Predicting S&P 500 Market Price by Deep Neural Network and Enemble Model

Feiyu Wang

The Oakwood School, GREENVILLE, NC, USA

Abstract—The method to predict the movement of stock market has appealed to scientists for decades. In this article, we use three different models to tackle that problem. In particular, we propose a Deep Neural Network (DNN) to predict the intraday direction of SP500 index and compare the DNN with two conventional machine learning models, i.e. linear regression, support vector machine. We demonstrate that DNN is able to predict SP500 index with relatively highest accuracy.

1 Introduction

In the past few years from the contemporary society, the whole world has experienced a tremendous economic or financial growth. With such rapid development in world economic aspect, people start to pay more attention in searching for proper ways to get profits. Thus, in this trend of development aspect, the term “stock market” starts to become extremely popular among companies since those investors know that the stock exchange can provide them the opportunity to issue the share and thus get funds for companies for further business requirements. Therefore, how to predict stock market has become a pretty significant concern among those investors. Stock itself acts as a double-edged sword, that is for any investors who dare to put money in a random company's stocks, there is a fifty-fifty percent chance for that person to either win or lose. In the past few years, the apple company is one of the most powerful and prestigious enterprises in the world, so every single time the company gain profits from a variety of reasons, all those previous investors can take advantage from that as well. However, who has the capability to recognize the prospect of such company from the time it was just established? Therefore, it’s a matter of chance to determine whether a person can gain huge earnings or not from those stocks. Now, back to current society, even though a great variety of advanced technology has already been put into used for the ultimate goal in predicting the stock market price as precise as possible, this objective is stilled considered highly challenging. This is primarily because of the uncertainties involved in the movement of the market, which can be influenced by several significant, but untracking factors like general economic conditions, investors and common citizens’ expectations as well as political events. Nevertheless, according to recent investigations, rather than behaving in a random order or series, those stock market prices were revealed as a dynamic and non-linear order [1], which is important because this discovery has provided those scientists or company managers a valuable chance to develop a mathematical machine model to trace the data especially for non-linear variables. Aside from this important discovery, it is surprisingly known that for most people and investors, instead of focusing on what the exact stock market price will be in the future for a certain company, the ability to find the right direction was regarded as a relatively more useful way to predict market prices.

The rest of this paper mainly focused on predicting the stock market price for s&p 500, that is an American stock market index based on the market capitalization of 505 large companies having common stock listed on NYSE, NASDAQ… ETC. Rather than focusing on one single model and its experimental procedures, this paper will be focused on testing a variety of other models for their accuracy and generate the best model by comparing the final result.

The remaining portion of the article is organized as follows: In section 2, we provide some related research that other scientists have done in the same area. In section 3 we explain and analysis our research methods. Detailed description and graph of each model we are using will be provided in section 4. After that, in section 5 the final experimental results will be discussed. Conclusion and overall summary will all be given in section 6 at last.

2 Related works

In the past few decades or so, numerous scientists and experts from different academic fields have worked together on predicting the future market price, and so far they are still continuously striving for a result with the greatest accuracy from those data information. From a

ewang@theoakwoodschool.org

great variety of machine learning techniques and artificial intelligence systems that have been utilized so far, neural network was commonly accepted as the most powerful technique for it's well organized structure and underlying principles. For instance, the Artificial neural network (ANN) has the ability to learn and model non-linear and complex relationships [1]. Moreover, it can generalize data from the previous market price movements and predict unseen information. The Multilayer Perceptron (MLP) is mainly used in the aspect for speech recognition and translation technologies for its nonlinearity [1]. More from that, the Convolutional Neural Networks (CNM) have demonstrated great efficiency and accuracy especially in image and video recognition areas, partially because it employs a convolutional operation that is able to extract local image patterns [1]. Other methods such as Particle Swarm optimization (PSO) [2] and classifier systems [3] have been utilized to predict the stock market price.

However, as the technology in the contemporary world was still growing at a tremendous speed, by just simply utilizing neural network might not satisfied most people's need in predicting the future market prices, thus other methods like SVM and Bayesian belief networks [4] or even some hybridizing method were currently being experimented.

Since our ultimate research goal is to derive the best accuracy result from a variety of different machine learning techniques including Neural network, linear regression, SVM and so on, my goal will primarily focus on comparison of the final result between those different techniques after implementing those one by one. Some experts have already attempted to generate the best data model by comparing results from different techniques. Erdinc Altay and Satman, M Hakan have tried to determine the model that functions better between Artificial neural Network (ANN) and linear regression [5]. Rohit Choudhry and Kumkum Garg combines the general algorithms (GA) and Support vector machines as the overall hybrid-SVM models, and the results revealed that this model outperforms the alone SVM system completely [6]. Moreover, MC_Lee has developed a hybrid model formed by SVM and F_SSFS (F-Score and Supported Sequential Forward Search) and compared with BPNN (Back propagation neural network) [7]. Eventually, their study demonstrated that SVM-F_SSFS hybrid model has the highest level of accuracy and generalization performances, thus outperform the single BPNN as well [7].

3 Methods

In this section, the major goal is to provide a detailed description of the research and the overall procedures.

3.1 Models

3.1.1 Linear regression:

Linear regression is a commonly used model to capture the linear combination and dependence between variables [8]. Its equation is given by:

$$y(X,W) = w0 + w1x1 + ... + wDxD \quad [8]$$

whereas W(w0 … wD) is the parameter determined by fitting the data to model, X(x1...xD) is the input independent variable and y is the output dependent variable.

3.1.2 SVM:

Support Vector Machine (SVM) is a supervised decision machine that is capable to extract the hyperplane from a variety of data points, and one major property of it is that the determination of the model parameters corresponds to a convex optimization problem, so any local solution is a global optimum [8].

Its equation is given by:

$$y(x) = wT\varphi(x) + b \quad [8]$$

While wT refers to a series of parameters in T matrix rotation, φ(x) denotes a fixed feature space-transformation, b means bias parameter and y(x) means the output dependent variable.

3.1.3 DNN:

Deep neural Network is a type of network that contains an intricate structure of two or more hidden layers that enable it to process through complex and abstract ideas [9]. The neural network structure is shown in Figure 1.

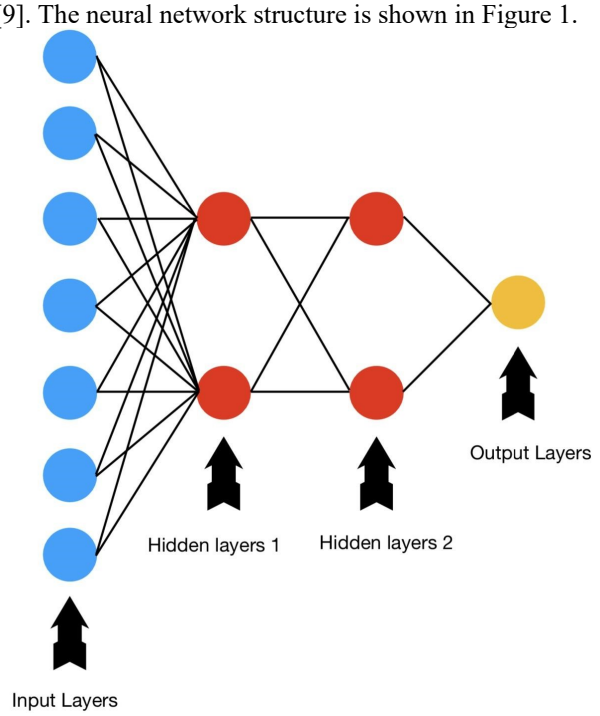

Figure 1. A basic structure of neural network

3.2 Features and Labels

TABLE I. FEATURE TABLE:

Feature name	Formulation
Feature #1	[high(t) - low(t)] / open(t)
Feature #2	[high(t) - low(t)] / close(t-1)
Feature #3	[open(t) - close(t-1)] / close(t-1)
Feature #4	close(t) / close(t-1)
Feature #5	[high(t) - low(t)] - [(high(t-1) - low(t-1)]
Feature #6	Feature #3 (t) - Feature #3 (t-1)
Feature #7	EMA, ATR, RSI

Features are extracted to normalize the changes involved in stock price changes. The variable for each feature is precisely chosen so the value can be best fit to the model. In a certain stock market trading day, the highest reachable stock price is called maximum price value (high(t)) and the lowest stock price is called the minimum price value (low(t)). While the opening price of a certain day is denoted (open(t)), the closing price is termed (close(t)). Addition to the standard EMA, ATR and RSI identifier parameters, which all have a value of 5, other features are selected primarily based on the relationship among those different values in a period of days.

EMA, the exponential moving average, can accurately calculate the average number of data values among all dates in the research. To calculate the EMA, the major equation is shown as:

$$EMA(t) = EMA(t-1) + alpha * (price(t) - EMA(t-1)) \quad [10]$$

Where the alpha refers to an exponential value that we can change and EMA(t-1) refers to the exponential moving average of the day before the given day t, and price(t) means the open price of a given date, then we normalized it with the equation to extract our feature:

$$f(EMA(t)) = (EMA(t) - EMA(t-1)) / EMA(t-1)$$

In this case, the number derived from the result is also 5.

Aside from EMA, since the features in the dataset contain different range, for instance: the "opening price", "closing price" and "maximum price" for each days were always varied differently, we built 6 variable for the normalized price changes feature to make it easier to calculate for the later model experiments:

- The feature #1 is defined as the difference between maximum and minimum price and divided by the same day's opening price.
- The feature #2 is defined as the difference between maximum and minimum price and divided by the closing price from last day.
- The feature #3 is defined as the difference between the opening price and the closing price from last day and divided by the closing price from last day.
- The feature #4 is defined as dividing last day's closing price by the day's before that day's closing price.
- The feature # 5 is defined as the difference between today's maximum and minimum price's difference and the day's before today's maximum and minimum price's difference.
- The feature # 6 is defined as the Subtract the normalized price change 3 of last day from the normalized price change 6 of today.
- The feature # 7 is defined as those standardize features.

We determine our labels after selecting our 7 features, that were defined as the direction of intraday price movement:

$$A(L) = (A(closeP) - A(open)) ./ (A(closeP)$$

Whereas A(L) refers to the label, A(closeP) refers to the close price of the day before a certain date and A(openP) refers to the open price of that certain date, so overall, the label is the range of price movement divided by the closed price of a previous date.

3.3 Backtesting

The backtesting in a stock market prediction scenario refers to the strategy to predict the future unknown price from the known prices of the historical data, which is our main goal. At first, we set our number training data to be 500 days.

Then we set our number of days that a model will be updated every single time to 20 days. If we are currently running a huge company and trying to avoid any unnecessary financial risks from stock prices fluctuation, we update the models in backtesting approximately on a monthly basis (20 days).

Then, we determined to start the data from January fourth in two thousand sixteen. After that, we started to train our model through linear regression, deep neural network and Support Vector machines.

4 Experimental Results

Since the main idea is to predict the stock market price in the future, we collected our data information from 2016 to 2017 to guarantee the lowest risk generated from the research.

Aside from the three models we have been trained on previously, we have added a new ensemble model, that contains unique properties from each of the previous three models to consummate the perfection of this experiment.

However, accuracy itself is just a number that works for a temporary amount of time to generate the best result. If such model is utilized for long term examination, it must be ensured to benefit the company's owner with real profits. Rather than proving the overall profits and loss (PnL) value through a whole period of time, it's much more beneficial to see the Average PnL, that is the mean value of the overall PnL, which demonstrates the daily profits gained through each models, thus determining the best model that can benefit the whole company.

TABLE II. MODEL ACCURACY CHART

Models	Accuracy (%)	Average PnL (Profit and loss)
SVM	53.08	0.5290
Linear Regression	57.26	2.5253
DNN	57.06	2.7198
Ensemble model	62.24	3.2786

5 Conclusion

In this paper, we tested our data through models like DNN, SVM and linear regression. Varies features, parameters and labels from the data like normalized price change by calculating through the open market price, close market price and maximum minimum prices are determined and utilized. The Results revealed that the proposed DNN model achieved better performance compared to linear regression and SVM in a long-term process for both its relatively high accuracy and PnL.

There are still lots of improvements that can be carried out on this research, and more work can be done in the field of predicting stock market prices. In the future, if all other political factors, humanity factors and economic factors were being considered and controlled, and more complex and efficient models have been created, more convincing and better results will be generated. Additionally, the combination among a certain number of models is more likely to obtain a higher accuracy also rather than a single one. Thus, overall, by taking all previous factors into consideration, the method to predict stock market prices precisely will not be a dream anymore, but a fact.

References

1. Mehta, Anukrati. "A Comprehensive Guide to Types of Neural Networks." 25 Jan. 2019, www.digitalvidya.com/blog/types-of-neural-networks/.
2. Hegazy, Osman, Omar S. Soliman, and Mustafa Abdul Salam. "A machine learning model for stock market prediction." arXiv preprint arXiv:1402.7351 (2014).
3. S. Schulenburg and P. Ross, "Explorations in LCS models of stock trading," Advances in Learning Classifier Systems, 2001, pp. 151–180.
4. R. K. Wolfe, "Turning point identification and Bayesian forecasting of a volatile time series," Computers and Industrial Engineering, 1988, pp 378–386.
5. Altay, Erdinc, and M. Hakan Satman. "Stock market forecasting: artificial neural network and linear regression comparison in an emerging market." Journal of Financial Management & Analysis 18.2 (2005): 18.
6. Choudhry, Rohit, and Kumkum Garg. "A hybrid machine learning system for stock market forecasting." World Academy of Science, Engineering and Technology 39.3 (2008): 315-318.
7. Lee, Ming-Chi. "Using support vector machine with a hybrid feature selection method to the stock trend prediction." Expert Systems with Applications 36.8 (2009): 10896-10904.
8. PRML: Bishop, Christopher M. Pattern recognition and machine learning. Springer, 2006.
9. Nielsen, Michael A. Neural networks and deep learning. Vol. 25. San Francisco, CA, USA: Determination press, 2015.
10. Shah, Vatsal H. "Machine learning techniques for stock prediction." Foundations of Machine Learning| Spring 1.1 (2007): 6-12.