A Mediated Multi-RNN Hybrid System for Prediction of Stock Prices

Ray R. Hashemi
Department of Computer Science
Georgia Southern University
Savannah, GA, USA
rayhashemi@gmail.com

Omid M. Ardakani
Department of Economics
Georgia Southern University
Savannah, GA, USA
oardakani@georgiasouthern.edu

Azita A Bahrami
IT Consultation
Savannah, GA, USA
Azita.G.Bahrami@gmail.com

Jeffrey A. Young
Department of Computer Science
Clemson University
Clemson, SC, USA
alanyoung7@gmail.com

Abstract-A multi-recurrent neural network (RNN) hybrid system made up of three RNNs is introduced to predict the stock prices for 10 different companies (five selected from the Dow Jones Industrial Average and five from the Standard and Poor's 500.) The daily historical data used to train and test the system are collected for the period of October 15, 2013 to March 5, 2019. For each company, the system provides two separate predictions of the daily stock price by using (1) historical stock prices and (2) historical trends along with the historical daily net changes in stock price. The two predictions are mediated to select one as the final output of the hybrid system. For each company, the accuracy of the system was tested for the prediction of the most recent 98 consecutive days using the forecast accuracy measure of the Mean Squared Error (MSR). The results revealed that for every company the difference between the predicted and actual stock price is not statistically different from zero, which is the ideal (error-free) forecast.

Keywords: Stock Prediction, Recurrent Neural Networks, Mediated Prediction Systems, Hybrid Prediction Systems, Multi-Recurrent Neural Networks systems

1. Introduction

The stock market is considered as a chaotic system consisting of two components of *deterministic* (predictable) and *random* (unpredictable) [1][2]. The rational behavior of investors that are predicted by *fundamental analysis* (i.e. use of macro/financial data) is considered as the deterministic component, whereas investors' psychological behavior captured by the *technical analysis* (i.e. historical price patterns—technical indicators) is considered as the random component. The former can be characterized by statistical techniques, while the later cannot be fully captured by such methods. Thus, using only statistical techniques will make it difficult to predict stock prices. A neural network has the ability to learn both deterministic and random features of the chaotic stock prices and provides a powerful forecasting tool [3][4][5].

In addition, stock prices have two intrinsic traits that add to the complexity of their accurate predictions. The first trait is that the stock prices are time-series data and tomorrow's stock price of the company is influenced by today's value. Conventional neural networks do not provide for this intrinsic trait. However, a Recurrent Neural Network (RNN) can lend itself easily to provide for such a trait by feeding its output at time t as part of its input at time t+1[6][7].

The second intrinsic trait is the lack of direct measurable attributes influencing the stock price. To explain it further, an event such as a car accident, for example, has several direct measureable attributes at the time of the accident such as speed of the vehicle(s), age of the driver(s), the amount of consumed alcohol by the driver(s), etc. Such attributes make the prediction of a car accident, in general, much easier. A stock price, however, does not have such direct measurable attributes (except for the stock price itself.) For example, the effect of a (political and/or non-political) event cannot be expressed as a crisp value (i.e. measured) at the time that the event took place. To overcome this intrinsic trait, researchers try to use auxiliary data (i.e. other financial data) for having a better prediction. Such auxiliary data are the foundation of the hybrid predictive systems that usually use a mixture of linear and nonlinear methods [8][9][10].

We hypothesize that the use of two technical indicators of the *daily stock price trend* (T) and the actual *daily net change in stock price* (V) as auxiliary data will substantially improve the prediction of stock price. Both of these indicators are also extracted from the historical data of the stock prices. For the ease of reference in the future, we refer to this hypothesis as the *T-V Hypothesis*.

The goal of this research has four sub-goals (a) Introducing and building a hybrid prediction system that predicts daily stock price for a company by using historical stock prices and predicts another daily stock price for the same company by using historical trends in stock prices along with historical daily net changes in stock prices, and then presenting a mediator that delivers the final predicted stock price for the company, (b) Checking the performance of the system for ten fortune 500 companies, (c) Validating the stock price predictions by using (MSE), and (d) Showing whether the above hypothesis for use of T and V is true.

The rest of the paper is organized as follows. The Previous Works are the subject of Section 2. The Methodology is presented in Section 3. The Empirical Results are covered in Section 4. The Conclusions and Future Research are subjects of Section 5.

2. Previous Works

The use of RNN for prediction of the stock prices have been reported in the literature [11] [12][13]. In general, such strand of the literature may be divided into pure and *hybrid* RNN systems. The first group uses an RNN to predict the stock price. Often a pre-processing step is involved to make data palatable to the RNN. The second group either feeds the outcome of the RNN into another black box (another system) for improving the outcome or gets the output from a black box and feeds it into an RNN to improve the prediction. The black box is usually a system that uses a methodology other than RNN.

We introduce a hybrid system that is made up of multiple RNNs and it is distinct from the other hybrid system for using historical stock prices along with two technical indicators of T and V. Practically, our hybrid system predicts two stock prices for each company and for each day by using: (i) The historical stock prices and (ii) The historical technical indicators of T and V and then try to mediate between the two predictions for delivering the final stock price forecast. To the best of our knowledge such methodology has not been reported in the literature.

3. Methodology

We introduce our prediction system that has four components as shown in Figure 1. The first three components are three recurrent neural networks (RNN_1, RNN_2, and RNN_3) in charge of predicting the daily stock price (Q), the daily trend (T) in stock price, and the amount of daily change in the actual stock price (V) for a given company. The fourth component (Mediator) takes the Q, T, and V and makes the final predicted stock price as the output of the system (P). The details of the four components are the subjects of the following four subsections.

A. Prediction of the daily stock price (Q)

A recurrent neural network (RNN) is capable of learning the patterns from the time series data. This means that there is a feed-back loop to serve the output of one time step as part of the input for producing the output for the next time step. Our RNN for the first component (RNN_1) is a three-layer neural network as shown in Figure 2. The input layer is composed of two different sets of nodes: *input* set and *context* set. The first set has only one node (X) that receives the stock price for day d_{i-1} and delivers the prediction of stock price for day d_i. The second set has several nodes $C=\{C_1, \ldots, C_n\}$ and they receive the output of the hidden layer nodes for prediction of stock price at the previous time step (i.e., prediction of stock price for day d_{i-1}). However, the output of node H_i in the hidden layer serves as a feedback only to node C_i of the context set (shown in broken arrows). Therefore, for our RNN_1 the number of context nodes and the number of nodes in the next layer (hidden layer) are the same. Since the input to the context nodes are only the outcome of the nodes in the hidden layer and such input data initially do not exist, we use a set of dummy small value as initial input to the context nodes

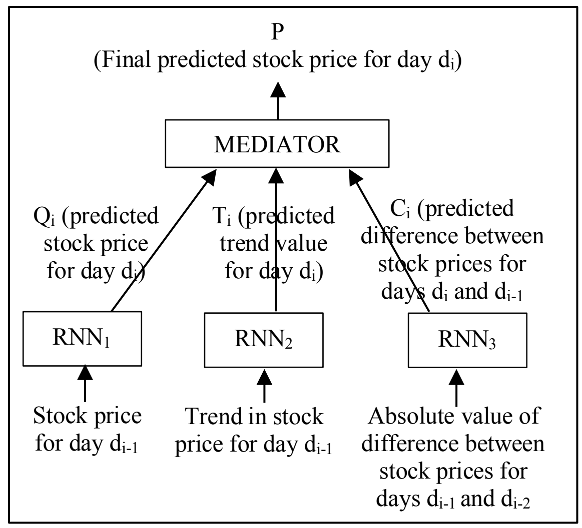

Figure 1: The Multi-RNN Hybrid system

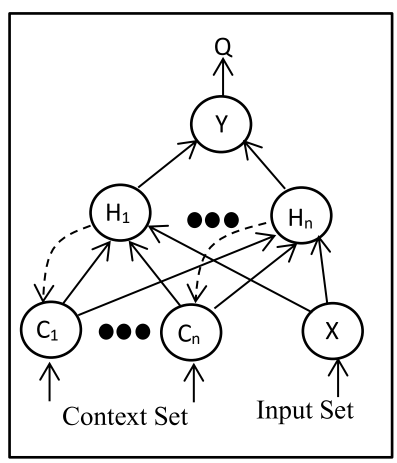

Figure 2: The RNN Architecture

The hidden layer accepts input from both sets of nodes in the input layer and produces output that goes to the Context nodes (as a feedback) and also to the output layer node to deliver the predicted intermediate stock price (Q). The output layer has only one node.

There are two weight matrices of W_1 and W_2. W_1 is the weight matrix for connections between nodes of input layer and hidden layer and it has (n+1) rows (the total number of nodes in the input set and the context set) and n columns (the number of nodes in the hidden layer). W_2 is the weight matrix for connections between nodes of hidden and output layers and it has n rows and one column. The weight matrices are created randomly and then normalized. The difference between the predicted and actual stock price (Q) is back propagated to change the weight matrices of W_2 and W_1, respectively.

The stock price for day d_{i-1} that is fed to the node in the input set goes through a two-phase scaling process. During the first phase, the magnitude of the stock price is scaled down such that the magnitude is within the range of (0-100). Therefore, the stock price is divided by 10^u , where the value of u is decided using Table 1.

Table 1: Value of u for the first phase scaling

Range of Stock price	Value of u
(0-100)	0
[100 - 1000)	1
[1000 - 10000)	2
[10000 - 100000)	3

One may ask why such scaling is necessary. Our empirical results supported the fact that the RNN1 learns the periodic patterns with a higher degree of precision enforcing this scaling.

During the second phase, the outcome of the first phase is divided by a constant value of B, where $275 \leq B \leq 300$. The purpose of the second phase is to provide a solution to an observed behavior of the RNN1. To explain this behavior further, we noticed that the plots of the predicted stock prices and the actual stock prices are extremely similar. However, the magnitude of the amplitudes in the predicted stock prices plot are much higher than the magnitude of the amplitudes in the actual stock prices plot as shown in Figure 3. The second phase of scaling brings the two magnitudes much closer to each other. The value of B was determined by trial and error.

The Rectified Linear Units (ReLu) is used as the activation function, which its behavior for input value of α is shown in formula 1.

$$\begin{cases} f(\alpha) = 0 & \alpha \leq 0 \\ f(\alpha) = \alpha & \alpha > 0 \end{cases} \quad (1)$$

Using ReLue is motivated by the fact that we do not have negative and zero stock prices and the RNN_1 performs much better.

B. Prediction of Trends in Stock prices

The RNN for the second component (RNN_2) has the same architecture as the RNN_1 and it predicts the daily trends in stock prices. Therefore, the input to the RNN_2 is the trend for the day d_{i-1} and the output is the prediction of the trend for d_i. The trend value may be up (T_i = +1), down (T_i = -1) or no changes (T_i = 0). Since the ReLu maps all the negative and zero input values to zero it is a legitimate concern about use of ReLu, as activation function, for RNN_2. To address this concern, the trend values were mapped to a set of non-negative integers [0, 2] by adding 2 to each trend value in the input layer. The new trend value was fed through the RNN_2 using the ReLu function. The output from the output node was lastly re-scaled by subtracting 2.

In contrast with RNN_1, there is no need for the two-phase scaling of the input data for RNN_2.

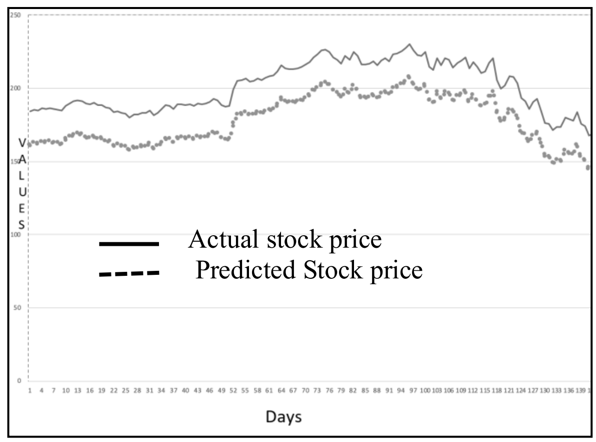

Figure 3: Plots of the predicted and the actual values for Apple's stock.

C. Prediction of the daily change in the actual stock price

The RNN for the third component (RNN_3) has the same architecture and uses the same activation function, ReLu, as the RNN_1. The input to the RNN_3 is the absolute value of the difference between the stock price for days d_{i-1} and d_{i-2} (|V|) that is used to predict the difference between the stock price for days d_{i-1} and d_i and the output is the prediction of the absolute value of the difference between days d_{i+1} and d_i. It is clear that the actual stock price for today is unknown. One may ask why we are interested in predicting |V| and not V itself. For two reasons: (i) The concern about having a negative V value, in reference to the use of ReLu, no longer exist and (ii) trend prediction delivered by RNN_2 can easily compensate for the sign of V value.

In contrast with RNN1, there is no need for the two-phase scaling of the input data for RNN3.

D. The Mediator

For the day d_i, tomorrow, the above mentioned three components deliver a predicted stock price (Q_i), a predicted trend in stock price (T_i), and a predicted (absolute value) of the difference between stock prices for day d_{i-1} and day d_i (D_i). We use Formula (2) to make another prediction of tomorrow's stock price and it is named *calculated stock price* (K_i):

$$K_i = T_i * D_i + (\text{Stock price for day } d_{i-1}) \qquad (2)$$

The value of $\delta_i = |Q_i - K_i|$ and a threshold value of T_v used by the Mediator to decide on the final prediction (P_i) of the stock price for day d_i using formula (3):

$$\left.\begin{array}{ll} P_i = K_i & \text{if } (\delta_i < T_v) \\ P_i = Q_i & Otherwise \end{array}\right] \qquad (3)$$

The threshold value, T_v, may be different from one company to the next and it can be calculated. The calculation process is discussed in the following section.

4. Empirical Results

Explanation of the historical stock prices, details of specifics in implementation of RNNs, and our findings are the subjects of the following three subsections.

A. Data

We have ten files of the daily stock prices for ten fortune 500 companies. Five of these companies selected from the Dow Jones Industrial Average and five from the Standard and Poor's 500. The list of companies is shown in Table 1.

Table 1: The top companies in the Dow and Standard and Poor's

Ticker	**Company**	**Index**
AAPL	Apple Inc.	Dow Jones
AMZN	Amazon Inc.	S&P 500
FB	Facebook Inc.	S&P 500
GOOG	Alphabet Inc. Class C	S&P 500
INTC	Intel Corp.	Dow Jones
JNJ	Johnson & Johnson	S&P 500
MSFT	Microsoft Corp.	S&P 500
PFE	Pfizer Inc.	Dow Jones
PG	Procter & Gamble Co.	Dow Jones
XOM	Exxon Mobil Corp.	Dow Jones

Each file contains the historical daily stock prices for the period of 15 October 2013 to 5 March 2019. The number of records in each file is 1354. The daily stock prices obtained from Wharton Research Data Services (WRDS) [14]. WRDS provides access to CRSP, NYSE, COMPUSTAT and other data sources.

The cheapest stock in the sample belongs to Pfizer Inc. (the average of $31.2) and the most expensive one is Amazon Inc. (the average of $831). Based on the coefficient of variation (CV), Exxon Mobil Corp. has the lowest volatility (i.e. lowest uncertainty) in our sample while Amazon.com has the highest. Overall, Johnson & Johnson, Pfizer Inc., Proctor & Gamble Co., and Exxon Mobil Corp. are considered as low volatility stocks.

B. Details of Implementation's Specifics

For the first two companies of Table 1, the Learning rate (η), number of nodes in the hidden layer (N) and the tolerance level (i.e., the maximum difference between the desired output and calculated output, denoted by τ) used in RNN1, RNN2, and RNN3 may be different and they are shown in Table 2. For the remaining companies in Table 1, the values for η, N, and τ remains the same and they are shown in Table 3.

Table 2: RNN1, RNN2, and RNN3 implementation details for the first two companies in Table 1

RNN	Ticker	Learning Rate (η)	# of nodes in Hidden Layer (N)	Tolerance Level (τ)
RNN1	AAPL	0.85	3	0.03
	AMZN	0.825	5	0.04
RNN2	AAPL	0.835	3	0.03
	AMZN	0.815	4	0.04
RNN3	AAPL	0.835	3	0.03
	AMZN	0.815	4	0.04

Table 3: RNN1, RNN2, and RNN3 implementation details for the last eight companies in Table 1

Ticker	Learning Rate (η)	# of nodes in Hidden Layer (N)	Tolerance Level (τ)
FB	0.85	3	0.02
GOOG	0.875	7	0.05
INTC	0.85	3	0.05
JNJ	0.85	3	0.05
MSFT	0.85	3	0.05
PFE	0.85	3	0.05
PG	0.85	3	0.05
XOM	0.85	3	0.05

C. Findings

As we mentioned before, historical stock prices are time series data and tomorrow's stock price of the company is influenced by today's stock price. In other words, we cannot set aside a set of daily stock prices as a test set and use them to evaluate the prediction power of the trained proposed system because today's stock price must participate in prediction of tomorrow's stock price. As a result, we looked at the prediction of the daily stock prices of each company delivered by our hybrid system for the most recent 98 trading days.

We determined the threshold value, $T_{\cdot}$, used by the Mediator component of the system for each company by observations. The list of companies' threshold values along with average daily error (in dollar) and percentage of average daily error in reference to the average stock price for 98 days are shown in Table 4.

Table 4: Threshold values for the 10 companies along with the average daily error (in dollars) and percentage of average daily error in reference to the average daily stock prices during the 98 days

Ticker	Threshold Value	Avg. Daily error ($)	% of Avg. Daily Error in Reference to Avg. daily stock price
AAPL	$2.5 \leq T \leq 3.8$	0.12	0.07
AMZN	$17 \leq T \leq 17.7$	1.21	0.07
FB	$T = 2.1$	0.11	0.07
GOOG	$16 \leq T \leq 19$	0.93	0.09
INTC	$2 \leq T$	0.08	0.16
JNJ	$1.7 \leq T$	0.09	0.06
MSFT	$0.9 \leq T \leq 2.6$	0.06	0.05
PFE	$0.7 \leq T$	0.03	0.06
PG	$1.08 \leq T \leq 1.24$	0.06	0.07
XOM	$1.1 \leq T$	0.05	0.06

Table 5: The mean squared errors along with the t-stats

Ticker	MSE	t-stat
AAPL	0.18	2.44
AMZN	17.58	2.42
FB	0.19	1.78
GOOG	9.93	2.25
INTC	0.04	2.98
JNJ	0.13	1.70
MSFT	0.04	1.97
PFE	0.01	2.02
PG	0.03	2.58
XOM	0.02	2.30

We used MSE to test the accuracy of our findings for each company. The MSE is a forecast accuracy measure that is the average squared difference between the predicted values and the actual value. The MSE for the ideal (error-free) forecast is zero. We test to find whether the MSE of our hybrid system is statistically equal to zero. To do so, we use the one sample t-test in which the null hypothesis is MSE = 0. The MSEs along with the t-stats are shown in Table 5. For nine companies we fail to reject the null hypothesis at the one percent significance level. These findings suggest that the MSE from our hybrid system is not statistically different from zero which is of the ideal forecast. The same result holds true for INTC at the 0.1 percent significance level.

6. Conclusion and Future Research

Four highly influential forces behind the stock market data shapes the structure of the hybrid system covered in this research effort. These four forces are: (a) The rational behavior of investors that make the deterministic feature of stock market, (b) The investors' psychological behavior that makes the random feature of the stock market, (c) The time-series nature of stock market data, and (d) The lack of ability to express the effect of an event (political and/or non-political) on stock market in crisp values.

The accuracy of our findings was tested by using MSE technique for ten companies and for the most recent 98 trading days. The results revealed that for every company the difference between the predicted and actual stock price is not statistically different from zero that is the ideal forecast. The percentage of average daily error for ITNC Company is equal to 0.16% of the average stock prices for 98 days. For the rest of companies such percentages are less than or equal to 0.09%.

Reader needs to be reminded that we hypothesized previously (T-V hypothesis) that the use of two technical indicators of the daily stock price trend and the actual daily net change in stock price as auxiliary data will substantially improve the prediction of stock price. Our findings revealed that the T-V hypothesis holds.

As future research, the use of fuzzy logic for expressing the effect of an event (political and/or non-political) on stock market is in progress.

References

[1] McKenzie M., "Chaotic behavior in national stock market indices: New evidence from the close returns test", Global Finance Journal, 2001, (12)1, 35-53.

[2] French K. R., Schwert G., and Stambaugh R. F, "Expected Stock Returns and Volatility", Journal of Financial Economics, 1987, 19 (1), 3-29.

[3] Bergerson K. and Wunsch, D. C., "A commodity trading model based on a neural network-expert system hybrid", the IJCNN--Seattle International Joint Conference on Neural Networks, IEEE, 1991, Vol. i, pp. 289–293.

[4] Yoon Y. and Swales G., "Predicting Stock Price Performance: A Neural Network Approach", The International Conference on System Sciences, 1991, pp. 1–7.

[5] Hushani, P., "Using Autoregressive Modelling and Machine Learning for Stock Market Prediction and Trading", the Third International Conference on Information and Communication Technology, 2019, Vol. 797, pp. 767–774.

[6] Connor J.T., Martin R.D., Atlas L.E., "Recurrent Neural Networks and Robust Time Series Prediction", IEEE Transactions on Neural Networks, 1994, 5(2), 240-254.

[7] Miljnovic M., "Comparative Analysis of Recurrent and Finite Impulse Response Neural Networks in Time Series Prediction", Indian Journal of Computer Science and Engineering, 2012, 3(1) 180-191.

[8] Kim K.J. and Han J., "Genetic Algorithms Approach to Feature Discretization in Artificial Neural Networks for the Prediction of Stock Price Index", Expert Systems with Applications, 2000, Vol. 19, 125–132.

[9] Leigh W., Paz, M., and Purvis R., "An Analysis of a Hybrid Neural Network and Pattern Recognition Technique for Predicting Short-Term Increases in the NYSE Composite Index". Omega, 2002, 30(2), 69–76.

[10] Cheng C.-H., Chen T.-L., and Wei L.-Y., "A Hybrid Model Based on Rough Sets Theory and Genetic Algorithms for Stock Price Forecasting", Journal of Information Sciences, Elsevier Inc. 2010, pp: 1610-1629.

[11] Rather A. M., Agarwal A., and Sastry V.N., "Recurrent Neural Network and a Hybrid Model for Prediction of Stock Returns", Expert Systems with Applications, 2015, 42(6), 3234-3241.

[12] Bhowmick A., Rahman A., and Rahman R.M., "Performance Analysis of Different Recurrent Neural Network Architectures and Classical Statistical Model for Financial Forecasting: A Case Study on Dhaka Stock Exchange", Proceedings of the 8th Computer Science On-Line Conference (CSOC 2019): Artificial Intelligence Methods in Intelligent Algorithms, 2019, Vol. 2, 277-286,

[13] Bekiros S.D. and Georgoutsos D. A., "Direction-of-Change Forecasting using a Volatility Based Recurrent Neural Network", 2018, 27(5), pp. 407-417.

[14] Wharton Research Data Services (WRDS) https://wrds-www.wharton.upenn.edu, Accessed on July 2019.