -1 \leq x_i \leq 1 \quad (19)$$

The desired outputs ($d_{h,i}$, $d_{l,i}$, $d_{c,i}$) for the high, low and close values are also normalized in the similar approach as shown below:

(20-22)

$$d_{h,i} = \frac{h_{i+1} - c_{i+1}}{c_i \lambda_h}, d_{l,i} = \frac{c_{i+1} - l_{i+1}}{c_i \lambda_l}, d_{c,i} = \frac{c_{i+1} - c_i}{c_i \lambda_c}$$

Another important fact for this kind of normalization is, once the predicted outputs (o_{h+1} , o_{l+1} , o_{c+1}) are obtained, the predicted stock prices can be transformed easily based on the following equations:

$$c_{i+1} = 2 c_i \lambda_c (o_c - 0.5) + c_i \quad (23)$$

$$h_{i+1} = c_{i+1} + c_i \lambda_h o_h \quad (24)$$

$$l_{i+1} = c_{i+1} - c_i \lambda_l o_l \quad (25)$$

NOEGM market pattern recognition module

By using the neural oscillatory-based elastic graph matching (NOEGM), market patterns from the time series stock pattern for each stock can be extracted and recognition based on the market pattern templates for the 1) major reversal patterns and 2) major continuation patterns being shown in figures 1 and 2 respectively. From the implementation point of view, according to the type of market patterns being extracted from the time series stock price pattern, they are grouped into a set of three different input nodes: 1) reversal Top pattern node; 2) reversal Bottom pattern node; and 3) continuation pattern node.

Hybrid RBF network module for stock prediction

According to the schematic diagram of NORN model, time series stock information of 33 Hong Kong major stocks in a period from 1990 to the end of 1999 are 'fed in' to the hybrid RBF network in the following two ways: 1) pre-processed time series stock data for each stock based on the normalization and transformation techniques discussed previously, with windows size ranging from 10 to 45 days (for long-term trend prediction) and 1 to 10 days (for short-term stock prediction); 2) market patterns being extracted and identified from the time series stock patterns, which is quantified by the correlation values discussed previously.

3.2 Experimental results

From the system validation and performance evaluation point of view, three different types of experimental tests are conducted:

- 1) Market pattern recognition test for NOEGM validation;
- 2) Long-term/short-term window size evaluation test;
- 3) NORN model performance test.

Market pattern recognition test

In the test, four representative stocks from the four main business sectors (i.e. banking, finance investment, public utility and property) were used for system evaluation. They were: HSBC Holdings (banking), Tian On China Investment (finance investment), Hong Kong Telecom (public utility) and Cheung Kong Holdings Ltd. (property). Based on the major market patterns being shown in figures 1 and 2 and the stock pattern for these stocks in the period from 1990 to

1993. The automatic pattern recognition result based on CNOM model is shown in table 1.

Table 1 - Market pattern recognition test results

Market Patterns	Market patterns recognition rate *				
	HSBC Holdgs.	Tian On China Invest.	Hong Kong Telecom	Cheung Kong Holdgs.	Overall
A) Reversal patterns (Top)	182/189 (96%)	160/170 (94%)	169/182 (93%)	170/184 (92%)	93.93%
B) Reversal patterns (Bottom)	154/167 (92%)	159/173 (92%)	163/182 (90%)	184/207 (89%)	90.53%
C) Continuation patterns	318/349 (91%)	296/327 (91%)	296/335 (88%)	266/301 (88%)	89.63%
Overall	654/705 (93%)	615/670 (92%)	628/699 (90%)	620/692 (90%)	91.00%

Note * The pattern recognition rates from different patterns are denoted by: Matched cases / Total cases (recognition rate %)

As illustrated in Table 1, out of a total of 2766 market patterns being identified for these four stock pattern series, the correct matching rate is over 90%, with a total of 2517 correct matched patterns. The overall recognition rate ranging from 89% (for continuation patterns) to about 94% (for reversal Top patterns).

Actually, in view of the detailed recognition rate of the sub-category pattern for each type of market patterns, the recognition rates maintain an acceptable level of around 85 to 96%, except for some particular categories such as the 'Round bottom' pattern in the reversal bottom patterns and the 'Rectangles' pattern in the continuation patterns.

There is one important point needed to bring out and is it the 'basis' of the recognition test - that is, the 'Total cases' figures are identified by human subjective justification which can be explained why some categories have scored a lower pattern, as there exist a certain degree of ambiguity in some template patterns.

For the processing speed of pattern recognition in our test, the average time for pattern segmentation and recognition are 35s and 12s respectively, as compared with the previous work based on Composite Neural Oscillatory Model (CNOM) for cloud pattern recognition [Lee and Liu, 2000], the pattern segmentation and recognition are 50s and 20s corresponding to an improvement of 30 and 40% respectively due to lesser degree of complexity of the proposed model [Lee and Liu, 1999b].

Long-term/short-term window size evaluation test

As the window size of the stock prediction model can provide a deterministic factor for the network efficiency and accuracy, this experimental test aimed at the evaluation of the optimized window size for 1) long-term trend prediction and 2) short-term stock price prediction.

In the test, window sizes of 10 to 45 days (for long-term trend prediction) and 2 to 10 days (for short-term price prediction) were used respectively. The predicted output for these two sets of predictions are as follows:

- 1) Long-term trend prediction:
 - 'Predicted desired-value (d_i)' for days between a buy and a sell signal.
- 2) Short-term price prediction:
 - 'Predicted next-day High (h_{i+1})
 - 'Predicted next-day Low (l_{i+1})
 - 'Predicted next-day Close (c_{i+1})

Based on the four typical stock items used in the previous test. Table 2 shows the experimental results in terms of average percentage error of the NORN model under different window sizes.

Table 2 - NORN model window size evaluation test results

Long-term trend prediction		Short-term price prediction	
Window size (days)	Av. % error	Window size (days)	Av. % error
10	2.345%	2	2.347%
15	1.075%	3	1.473%*
20	1.073%*	4	1.476%
25	1.238%	6	1.487%
30	2.487%	8	2.746%
40	5.347%	10	3.248%

As illustrated in Table 2, the optimal window size for long-term and short-term stock prediction are 20 days and 3 days respectively, with average percentage errors of 1.075% and 0.473% being achieved.

NORN model performance test

In this test, the ten-year (1990-1999) stock information of 33 major Hong Kong stocks were adopted. For comparison purpose, a neural network based forecasting aid 'NeuroTM Forecaster' from Neuro Intelligent Business Software was adopted in the test. Actually, NeuroTM Forecaster provides the following neural network models:

- 1) Time series feed-forward back-propagation model (FFBP) with different choices of transfer functions, which includes: standard sigmoid function, hyperbolic tangent function, Neurofuzzy function, etc.
- 2) 'Genetica Net Builder' - based on Genetic Algorithms (GA) for the construction and optimization of the network model.

Table 3 - NORN model performance test results
(For next-day stock price prediction)

Business Type	Neuro Forecaster (Average % error)				NORN (Av. % error)
	Sigmoid	Hyperbolic tangent	Neuro-Fuzzy	Genetica	
A) Banking	4.647%	3.457%	8.451%	3.427%	1.427%
B) Finance & investment	3.745%	3.758%	7.845%	4.747%	1.573%
C) Public utility	3.412%	4.285%	10.457%	2.747%	1.347%
D) Property	4.452%	5.474%	7.457%	3.417%	1.417%
E) Others	5.124%	4.789%	8.982%	3.746%	1.379%
Overall	4.217%	4.648%	8.975%	3.417%	1.401%

For the ease of comparison, these 33 stock items are grouped under the four critical business sectors, namely: banking, finance and investment, public utility, property and others. The experimental results for the next-day stock price prediction are shown in Table 3.

4. Conclusion

In this paper, we have presented an innovative, fully automatic and integrated stock prediction model to serve the purposes of 1) long-term stock trend prediction, and 2) short-term stock price forecast. Unlike contemporary neural network models for financial prediction [Vellido *et al.*, 1999] which focus on the 'replacement' of the conventional financial analysis and prediction techniques with various neural net technologies, this paper explores a new era for the implementation of neural networks in financial engineering in which neural networks can be successfully integrated with conventional financial analysis techniques such as technical analysis tools to improve the efficiency and effectiveness of the financial prediction results.

Acknowledgment

The authors are grateful to the partial supports of the Departmental Grant #H-ZJ87, Central Research Grant #PolyU 5091/00E and the iJADE projects of Hong Kong Polytechnic University.

References

- [Ishikawa, 1996b] M. Ishikawa. Structural Learning with Forgetting. *Neural Networks*, 9(3) 509-521, 1996.
- [Lee *et al.*, 1999] R. Lee, J. Liu and Y. You. Face Recognition: Elastic Relation Encoding and Structural Matching. In *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics (SMC'99)*, Tokyo, Japan, pages 172-177, 1999.
- [Lee and Liu, 1999a] R. S. T. Lee and J. N. K. Liu. An Oscillatory Elastic Graph Matching Model for Scene Analysis. In *Proceedings of International Conference on Imaging Science, Systems, and Technology (CISST'99)*, Las Vegas, USA, pages 42-45, 1999.
- [Lee and Liu, 1999b] R. S. T. Lee and J. N. K. Liu. An Automatic Satellite Interpretation of Tropical Cyclone Patterns Using Elastic Graph Dynamic Link Model. *International Journal of Pattern Recognition and Artificial Intelligence*, 13(8) 1251-1270, 1999.
- [Lee and Liu, 2000] Raymond. S. T. Lee and James N. K. Liu. Tropical Cyclone Identification and Tracking System Using Integrated Neural Oscillatory Elastic Graph Matching and Hybrid RBF Network Track Mining Technique, *IEEE Transaction on Neural Networks*, 11(3) 680-689, 2000.
- [Murphy, 1986] J. J. Murphy. Technical Analysis of the Future Markets. *The New York Institute of Finance*, Prentice Hall, New York, 1986.
- [Ritchie, 1996] J. Ritchie. *Fundamental analysis: a back-to-the-basics investment guide to selecting quality stocks*, Chicago, Irwin Professional Pub, 1996.
- [Sutton, 1983] R. S. Sutton. Learning to Predict by the Methods of Temporal Difference. *Machine Learning*, No. 3, pages 8-44, 1983.
- [Vellido *et al.*, 1996] A. Vellido, P. J. G. Lisboa and J. Vaughan. Neural Networks in Business: A Survey of Applications (1992 - 1998). *Expert Systems with Application*, 17(1) 51-70, 1999.
- [Zirilli, 1997] J. S. Zirilli. *Financial Prediction using Neural Networks*, International Thomson Computer Press, 1997.