

PAPER • OPEN ACCESS

A Hybrid Chaotic Oscillatory Neural Network (HCONN) Based Financial Time Series Prediction System

To cite this article: Yifu Qiu and Raymond S. T. Lee 2019 *IOP Conf. Ser.: Mater. Sci. Eng.* **646** 012024

View the [article online](#) for updates and enhancements.

A Hybrid Chaotic Oscillatory Neural Network (HCONN) Based Financial Time Series Prediction System

Yifu Qiu¹ and Raymond S. T. Lee^{1,*}

¹ Beijing Normal University-Hong Kong Baptist University United International College, 2000 Jintong Road, Tangjiawan, Zhuhai, Guangdong Province, China

*Corresponding author email: raymondshtlee@uic.edu.hk

Abstract. Financial time series prediction is one of the most complex and challenging problems in both AI and finance engineering. In our research, we proposed a Hybrid Chaotic Oscillatory Neural Network (HCONN) model by replacing the traditional sigmoid-based activation function with chaotic oscillatory activation function, which provides significant performance in the global minimum convergence through the application of Adaptive Moment Estimation optimizer. In addition, by integrating the latest R&D on Quantum Finance Theory (QFT) and its Quantum Price Level (QPL) as the deep features' extraction, we add the daily 8 nearest QPLs together with the time series price variables as the input of our HCONN. In terms of system implementation, 12 different forex products including the AUDCHF, AUDUSD, CADCHF, EURAUD, EURCHF, EURGBP, EURUSD, GBPAUD, GBPCAD, GBPUSD, USDCAD and USDCHF are used. System performance results reveal that HCONN outperforms other financial models including: Feedforward Backpropagation Neural Network (FFBPN) and Chaotic Oscillatory Neural Network (CONN) in terms of training performance and forecast accuracy.

1. Introduction

Nowadays, one of the major areas for artificial intelligence (AI) applied in finance is the analysis and prediction of the financial time series, which is also the trend and the hottest topic for many decades in both financial industry and academia. In the past decades, AI tools such as machine learning and deep learning, are popularly utilized in financial time series analysis and forecasting, which have also been proved with high degree of validity and correctness [1].

In general, prediction of financial instrument can be categorised into two related directions based on the scope we observe the market: long-term market trend prediction and short-term price prediction. In long-term trend prediction, we try to predict the future trend of the market movement according to the selected financial variables such as technical indicators in a certain previous trading day timeframe. In the short-term prediction, we try to predict the price in the future day based on the previous several days price data. These two kinds of prediction are tightly related with each other in finance, the long-term market trend prediction provides the direction for the short-term prediction. On the other hand, short-term prediction can be considered as a kind of microscopic analysis of the long-term prediction.

For both kinds of prediction in financial time series data, the essence is that we construct a learning algorithm model which could learn and extract some key patterns from the large volume of the past financial time series. Then based on these patterns and current financial time series, we apply machine learning algorithms to do the pattern matching or so-called prediction for future market movement.



In machine learning perspective, Artificial Neural Networks (ANN) and Deep Neural Networks (DNN) are commonly used on financial time series prediction. In summary, ANN is a kind of simulation of the human brain's operation by simulating the connection among various neurons in the human brain. The neural dynamics are represented by the regulating of the weights between neurons and transfer of information by means of activation function. Through network training, ANNs can learn complex patterns and perform various classification and even prediction operations [2]. In short, DNN can be considered as a kind of extension of ANN. In terms of network structure, DNN possesses more complex hidden layers and neural dynamics, which believes can be used to tackle with more complex problem, financial prediction for example [3].

With the development of neural networks, the architecture of contemporary neural networks is no longer limited in the fully connected neural network structure. Other neural networks structure such as Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN) and Generative Adversarial Networks (GAN) are gradually demonstrating their powerful capabilities in various fields of application. Specially, in finance, recurrent neural networks which intuitively accept the time series as input, have gradually become the most popular neural networks in the financial industry and academia.

In this paper, we proposed an integrated financial prediction system called Hybrid Chaotic Oscillatory Neural Network (HCONN), which integrates the Chaotic Oscillatory Neural Network (CONN) and Quantum Price Levels (QPL) from the latest research on Quantum Finance Theory (QFT) [4], which are considered as the intrinsic properties of the financial markets. From the implementation perspective, HCONN takes the nearest 8 QPLs and the past 9 trading-days timeseries of open, high, low and closing prices as inputs to forecast the next-day high and low prices. For system testing, 2048 trading-day timeseries from 5 Oct 2012 to 12 May 2019 are used, together with 12 major forex products include: AUDCHF, AUDUSD, CADCHF, EURAUD, EURCHF, EURGBP, EURUSD, GBPAUD, GBPCAD, GBPUSD, USDCAD and USDCHF for system implementation. For performance analysis, HCONN is compared with Feedforward Backpropagation Neural Network (FFBPN) and Chaotic Neural Network (without QPL) as benchmarks for training performance and forecast accuracy performance testing.

2. Literature Review

Since using neural network to do the forecasting for the financial time series is a popular and developing topic in the application of neural network, there are substantial number of scholars and studies working in this field. In this literature review, the authors will attempt to summarize those significant works on the financial time series data in the past few years.

2.1. Quantum Finance Theory (QFT)

In order to extract the deep features of each financial time series data, we apply the author's latest research on Quantum Price Levels (QPL) of the Quantum Finance Theory (QFT)[4]. By modelling the market movements in the secondary financial market using quantum anharmonic oscillator (QAO), and through the solving of Quantum Finance Schrödinger Equation (QFSE), we calculate the energy levels for the Quantum Financial Particle (QFP) in each financial market directly – the QPL. According to the QFT, QPL could be used in both financial forecasting and trading strategy design as analogue to the resistance and support lines in technical analysis. As compared with traditional features we used in the prediction problem, QPL reflects the nature of the particle's dynamic, which sheds new lights in the exploration of the intrinsic characteristics of the QFPs in financial markets.

2.2. Chaotic Oscillatory Neural Network (CONN)

Due to the highly degree of chaotic and complex fluctuation of financial markets, they are difficult to be modelled successfully by traditional neural networks. According to our experiment, by using the traditional activation function in neural network such as ReLU, tanh and sigmoid functions, the fitting results for the financial time series are not so promising, so we introduce the Lee-oscillator model [5].

Lee-oscillator is a chaotic neural oscillator with remarkable transient-chaotic property which provides a perfect solution to serve as Bifurcation Transfer Unit (BTU) to model highly chaotic and complex problems such as financial timeseries prediction [6].

3. Methodology

From the system framework perspective, HCONN is the integration of: 1) 8 closest Quantum Price Levels (QPL) evaluated from quantum finance model; 2) 9-day time series of the financial products; 3) Chaotic Oscillatory Neural Network (CONN) model with the adoption of Lee-oscillators as model framework.

3.1. Quantum Price Levels (QPL)

With the continuous development of the econ-physics in the recent years, Quantum Finance Theory (QFT) based on Quantum Mechanics and Quantum Field Theory to extract the deep features of the motion of the quantum financial particles (QFP) become one of the hottest topics in Fintech and quantum computing technology. Based on the author's latest research on QFT, the Quantum Finance Schrödinger Equation (QFSE) is given by [5]:

$$\left[\frac{-\hbar}{2m} \frac{d^2}{dr^2} + \left(\frac{\gamma\eta\delta}{2} r^2 - \frac{\gamma\eta\nu}{4} r^4 \right) \right] \psi(r) = E\psi(r) \quad (1)$$

In QFSE, the first term denotes the kinetic energy term, and the second term represents the potential energy term of the quantum finance market, $\psi(r)$ is the wavefunction and E represents the corresponding quantum energy levels of the QFPs. By solving QFSE, we could calculate the discrete QPLs which directly analogue to the energy level of the QFPs. From the perspective of the QFT, if there is no outside stimulus, the financial particle will remain at its own energy level and perform chaotic oscillation. Once there is outside stimulus, the QFPs will discretely absorb/release the quantized energy and jump to other QPLs.

In our financial forecasting problem, we take both time series price variables and the QPLs as the inputs since the QPL is a direct reflection corresponding to the result of price movement, which symbolizes the possibility and stages of the movement of financial product prices within the trading day. In this way, we extract the deep characteristics of price movements by calculating the QPLs and input it into the chaotic neural network together with the time series price variables for system training and next-day prediction.

3.2. Time series to time sequence

In order to tackle the gradient descent deadlock problem during the training period and reflect the intrinsic relationship among different prices and QPLs, many methods have been proposed. In our research, at the pre-processing phase, we do the normalization for the time series data based on the Lasso normalization method and the equation for normalization factor is given as follow:

$$x' = \frac{x}{\sum_{i=1}^n |x_i|} \quad (2)$$

After applying this normalization method to the timeseries, we reflect our two groups of feature values into the range between 0 and 1. In this way, we could maximize the usage of our chaotic activation function and its chaotic region to jump out from the local minimum which we will show in the following section.

In order to forecasting the future price movement, we train the HCONN by capturing the patterns of price movements in the past 2048 trading-days. Therefore, in the process of the data pre-processing, we spilt the time series continuously through a sliding window method shown as followed, which enables us to get 2038 fixed scale training samples. We take the past 9-day daily OHLC (Open, High, Low and Close) financial time series variables as input and predict the next day highest price and

lowest price because in the actual trading, daily high/low prices are commonly for reversal and breakout trading strategy. The working mechanism of the sliding window is given in Figure 1.

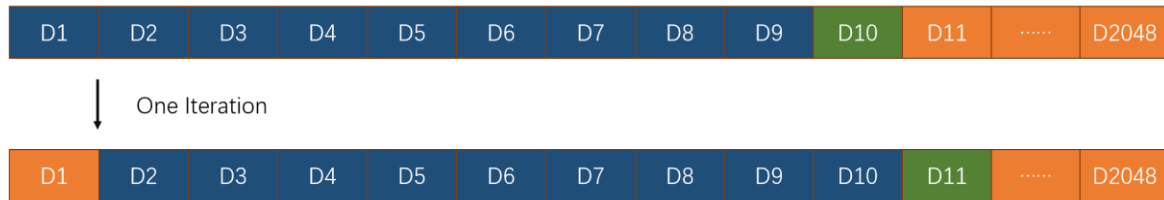


Figure 1. Spilt time series into sequence based on 10-day fixed size sliding window.

In order to prevent the market trend influence of our model to learning the past price movement pattern, for example, if the market is in the bullish or bearish, the divided 10-day sequence's pattern might be influence by the current market trend, and ensure that our model has sufficient generalization predictive ability in different market condition, we randomly choose the sequence and append it into the training set and testing set. In here, we follow the traditional 75% and 25% rule to construct the training set and testing set.

3.3. Hybrid Chaotic Oscillatory Neural Network (HCONN)

The structure of our chaotic neural network follows the traditional fully-connected proposed by Rumelhart and McClelland in 1986, which is the fundamental structure for all the neural networks but contains power fitting ability to the complex and non-linear data. In our Hybrid Chaotic Oscillatory Neural Network (HCONN), by considering with the volume of the training data, we construct an eight-layer based fully connected neural network containing three chaotic oscillatory layers, three batch normalization layers and two dropout layers, for each layer it contains 20 neurons.

In our HCONN, the input layer contains 102 neurons which includes two major parts: 1) 9-day time series data, which include Open, High, Low and Closing prices; 2) each day's 8 nearest Quantum Price Levels (QPLs). The 8 hidden layers including three different types: 1) Chaotic oscillatory layer, 2) Batch normalization layer and 3) Dropout layer. The output layer contains 2 neurons corresponding to the next day High/Low prices.

In order to solve the problem in fitting the highly random and stochastic time series data, we adopt the Lee-Oscillator [5] as the activation function in our HCONN. Its neural dynamics is given by:

$$E(t+1) = \text{sigmoid}(e_1 \cdot E(t) - e_2 \cdot I(t) + S(t) - \zeta_E) \quad (3)$$

$$I(t+1) = \text{sigmoid}(i_1 \cdot E(t) - i_2 \cdot I(t) - \zeta_i) \quad (4)$$

$$\Omega(t+1) = \text{sigmoid}(S(t)) \quad (5)$$

$$L(t) = [E(t) - I(t)] \cdot e^{-kS^2(t)} + \Omega(t) \quad (6)$$

Where E, I, Ω and L represent the Excitatory, Inhibitory, Input and Output neurons, e_1 , e_2 , i_1 and i_2 are the weights, ζ_E and ζ_i are the threshold values and $S(t)$ is the external input stimulus. By simulation in the Matlab program, the bifurcation diagram is shown in Figure 2.

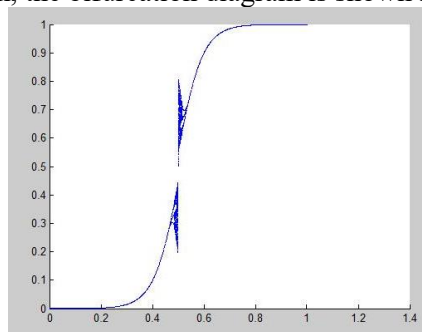


Figure 2. Bifurcation diagram of the Lee-oscillator

By applying the oscillator model as the activation function, we effectively convert our traditional neural network into a chaotic neural network. In this kind of transformation, during the training of our chaotic oscillatory neural network, the chaotic region provides different degree of “hysteresis” during network training, which provides the ability for the network to jump out of the local minimum and to resolve the “deadlock” and “over-training” problems which are commonly found in traditional neural networks.

Inside the hidden layer structure, in order to fully use the chaotic region and prevent the gradient problem to stuck our chaotic activation function, we need to limit the input distribution of the chaotic neurons within the range between 0.3 to 0.7 as much as possible, by combing with the training acceleration technique [7], we add the batch normalization layer between each chaotic oscillatory layer. In addition, considering our input contains lots of feature of the time series, which are 9-day time series price variables and 9-day eight nearest daily quantum price levels, in order to prevent the overfitting problem [8], we construct the dropout layer with 0.8 as the keep probability inside the hidden layers. The hidden layers architecture is given in Figure 3.

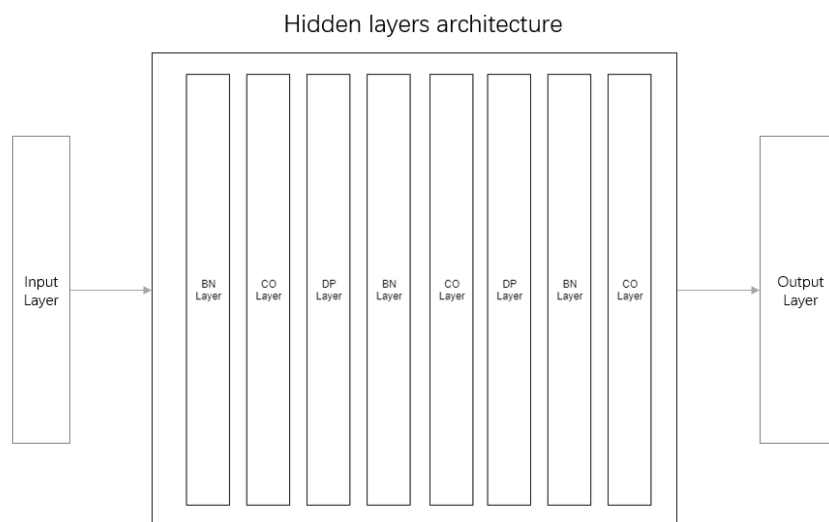


Figure 3. Architecture of the hidden layers in the HCONN.

Where the BN, CO, DP Layers represent the batch normalization layer, chaotic oscillatory layer and dropout layer respectively.

During system training, we use the ADAM optimizer with default learning rate 0.01 to minimize the cost function, which has the state-of-art performance in the deep neural network training [9]. The main reasons to adopt ADAM optimizer include: 1) ADAM optimizer could adjust its learning rate based on different parameters, which can be applied in several different models; 2) In our chaotic activation function, the calculation is much more complex than the traditional activation function. So, in order to ensure the effective usage of the neural network in the actual trading and forecasting, the time consuming of the training of the network should be minimize; 3) For the chaotic activation function, it could jump out of the local minimum because of its chaotic region and the stochastic gradient calculation. However, in order to prevent it from jumping out of the global minimum and ensure it could converge in the global minimum, we need to choose the optimizer which could adjust the learning rate with the process of convergence.

4. Experimental Tests

In this section, HCONN is evaluated in two major perspectives following the traditional deep learning model evaluation principle, which are 1) the accuracy of the prediction test which quantified by the mean square error. 2) the speed of convergence test quantified by the training epochs. In the model comparison, the HCONN will be compared with FFPBN and CONN (without QPLs).

4.1. Experimental data

All the experimental data are fetched from the MetaTrader 4 forex trading platform historical data centre. In total, 12 forex products are used: AUDCHF (2051), AUDCHF (2051), AUDUSD (2051), CADCHF (2051), EURAUD (2051), EURCHF (2051), EURGBP (2051), EURUSD (2047), GBPAUD (1531), GBPCAD (1531), GBPUSD (2047), USDCAD (2047), USDCHF (2051), where numbers in () shows the number of trading-day of the timeseries being used for system training and testing.

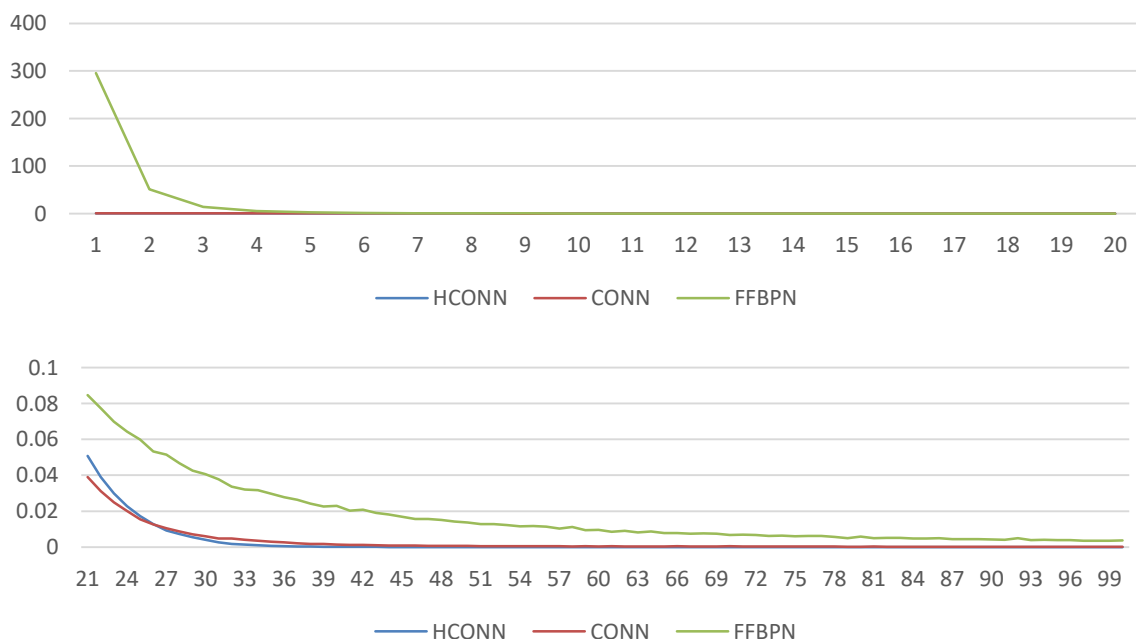
4.2. Experimental environment

For the QPL calculation, we conduct this process in the MetaTrader 4 platform by using the MQL4 programming language. For the data pre-processing, we conduct it in the Python 3.6, relying on the packages numpy and pandas. In the model construction, we develop the HCONN and its training comparison model CONN and FFBNP with Google tensorflow with version 1.31 which is a powerful framework for the machine learning research and development. In addition, for the accuracy comparison model including support vector machine regressor and random forest regressor, we construct it by using the sci-kit learn package. All the models including the HCONN and its comparison model are running in CPU cluster with Windows 10 operating system and Intel(R) Core(TM) i5-6300HQ CPU @ 2.30GHz, 2304 Mhz processor.

4.3. Comparison model

By following the two measurement methods which are 1) the accuracy of the prediction test which quantified by the mean square error. 2) the speed of convergence test quantified by the training epochs. In the model comparison, we evaluate and compare the three model: HCONN, CONN (without QPL) and FFBNP in both training set and testing set.

4.3.1 Convergence comparison. During the convergence comparison, we plot the mean square error versus the training iteration for three different models to compare the speed of convergence. Due to the huge numerical difference during the convergence process between the 3 models, we draw a point representing each 5 epochs, in this way we have 300 points in the training process graph. Then we separate the 1500 training epochs into 3 phases define as: 1) 1 epoch to 20 epoch; 2) 21 epoch to 100 epoch; 3) 101 epoch to 300 epoch, these three-period graphs are given in Figure 4.



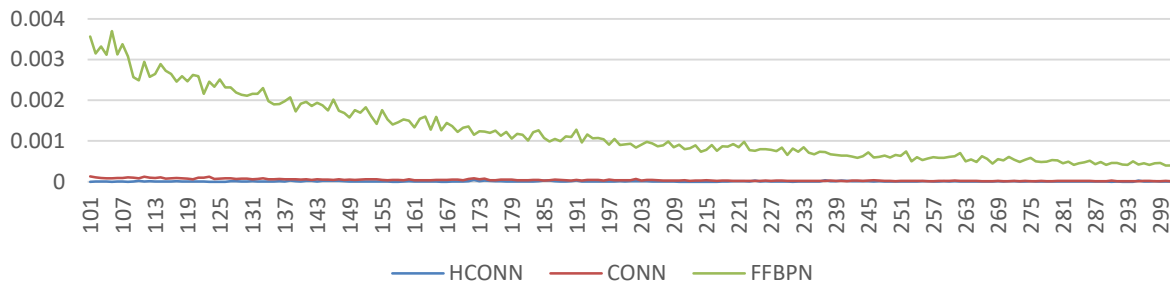


Figure 4. Average mean square error among 12 products versus the training iteration graph in three period: Top) 1 epoch to 20 epoch; Middle) 21 epoch to 100 epoch; Bottom) 101 epoch to 300 epoch.

From these three charts, it reveals that in the early phase, FFBPN almost has the highest mean square error. Also, both HCONN and CONN have fast convergence speed and better global minimum to convergence, one of the main reasons is that for the HCONN and CONN, the chaotic oscillatory activation function could enable the neural network to jump out of the local minimum during the training process. By combining with ADAM optimizer, its learning rate will be limited in an appropriate range during the training process. In conclusion, the combination of the oscillatory activation function with ADAM optimizer is one of the best training techniques combination in the financial prediction question case, the oscillator enable the model to have the ability to escape from local minimum without too much sacrifice in the accuracy by ADAM's mechanism.

In order to compare HCONN with CONN in detail, we plot the average mean of Mean-Square-Error (MSE) for the network training of the 12 forex products versus training epoch in Figure 5, in which we could see that the quantum price level will accelerate the speed to convergence to the global minimum. But for the local minimum result, the HCONN and CONN have no significant difference for the training set. The detail comparison we will discuss in the accuracy comparison and analysis.

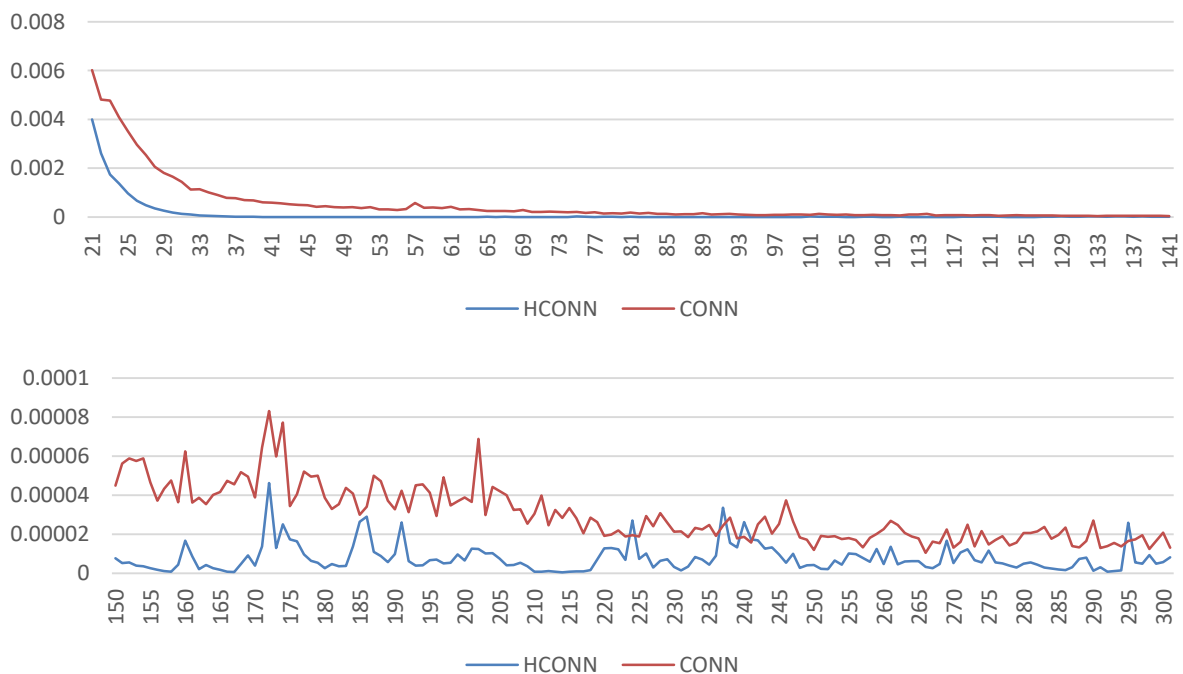


Figure 5. HCONN and CONN comparison: Average mean square error among 12 products versus the training iteration graph in two period: Top) 20 epoch to 150 epoch; Bottom) 150 epoch to 300 epoch.

4.3.2 Accuracy comparison. For each forex product, we show the mean square error of the prediction in the corresponding testing set for the three models (Table 1).

Table 1. Mean square error in each product for three models

	AUDCHF	AUDUSD	CADCHF	EURAUD	EURCHF	EURGBP
HCONN	2.36E-06	1.81E-06	5.58E-06	1.72E-06	2.17E-05	1.72E-06
CONN	0.00032897	0.00045184	1.87E-06	1.73E-06	1.76E-06	1.84E-06
FFBPN	3.22E-06	0.14337366	0.01719210	4.33E-03	1.19E-03	1.18E-05
	EURUSD	GBPAUD	GBPCAD	GBPUSD	USDCAD	USDCHF
HCONN	2.69E-06	7.83E-06	1.02E-05	1.75E-06	1.79E-06	3.65E-05
CONN	1.73E-06	2.53E-05	3.48E-05	4.200E-05	7.09E-05	0.00012943
FFBPN	4.52E-03	4.77E-03	0.09754947	0.01744503	0.00107840	1.67E-06



Figure 6. Normalized mean square comparison, each column means the proportion of three model's error in a certain product, the bottom two graphs are the absolute error comparison for three models.

In order to illustrate the error analysis among the three forecasting models, we plot the proportion of three model's error for each certain product in Figure 6 and plot the absolute error value comparison between HCONN and CONN in Figure 7. From the result of the normalized error analysis, we could see that for each product, most of the error is contributed by the FFBPN model and CONN model. By comparing the HCONN with two benchmarks model, we could see that for the HCONN, it contains the most powerful and stable ability in prediction. The accuracy difference between CONN and FFBPN shows that by applying the chaotic oscillatory activation, we could efficiently enable the neural network to fit the complex, chaotic and highly random time series patterns. In addition, the stability in prediction among different products show that by applying the chaotic oscillatory activation, the generalization ability of the neural will increase as well, which means the model could adapt the different product well in the market, which is important for reduction of consuming in the actual trading.

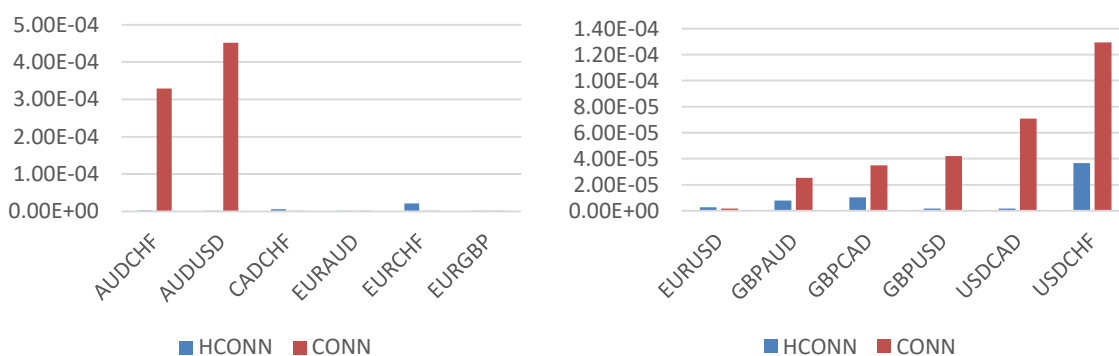


Figure 7. Absolute mean square error comparison between HCONN and CONN.

By comparing the MSE difference between CONN with HCONN as shown in Figure 7, it reveals that by applying the quantum price level as the feature extraction method the accuracy of the prediction can be improved significantly.

5. Conclusion

In this paper, we apply the Hybrid Chaotic Oscillatory Network (HCONN) which integrate chaotic neural network with the quantum finance theory to extract the intrinsic financial property of each financial product to tackle short term daily prices prediction problem. We take the past 9-day time series prices and quantum price levels as the input, to predict the next day highest and lowest price.

Specially, by integrating with Lee-oscillator as the chaotic activation function with ADAM optimizer, the model has the more powerful ability in the convergence to the global minimum and training acceleration. By comparing with the original chaotic oscillatory neural network, the quantum price level has the significant positive contribution to minimize the loss value by extracting the deep feature of the financial product.

Overall, based on the experiment result, we could see that the hybrid chaotic oscillatory neural network (HCONN) model could successfully extract the price movement pattern from the past chaotic, stochastic and complex data. Compared to other models, it has the significant ability in both the short term prediction accuracy but also the generalized prediction ability, which produces the reliable result in terms of trading analysis.

Acknowledgments

The author wants to thank for the MetaTrader 4 platform for the historical data providing. The authors also wish to thank Quantum Finance Forecast Center of UIC for the R&D supports and the provision of the channel and platform qffc.org for system testing and evaluation. This paper is supported by

Research Grant R201948 of Beijing Normal University-Hong Kong Baptist University United International College (UIC).

References

- [1] Bao W, Yue J and Rao Y 2017 A deep learning framework for financial time series using stacked autoencoders and long-short term memory *PloS One* **12**(7) pp. e0180944
- [2] Zhou Z 2016 *Machine Learning*, Tsinghua University Press, Beijing
- [3] Cucu GS, Ciotec AD and Neagoe VE 2018 Deep Convolutional Neural Networks versus Multilayer Perceptron for Financial Prediction *2018 International Conference on Communications (COMM)* pp 201-206
- [4] Lee RST 2019 *Quantum Finance - Intelligent Financial Forecast and Program Trading Systems*, Springer NATURE, Singapore (in printing).
- [5] Lee RST 2004 A Transient-chaotic Auto-associative Network (TCAN) based on Lee-oscillators *IEEE Trans. Neural Networks* **15**(5) 1228-1243
- [6] Lee RST 2019 Chaotic Type-2 Transient-Fuzzy Deep Neuro-Oscillatory Network for Worldwide Financial Prediction *IEEE Transactions on Fuzzy Systems* DOI:10.1109/TFUZZ.2019.2914642
- [7] Sergey I Christian S 2015 Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift *International Conference on International Conference on Machine Learning* 448 – 456
- [8] Srivastava N, Hinton G, Krizhevsky A, et al. 2014 Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, **15**(1), 1929-1958
- [9] Kingma D and Ba J 2015 Adam: A Method for Stochastic Optimization. *The 3rd International Conference for Learning Representations* San Diego. 1–13