Temporal analytics for forecasting share prices

Haim Osadon
Shofet Abraham 45
Beersheba, Isreal
haimos@bgu.ac.il

Yechiav Yitzchak
Sirikin 2
Tel-Aviv, Israel
Yechiavil@gmail.com

ABSTRACT

Prediction of the stock market has always been intriguing research question for its complexity and the challenge its present. One method that is most commonly used in the process of identification is the time-series analysis. In this paper we examined ARIMA, Holt-Winters, Feed-Forward ANN and RNN method in stock price prediction in 5 scenarios evaluating the methods in various aspects. While all method struggled in this complex task one method - RNN - showed more promising results in the domain of stock price prediction.

Keywords

ARIMA, Holt Winters, ANN, RNN, Stock, Prediction

1. INTRODUCTION

The biggest benefit from the prediction of the stock market will be implementation of automated system that in every moment will be able to accurately predict shares, for that reason, a lot of effort by scientists and investors was put in to research and develop an automated prediction system. In this paper we present two concepts which are based on analysis of the past values of the prices of the stock market. Moreover, both of these concepts are opposing the Efficient Market Hypothesis (EMH), which will be explained later in this paper. The first model is based on the ARIMA and Holt Winters model, whereas the latter is based on the concept of artificial neural networks (ANN,RNN).

2. BACKGROUND AND RELATED WORK

Making precise predictions of stocks are challenging because stock data is noisy and non- stationary (Abu-Mostafa Atiya, 1996)[1], and Although the Random Walk Theory claims that the change of stock price are independent of its history and we cannot obtain any indication to predict future price trends from stock price history data there was a lot of research done which proves low accuracy in stock price prediction.

The financial time series models expressed by financial theories have been the basis for forecasting a series of data in the twentieth century. Yet, these theories are not directly applicable to predict the market values which have external impact. The development of multi-layer concept allowed ANN (Artificial Neural Networks) to be chosen as a prediction tool besides other methods. Various models have been used by researchers to forecast market value series by using ANN.

[2]CONG et al. (2016) suggested ARIMA model for time series forecasting of the fluctuation of CSI 300 index futures, they found that price fluctuation prediction of CSI 300 index futures is feasible and simulation results were very close to the market real data. CONG et All (2016) concluded that prediction by ARIMA model was more accurate for short term prediction than a long term prediction. Although simulation experiment of using the model buy and sell order was profitable they did not incorporated the transaction and commission costs in the simulation trading.

[3]Georgi Kostov, Sasho Nedelkoski and Goran Stojanovski (2016) reviewed a paper about time series forecasting on Apple stock. In this review they use ARIMA model on the one hand, and convolutional neural networks on the other hand for short term prediction. With both models they shows that we get financial gain and from the result it is clear that the system that based on the convolutional neural networks is more accurate than the system based on the ARIMA model. More so, the concept could be significantly improved - one particular idea for improvement of the algorithm efficiency is combining the CNN with text analysis of relevant journals/portal as an additional input.

The earliest stock market prediction model based on ANN was implemented by [4]White He used Feed Forward Neural Networks (FFNNs) to decode previously undetected regularities in the asset price movements such as fluctuations of common stock prices

Comparison of various types of feedforward neural networks (ANN) utilizing different type of backpropagation (BPN); regression neural network (GRNN), the class-sensitive neural network (CSNN), and the conjugate gradient trained network (CGNN) concluded that CSNN is the best preforming BPN in both predication and trend estimation [5](CHEN 1994)

Inputting not only the raw data but also additional indicators (rate estimator, momentum, moving average) concluded with good indicators using only previous 14 days of trade information researchers proved to achieve best results [5](CHEN 1994)

[6]Hill et al. (1994) suggest that ANNs are likely to work best for high frequency financial data and [7]Balkin and Ord (2000) also stress the importance of a long time series of observations to insure optimal results from training neural networks.

Unlike forward networks, the recurrent neural networks use feedback connections to model spatial as well as temporal dependencies between input and output series to make the initial states and the past states of the neurons involved in a series of processing.

[10]Kim, Han, and Chandler (1998) used a recurrent Elman neural network to predict the price of stocks in the Japanese stock exchange their proposed model yields higher profit than other comparable models and buy-and-hold strategy.

3. DEFINITIONS AND PRELIMINARIES

In this subsection we present the known definitions for the different methods.

ARIMA. The most prominent methods in financial forecasting. ARIMA models have shown efficient capability to generate short-term forecasts. It constantly outperformed complex structural models in short-term prediction. In ARIMA model, the future value of a variable is a linear combination of past values and past errors, expressed as follows:

$$\Delta y_t = S_0 + S_1 \Delta y_{t-1} + \ldots + S_p \Delta y_{t-p} + \epsilon_t + \phi_1 \epsilon_{t-1} + \ldots + \phi_p \epsilon_{t-p} \ ,$$

$$t = 1, 2 ... T$$

Where y_t is the actual value and Δy_t is represent differences of order d, ϵ_t is the random error at t, S_i and ϕ_i are the coefficients, p and q are integers that are often referred to as autoregressive and moving average, respectively.

The steps in building ARIMA predictive model consist of model identification and parameter estimation we use tow methods AIC and BIC to chosen the best model.
(4) AIC = - · 2 Ln (L)+2 k
(5) BIC = - · 2 Ln (L)+k Ln (n)
In formula (4) and (5) where, L is the maximum likelihood function, N is the sample size, and k is the number of parameters to be estimated in the model. With the Information criteria, we prefer to choose the smallest information criteria model.

Holt-Winters. The Holt-Winters Smoothing Algorithm uses weighted historical trending to predict the future values of an account. It is more accurate for accounts that tend to trend in one direction over time. The modified version of this algorithm looks at the financial data from past year and determines a value to place on the trend itself. Holt-Winters Model is a double exponential smoothing method that is appropriate for series with a linear trend and no seasonal variations. It is an extension of simple exponential smoothing method that is originally designed for time series with no trend nor seasonal patterns.
Basically, the Holt-Winters model contains two vital components; an exponentially the Holt-Winters model contains two vital components; an exponentially smoothing constant (E, α) and a trend component (T, β).The forecasted series is computed using the formula below:

$$F_{t+k} = E_t + kT_t$$

Here, E and T represent the intercept and slope respectively, and are computed recursively as follows:

$$E_t = \alpha Y_t + (1 - \alpha)(E_{t-1} + T_{t-1})$$

$$T_t = (E_t - E_{t-1})\, \beta + (1 - \beta) T_{t-1}$$

The symbols in the model are defined as follows:
$F_(t + k)$ = forecast value k periods from t
$Y_(t - 1)$ = actual value for previous period t - 1
$E_(t - 1)$ = estimated value for previous period t - 1
$T_(t - 1)$ = trend value for previous period t - 1
α = smoothing constant for estimates
β = smoothing constant for trend
k = number of periods
The coefficients α and β are smoothing factors, each coming from the closed interval [0,1]. When α and β have been selected, the smoothing equations are then employed to update the estimates of the constant and trend components. The updated estimates are used to compute forecasts of future time series values.

Feed-forward ANN model. Artificial Neural Network also known as multilayer perceptron (MLP) is a flexible method which structured from layers of perceptrons that can be used for prediction, the network can be structured from any number of layers where each layer holds numerous number of perceptrons. This algorithm consists of 2 phases, the training phase where input is given and based of the delta from the output to the actual result the network recalculates its inner weights to achieve the minimum loss function.

It is called a feedforward neural network since information is carried from the input node to the output node in a forward only motion. And the output of each layer is input of the following layer in the network. There is a big verity of parameters of neural network, in the scoop of this paper we will address the following items:

- The network structure: the layers and nodes of the layer
- The number of input nodes: we will refer to it as "look back" the number of inputs of the data given to each node
- Batch size: the number of samples given before preforming the back prorogation
- Epoch: the number of repetitions of the providing the complete training data for back-propagation

RNN. Recurrent neural network is a class of ANN which a nodes are connected to themselves by a directed cycle, there are number of RNN in this paper we will focus on neural network which utilize the long short-term memory (LSTM) node. A LSTM node is a node which is connected to itself by passing through “forget gate” a gate which determines how much the previous value of the node is influencing the current status of the node. This ability of holding the current and the pervious values of a node is the key factor of the reason why LSTM neural network proved to be successful in a temporal input data such as time series or speech recognition

Like Feed-forward ANN RNN has a big verity of parameters, we will tune the same set of parameters as ANN

Buy&Sell Algorithm. This algorithm set to test the trend prediction ability of each method, wrapping each method with a buy, sell and hold actions. This is implemented by adding a threshold g that will indicate if the predicted change is significant enough to Take action or to hold. The algorithm finds the trend and makes the decision as described:

F_t=predict(t)-predict(t-1)>0→buy in time (t-1)
F_t=predict(t)-predict(t-1)<0→sell in time (t-1)
F_t=predict(t)-predict(t-1)=0→hold

4. EVALUATION

4.1 Research Questions

The goal of this paper is to compare the performance of each method in predicting share prices, given the different nature of each model we would like to examine the performance of each model in the following scenarios.

Prediction time: As the database is given in a daily value, the ability to predict the next day value is as important as predicting the price for a future value, in this section we will examine the prediction for 1,3 and 5 days from the end of the input database. E.g prediction for time t+3 we will provide the algorithm all the dataset available from start time to time T to predict T+3 value of the stock price

ETF vs Stock:. Exchange-traded fund (ETF) is an investment fund traded on stock exchanges, much like stocks. In this paper we will compare the ability of each model to predict ETF price that are invested in the indexes of a stock exchange (e.g the DAX is the index of the 30 biggest companies in the German stock exchange) as ETF is comprised of multi stock prices we would like to evaluate the algorithms performance on a less noisy stock prices compare to a single stock price which is influenced heavily by the single company performance and reports.

Combination of multiply Stocks prices. : The basic assumptions is that stock prices of companies in same sector is influenced by a common factors like legislation the industry factors (e.g. a storm will influence companies in the insurance sector possibly in a similar way).
As Holt winter and ARIMA are auto regression algorithms in this sections we will examine the RNN performance of predicting a T+1 stock prices of a single stock by inputting 2 stock prices of same sector companies.

Buy&Sell Algorithm. Designed to measure the ability of each method to predict the trend of the stock price, this method simulates an investor and will suggest a *buy* and *sell* and *hold* actions.

4.2 Dataset

The financial data used in this research is a daily closing price of 4 stocks and 3 index world markets (ETF).The entire data download directly from yahoo.finance. The 3 index world markets are DAX, NASDAQ and SAP500.And the 4 stocks Apple, Target, Google and Walmart. The entire data set covers the period from 01th July 2015 to 10th July2016.There are 253 days of observations. The data set is divided into two periods: the first data period is from 01st July 2015 to 10st April 2016 (195 days of observations) while the second period is from 11st April 2016 to 10st July 2016 (60 days of observations). The first period, assigned to in-sample estimation, is used to determine the specifications of the models and to estimate their parameters. The second period, assigned to test the models built.

4.3 Evaluation Metrics

In evaluating the algorithms performance the MAPE evolution was elected, Mean Absolute Percentage Error is a measure of prediction accuracy of a forecasting given by the following formula:
$M = \frac{100}{N} \sum_{t=1}^{n} \left| \frac{A_t - F_t}{A_t} \right|$
Where A_t is the actual value and F_t is the forecast value. Since stock actual price is a non-determining factor in this domain and investors are looking for gain in percentage the MAPE is a measure which evaluate the percentage in the difference between actual and predicted value is chosen. Furthermore since we would like to evaluate performance for number of stock prices put together this measure allows incorporating number of stocks in evaluating the method. the formula for multi stock prices evaluation will be given by:

$$M = \sum_{i=1}^{n} \frac{M_i}{N}$$

Where M_i is the MAPE of a prediction of Stock i And N is the number of stocks This measurement will be used for the following sections: Prediction time, ETF Vs Stock, Combination of multiply Stocks prices.
for *Buy&Sell* algorithm, the Hit rate will be used to evaluate the methods on 55 observations.

4.4 Experimental Plan

ARIMA. The ARIMA model is implemented in the program package R, as it is very suitable method for working with time series analysis, as it has functions for autocorrelation, partial autocorrelation, tests for stationary. Consists of the following steps:

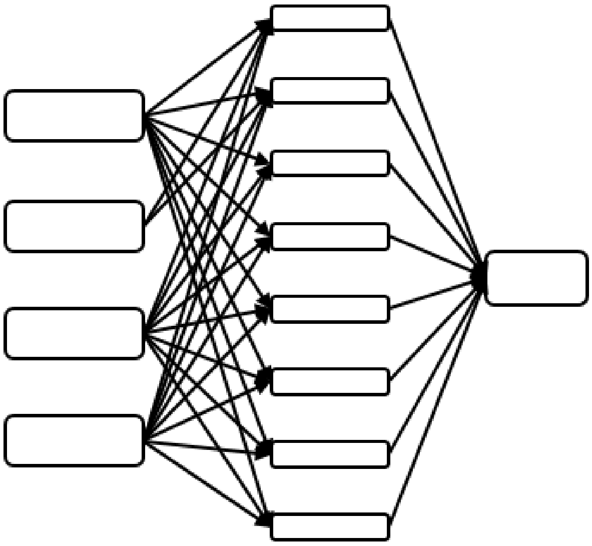

Figure 1: Feed-Forward ANN

1. Read the data
2. Remove all bug close data
3. Divide into 75% train data and %25 test
4. Use the training data to run a loop that change the ARIMA model parameters, find the best model with the lowest AIC and BIC
5. Use the parameters to design ARIMA model and to calculate a prediction sequence. Every time we go forward with the data and calculate the desired forecast
6. Repeat step 5 until end of test dataset

Holt-Winters. As ARIMA this method implemented in R, steps same to ARIMA procedure.

The ANN and RNN method implemented in python using the **Keras** package using a **Theano** back end engine, the training was done on the training set of each separate stocks. for both methods input was normalized to range of (0-1) and prediction was reversed to original values the loss function set to RMSE.

ANN. : Constructed of one input layer with number of "look back" nodes, hidden layer with 8 perceptrons with additional output layer Figure 1.

RNN. : Constructed of one input layer with number of "look back" nodes, hidden layer with 4 LSTM nodes with additional output layer Figure 2.

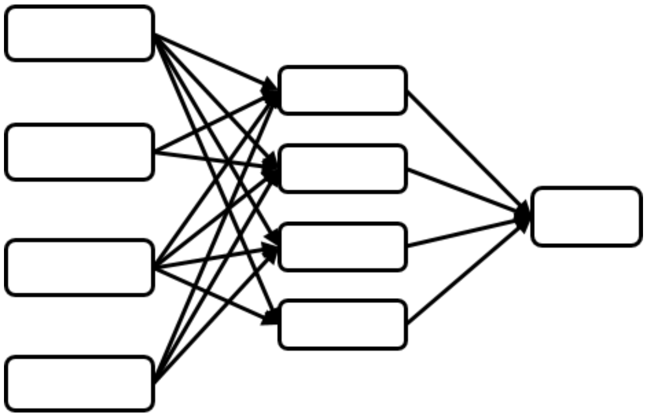

Figure 2: Recurrent Neural Network

Merge RNN. : The "Merge RNN" neural network used for 2 stock price input for prediction of the primary stock, this neural network is combination of the original **RNN** network as a left and right branch with a single layer of output. in this method the same parameter found in RNN parameter optimization section were used due to similarity of the 2 networks.

5. RESULTS

5.1 Parameter tunning results

ARIMA, and Holt-Winter tunning was done using the auto-arima and auto Holt-winter which finds the best parameters for each separate stock prediction. **ARIMA**: best parameter used: P:0, d:1, q:0. **Holt-Winter**: best parameter used: (Weights)Alpha: 0.92 (trend factor)Beta: 0.02 (smoothing factor)Gamma: 0.81

In tuning the parameters of the ANN and RNN the following parameters where used in a method of grid search:

- Look Back 3,5,7 (ANN only)
- Epochs: 20 ,50,100
- Batch Size: 1,2,3
- Optimizer: SGD, Adam, Nadam

ANN: All parameters but optimizer deemed little change to final result. the following parameters yield the best performance measured by RMSE: Look-back: 3 Epochs: 50 Batch Size: 2 Optimizer: Nadam

RNN: Like ANN, optimizer was the most influenced parameter. the following parameters yield the best performance measured by RMSE: Epochs: 20 Batch Size: 3 Optimizer: Nadam

5.2 Experimental results

Visual inspection. Reviewing the 4 method prediction compare to actual stock price Figure 3, it can be seen that none of the methods manged to predict change in stock trend ahead of time, some methods like RNN managed to react faster to trend change compare to other methods. Figure 3

Figure 3: DAX stock price prediction over time

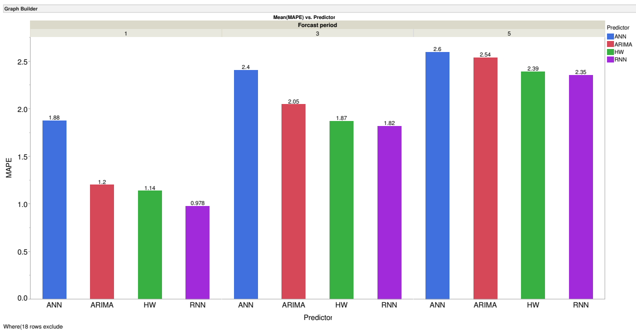

Figure 4: Mean(MAPE) Per method Per prediction time

Prediction Time. In evaluating the performance of all methods for predicting for T+1/3/5 time RNN yielded best performance in all prediction times, slightly over the auto regression methods, the simple ANN proved to be least effective. all methods MAPE was lower as prediction time was closer to the given test set Figure 4.

ETF Vs Single Stock. For prediction stock prices for time T+1 incorporating all dataset divided into ETF and stocks all methods showed better MAPE for ETF compare to single stock prediction. this is aligned with our assumption where ETF which cooperates multiply stock is better candidate for stock prediction Figure 5.

Combination Of Multiply Stocks Prices. In this method we entered the neural network a training and test set of a stock and a reference stock from the same business sector, as can be seen in 6 this method did not proved to be more accurate in predicting stock prices compare to the conventional RNN method. one stock was equal and in the second poor results compare to RNN.

Buy&Sell Algorithm. Examining the ability of each method in predicting stock price trend yielded results between 36% to 63% Hit rate, as previous chapters RNN proved to be the most accurate method in predicting the stock trend with maximum of 35 out of 55 hits. Table 1 and Table 2.

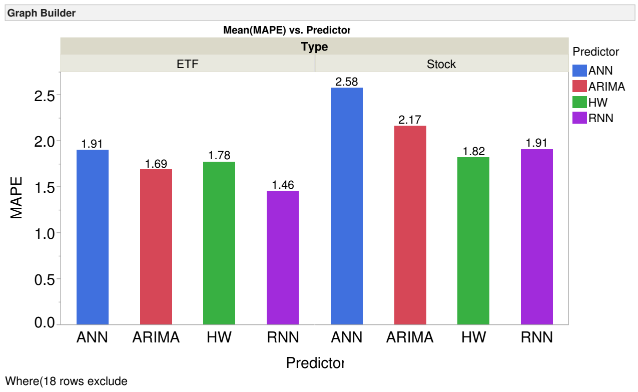

Figure 5: Mean(MAPE) Per method Per stock type

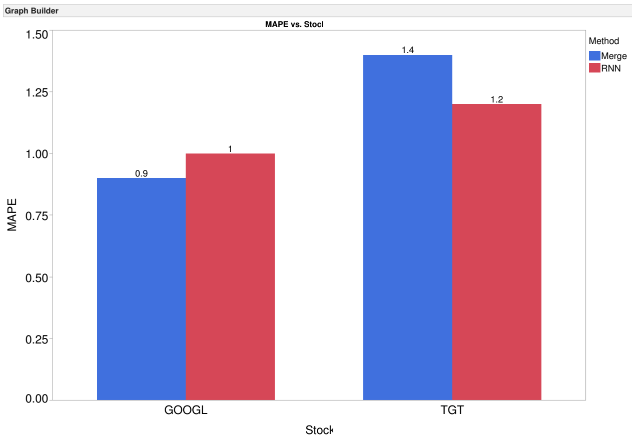

Figure 6: MAPE Per method Per stock

	1	**3**	**5**
RNN	31	32	32
ANN	31	32	30
ARIMA	24	26	31
Holt-Winter	20	26	28

Table 1: DAX Hit count

	1	**3**	**5**
RNN	28	28	34
ANN	29	31	35
ARIMA	20	32	35
Holt-Winter	20	31	35

Table 2: Apple Hit count

6. CONCLUSION

In this paper we examined the ability of predicting a closing price of a stocks and ETF using the previous closing price, we have compared th 4 algorithms in 5 scenarios. Prediction for different time periods, comparison of single stock to ETF, prediction of the trend and prediction of a stock closing price utilizing information from a stock price of a company in the same sector. While no method showed ability to predict a change in the stock price (shift in trend) few methods reacted faster to the change and outperformed. Throughout the experiment the recurrent neural network (RNN) proved to be the most efficient method. The prediction of trend showed the ability to predict a change in the stock price, all methods predict the trend change but the recurrent neural networked (RNN) proved to be the most efficient in every aspects.

The paper presents strong evidence that the RNN could be the leading algorithm in stock price prediction. we suggest to further develop the various aspects of utilizing the RNN in stock price prediction. one option is increasing the training dataset for more then the requested stock price prediction forcing the network to detect patterns in the stocks.

Additional work can be used by incorporating additional information such as text analysis Of relevant journals/portals and find correlation between the stock and ETF then give there closing price to the method, The idea is to introduce relevant information in the prediction system before it affects the stock market. i.e. if a CEO of a company of interest tweets very important information, we could use this information to improve our prediction on the future behavior of the company share value.

The dataset we use was a for most part monotonous and there was no significant change in trend (shift from a positive rise in price to a negative decrease).To better evaluate the methods prediction we suggest evaluating on a bigger dataset(couple of years) which the trend has a significant movement.

7. REFERENCES

[1] Abu-Mostafa, Yaser S., and Amir F. Atiya. "Introduction to financial forecasting." Applied Intelligence 6.3 (1996): 205-213.âĂŔ

[2] CONG, Shan-shan, et al. "Research on the Price of Stock Index Futures with ARIMA Model." DEStech Transactions on Economics and Management iceme-ebm (2016).âĂŔ

[3] Kostov, Georgi, Sasho Nedelkoski, and Goran Stojanovski. "Implementation of two automated systems for trading shares on the stock market."âĂŔ

[4] White, H. 1988, âĂŸEconomic prediction using neural networks: a case of IBM daily stock returnsâĂŹ, IEEE International Conference on Neural Networks, vol. 2, pp. 451-458.

[5] Chen, C. H. "Neural networks for financial market prediction." Neural Networks, 1994. IEEE World Congress on Computational Intelligence., 1994 IEEE International Conference on. Vol. 2. IEEE, 1994.âĂŔ

[6] Hill, Tim, et al. "Artificial neural network models for forecasting and decision making." International Journal of Forecasting 10.1 (1994): 5-15.âĂŔ

[7] Balkin, Sandy D., and J. Keith Ord. "Automatic neural network modeling for univariate time series." International Journal of Forecasting 16.4 (2000): 509-515.âĂŔ

[8] Elman, Jeffrey L. "Finding structure in time." Cognitive science 14.2 (1990): 179-211.âĂŔ

[9] Ghahramani, Zoubin, and Robert B. Allen. "Temporal processing with connectionist networks." Neural Networks, 1991., IJCNN-91-Seattle International Joint Conference on. Vol. 2. IEEE, 1991.âĂŔ

[10] Kim, Kyoung-jae, Ingoo Han, and John S. Chandler. "Extracting trading rules from the multiple classifiers and technical indicators in stock market." Proceedings of KMISâĂŹ98 International Conference. 1998