2022 International Conference on Machine Learning and Intelligent Systems Engineering (MLISE) | 978-1-6654-9246-1/22/$31.00 ©2022 IEEE | DOI: 10.1109/MLISE57402.2022.00043

Stock Price Prediction using Linear Regression and LSTM Neural Network

Haorui Zhang
International School of Business and Finance, Sun Yat-sen University
Guangzhou, China
zhanghr55@mail2.sysu.edu.cn

ABSTRACT: The stock market has a profound influence on the modern society. Therefore, predicting stock prices is always a hot research topic. In this paper, we use linear regression models and LSTM models based on machine learning to predict the stock price of Amazon. In order to let the algorithm more available for individual investors, we only use the historical stock price of the company as data sources. We use MSE, MAE, RMSE, R^2 to evaluate those models. We find in the case of limited data sources, the simplest linear regression is even better than LSTM models. In LSTM neural network, Single Layer LSTM performs better than Multi-layer LSTM and Bi-directional LSTM. We also find LSTM algorithms that epoch = 200 are more suitable in stock prediction than both algorithms that epoch = 50 which are faced with underfitting problem and algorithms that epoch = 500 which are faced with overfitting problem. This paper applies Bidirectional LSTM and Multi-layer LSTM into stock price prediction, which also provide the possibility of parameter setting improvement.

Keywords; Stock price prediction; Machine Learning; Deep Learning; Linear regression; Single layer

I. INTRODUCTION

Stock market has a profound impact in economy of modern society. A rise or fall in the share price plays a foremost role in determining the investor's gain. For institutional investors, they can predict future stock price relatively more accurate since they have much more information public don't know and they can build accurate quantitative models through their powerful database. However, these algorithms and databases are not available for individual investors, they only have constricted data sources like historical stock price and personal computer that cannot support large computing operations. Furthermore, institutional investors typically construct complex investment strategies and change their inventory through high frequency trading for purpose of hedging the large risk of the investment market, which is even more difficult for single individual investors. For individual investors, the most viable strategy is hold one or more stocks for a period of times and make a profit from capital gains (the selling price minus purchasing price). Therefore, we build linear regression models and LSTM models to indicate the price change of single stock by using Amazon, a significant successful company, as the testing company.

For these reasons, we build linear regression models and LSTM models to predict the stock price only rely on historical close price. Even if other factors are also relevant to future stock price, we neglect them because they are not only inapplicable to all stocks but also impossible for individual investors to collate into high-frequency databases. Based on the conclusion of Efficient Market Hypothesis (EMH), indicating the stock price entirely accurate is impossible but relative accurate prediction is still achievable. In this paper, we focus on how to promote the accuracy of prediction through close price. Research about future stock price prediction is always in great demand because it could bring tremendous benefits for people who accurately or relatively accurately predict the stock price and many scholars have studied this topic through a variety of algorithms. Henriquea et al. (2018)" Use support vector regression (SVR) algorithms to predict share prices for large and small businesses and conclude that SVR performs well when model is updated periodically [1]. Patel, Shah, Thakkar and Kotecha (2015) research on predicting direction of movement of stock and stock price index for Indian stock market by SVM, ANN, random forest and naive-Bayes through previous data and ten technical indicators [2]. Alkhatib et al. (2013) use K-Nearest Neighbor (KNN) algorithm and non-linear regression approach to predict stock prices of six major companies listed on the Jordanian stock exchange and conclude that prediction result of KNN is very ideal [3]. Vijh et al. (2020) use ANN, Random Forest to predict the stock price and evaluate through RMSE and MAPE [4]. Yu and Yan (2020) use Deep neural networks (DNNs) to predict multiple stock indices for different periods and conclude that the accuracy is very high [5]. Kumar et al. (2018) use SVM and tree methods and Softmax and Naïve Bayes as using algorithms. They conclude that for large datasets, the Random Forest algorithm is the best, but for small datasets, naive Bayesian categorizer [6]. Linear regression is one of the most mainstream prediction methods and many researchers studied on that. Roy et al. (2015) propose an LASSO method based on a linear regression model and predict the stock price of Goldman Sachs Group Inc. and made the conclusion that it outperforms than ridge linear regression model [7]. Ismail et al. (2009) predicted the price of Gold by Multiple Linear Regression (MLR) and consider factors of local and American stock index and several important factors of economic conditions like inflation and money supply and build two models according to different quantities of factors and conclude that the model with only four variance is more precise [8]. LSTM is also extremely popular in stock price prediction since the development of deep learning algorithms. Selvin et al. (2017) use LSTM, RNN and CNN to predict NSE listed companies and benchmark their performance [9]. Roondiwala et al. (2017) use the Recurring Neural Network (RNN) and Long-Term Memory (LSTM) method to forecast stock indexes [10]. Sunny et al. (2020D) uses the LSTM model and the bi-directional LSTM model, changing the number of epochs and selecting different layers to find a better, more accurate model for forecasting future market prices [11]. Zhuge et al. (2017) use LSTM Neural Network select the stock exchange data, Shanghai Composite index and emotional data as the input variables in order to guess stock price [12]. Most of them use several non-public datasets or unavailable datasets while we only use overt, available datasets.

978-1-6654-9246-1/22/$31.00 ©2022 IEEE
DOI 10.1109/MLISE57402.2022.00043

However, many researches use factors not available to individual investors or the sample size of stock price is not big enough. In this paper, we use all time stock price of Amazon and only focus on constricted data sources. In addition, some of them only use complex deep learning algorithms and lack of comparison while we compare both basic models and complex deep learning models. About the setting of LSTM, many researches only use the simplest Single Layer LSTM and don't public their parameter setting, while in this paper, we also apply Multi-layer LSTM and Bi-directional LSTM into prediction and try to modulate parameters of them, which overcome the shortcoming previously. Although both Linear Regression and LSTM are well adopted methods in this field, we fill the blank of parameter setting in Linear Regression and layer constriction in LSTM, which bring some innovation in this research.

The rest of the paper is laid out as follows. In Section 2, data and it's preprocessing are introduced and the linear model and LSTM model are explained respectively and we well introduced evaluation indicators. In Section 3, the empirical outcomes are presented by tables for evaluating indicators. In the end, Section 4 sums up.

II. Methodologies

A. Data and Preprocessing

We obtain daily data of Amazon from May 15, 1997 to March 29, 2022 from yahoo finance dataset. We don't add any other variables in light of the availability to individual investors. Here we made a descriptive statistics table and plot a figure to visualize the overall performance of Amazon for the reason to concisely introduce the dataset.

TABLE I. DESCRIPTIVE STATISTICS OF AMAZON FROM MAY 15, 1997 TO MARCH 29, 2022

	Open	High	Low	Close	Volume
Count	6260	6260	6260	6260	6260
Mean	566.3303	572.6860	559.3188	566.1539	7268439
Std	919.8846	929.8312	908.7287	919.2979	7107488
Min	1.4063	1.4479	1.3125	1.3958	487200
25%	39.0100	39.8000	38.3750	39.0825	3528800
50%	107.5625	111.4900	103.2188	105.8750	5420950
75%	598.0925	602.6025	589.4050	596.0425	8235100
Max	3744.0000	3773.0801	3696.7900	3731.4099	104329200

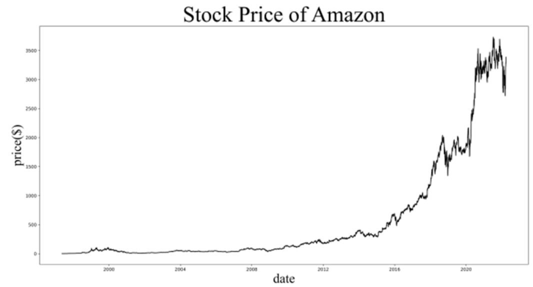

Figure 1. Daily close price of Amazon(AMZN) from May 15, 1997 to March 29, 2022

After that, we preprocess Amazon data. First, we normalize the data range from 0 to 1 by MinMaxScaler in order to make the data more stable. Second, we build a time stamp function. Third, we split the data into training set and test set, the size of test set is 300 in Linear Regression and 300 + the number of timestamp in LSTM model in order to stay size of predict pictures same as Linear Regression.

B. Linear Regression

Various types of linear regression model could be developed by the stock close price. In this project, we have considered to use the last several days as X and next day as Y which is also known as autoregressive model (AR).

$$Expected\ Close\ Price_{n+1} = \sum_{i=1}^{n} \beta_i Close\ price_i + \beta_0 \quad (1)$$

Where n: the number of previous days used to predict the stock price of day following, β_i: the parameters of each close price, β_0: the constant term. This model uses Ordinary Least Squares (OLS) method to decide the value of β_i which could minimize the number of residual sum of squares (RSS).

$$RSS = \sum_{j=1}^{N} \left(y_j - \hat{y}_j\right)^2 \quad (2)$$

$$\min_{\sum_{i=1}^{n} \beta_i} RSS\left(\sum_{i=1}^{n} \beta_i\right) = \sum_{j=1}^{N} (y_j - \beta_0 - \sum_{i=1}^{n} x_{ij} \beta_i)^2 \quad (3)$$

Where N: the number of sample size, y_j: the true value, $\hat{y}_j$: the predicted value. For the sake of finding out the best parameter n (number of previous days we set), we separately set n as 7, 15, 30, 60, 100 and we call them AR (7), AR (15), AR (30), AR (60), AR (100).

C. LSTM

Long short-term memory (LSTM) is a special form of Recurrent Neural Network (RNN). RNN is an extremely important method in deep learning (DL) which is very practical and advanced in processing sequential data especially in short period sequential data. However, in long sequential data, the problem of vanishing gradients and exploding gradients makes RNN to be a poor performer. LSTM effectively resolve these problems so that it has better performance in long sequential data. Therefore, it is pragmatic and popular in stock price prediction since daily stock price is a relatively long sequential data [13].

In RNN, there are only one transmission state, while LSTM replaces it with two transmission states c^t (cell state) and h^t (hidden state). Generally, c^t is the long-term memory since it changes slowly and h^t is the short-term memory since it changes fast. The structure of LSTM model is recealed in the following Figure.

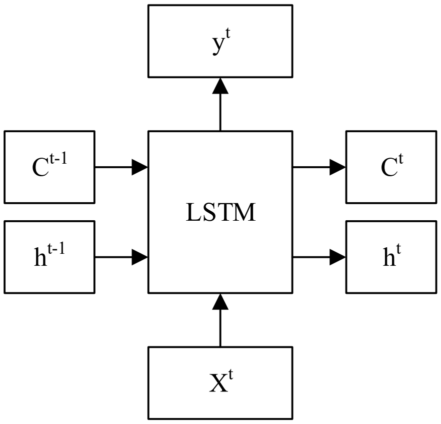

Figure 2. Operating principle of LSTM.

c^t and h^t are determined by LSTM cells. An LSTM cell is made up of four gates called input gate, cell gate, forget gate and output gate. It is activated by sigmoid layer and tanh layer. Sigmoid layer generates values ranging from zero to one, indicating how much of each component should be allowed to pass and tanh layer creates a new vector that is stored in the state. There are 3 stages in an LSTM cell. The first stage is forgotten stage when input gate gives information and forget gate decide which information of c^{t-1} need to be forgotten. The second stage is selective memory stage when information would be selectively remembered and c^t would be output by cell gate. The last stage is output stage when h^t is decided by output gate. The equations are as follows.

$$f_t = \sigma_g\left(W_f \cdot [h_{t-1}, x_t] + b_f\right) \tag{4}$$

$$i_t = \sigma_g(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{5}$$

$$\tilde{c}_t = tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \tag{6}$$

$$c_t = f_t * c_{t-1} + i_t * \tilde{c}_t \tag{7}$$

$$o_t = \sigma_g(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{8}$$

$$h_t = o_t * tanh(c_t) \tag{9}$$

Where σ_g: sigmoid function, tanh: tanh function, x_t: vector of input, h_t: vector of output, f_t: forget gate's activation vector, i_t : input gate's activation vector, $\tilde{c}_t$: vector of cell input activation, c_t: vector of cell state, o_t: vector of output gate's activation, W : parameter matrix of corresponding gate, b : parameter vector of corresponding gate.

There are two more types of LSTM algorithms we use in this paper: Multi-layer LSTM and Bi-directional LSTM. In Multi-layer LSTM, we set 2 layers of LSTM layers added up to determine y_t, the output of Layer 1 is also the input of Layer 2, then the y_t is determined by the output of Layer 2. Bi-directional LSTM prioritize the sequence information in both backward direction and forward direction. In this model, y_t is determined by 2 LSTM layers, the first one is Forward Layer contains positive time series LSTM cells while the second one Backward Layer contains negative time series LSTM cells.

In LSTM models, we can choose the epoch of training process to promote the performance of prediction. For simplicity and intuition, we will use LSTM (type of algorithms, number of previous stock price as input, number of epoch) where Single Layer LSTM: type 1, Multi-layer LSTM: type 2, Bi-directional LSTM: type 3.

D. Analysis

So as to analyze the performance of the system, we used four indices: Mean Square Error (MSE), Root Mean Square Error (RMSE), Mean Absolute Error (MAE) and R-Square (R^2). All of them can estimate the degree of deviation between prediction value and real value. In fact, usually a better algorithm is better than worse algorithms in all of these indices.

$$RMSE = \sqrt{\frac{1}{T}\sum_{i=1}^{T}(y_i - \hat{y}_i)^2} \tag{10}$$

$$MSE = \frac{1}{T}\sum_{i=1}^{T}(y_i - \hat{y}_i)^2 \tag{11}$$

$$MAE = \frac{1}{T}\sum_{i=1}^{T}|y_i - \hat{y}_i| \tag{12}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{T}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{T}\left(y_i - \bar{y}_i\right)^2} \tag{13}$$

Where T: the total samples. For MSE, RMSE, MAE, the lower the number, the better. For R^2, the larger the number, the better. If the number of R^2 is close to 1, we could tell that the effect of stock price prediction process is available. Later we will show them on the same table.

III. RESULTS

The analysis tables of stock price prediction are as follows:

TABLE II. TEST ERROR OF LINEAR REGRESSION AND LSTM MODELS

Test Error	MSE	RMSE	MAE	R2
AR (7)	3718.8361	60.9823	44.0624	0.9024
AR (15)	3790.9247	61.5705	45.1151	0.9005
AR (30)	3815.3859	61.7688	45.1282	0.8999
AR (60)	4033.5327	63.5101	47.3664	0.8942
AR (100)	4225.9131	65.0070	48.7786	0.8891
LSTM (1,40,50)	6994.1378	83.6310	65.1978	0.8097
LSTM (1,40,200)	4046.3681	63.6111	45.7850	0.8857
LSTM (1,40,500)	4495.4494	67.0481	50.9127	0.8753
LSTM (1,60,50)	6712.3415	81.9289	63.4330	0.8069
LSTM (1,60,200)	3808.7773	61.7153	45.1862	0.8972
LSTM (1,60,500)	4880.6732	69.8618	52.5061	0.8655
LSTM (2,40,50)	10788.6765	103.8686	85.0440	0.6754
LSTM (2,40,200)	4019.1949	63.3971	45.3440	0.8876
LSTM (2,40,500)	5320.5861	72.9424	57.4920	0.8458
LSTM (2,60,50)	10311.2793	101.5445	81.8193	0.7236
LSTM (2,60,200)	5630.4541	75.0364	57.1299	0.8635
LSTM (2,60,500)	4803.9233	69.3103	51.4532	0.8529
LSTM (3,40,50)	8727.9546	93.4235	77.1092	0.7207
LSTM (3,40,200)	3940.1580	62.7707	44.9827	0.8849
LSTM (3,40,500)	5092.0272	71.3584	54.7994	0.8412
LSTM (3,60,50)	5588.4872	74.7562	58.5384	0.8608
LSTM (3,60,200)	4008.7983	63.3151	46.1848	0.8927
LSTM (3,60,500)	5101.3734	71.4239	54.4345	0.8524

TABLE III. NING ERROR OF LINEAR REGRESSION AND LSTM MODELS

Training Error	MSE	RMSE	MAE	R2
AR (7)	281.7864	16.7865	6.3166	0.9994
AR (15)	277.3426	16.6536	6.3084	0.9994
AR (30)	271.4847	16.4768	6.3418	0.9995
AR (60)	255.3814	15.9807	6.3790	0.9995
AR (100)	240.9491	15.5225	6.3911	0.9995
LSTM (1,40,50)	533.3023	23.0933	9.8847	0.9988
LSTM (1,40,200)	300.3487	17.3306	8.1917	0.9993

LSTM (1,40,500)	340.3597	18.4488	9.0858	0.9992
LSTM (1,60,50)	526.3857	22.9431	14.2158	0.9988
LSTM (1,60,200)	299.1424	17.2957	9.6466	0.9993
LSTM (1,60,500)	276.0172	16.6138	7.3573	0.9993
LSTM (2,40,50)	726.7572	26.9584	12.6729	0.9983
LSTM (2,40,200)	537.2758	23.1792	18.7309	0.9988
LSTM (2,40,500)	354.2281	18.8210	8.1928	0.9992
LSTM (2,60,50)	594.9164	24.3909	12.2013	0.9986
LSTM (2,60,200)	335.1069	18.3059	8.7270	0.9993
LSTM (2,60,500)	241.7809	15.5493	6.3026	0.9994
LSTM (3,40,50)	484.4273	22.0097	9.5178	0.9989
LSTM (3,40,200)	344.1142	18.5503	10.1900	0.9992
LSTM (3,40,500)	326.5072	18.0695	8.3448	0.9993
LSTM (3,60,50)	483.9637	21.9992	13.3486	0.9989
LSTM (3,60,200)	269.1287	16.4051	7.7559	0.9994
LSTM (3,60,500)	272.0678	16.4945	7.2597	0.9993

As tables show, both Linear Regression and LSTM can relatively accurately predict the stock price since in most algorithms, the number of R^2 is close to one. the overall performance of Linear Regression models is better than all kinds of LSTM models.

In LSTM models, algorithms that epoch = 200 have better performance than algorithms that epoch = 500 and algorithms that epoch = 50. We also find some algorithms that epoch = 50 have bad performance like LSTM (2,40,50) and LSTM (2,60,50). We believe that algorithms that epoch = 50 have underfitting problem and algorithms that epoch = 500 have overfitting problem. Algorithms that epoch = 200 are more suitable to predict stock price. Single layer LSTM algorithms have the best performance, Bi-directional LSTM algorithms are a bit poor, Multi-layer LSTM algorithms have the worst performance.

About Linear Regression models, we find that with the increasement of previous days we use, the training error becomes lower while the test error tends to be higher. In this prediction process, using many days lead to overfitting problem. These algorithms overlearning the data of training set which result in worse performance of fitting in real data.

IV. Conclusions

Predicting stock prices is a fiendishly difficult business even for informed institutional investors. It is impossible to predict stock prices absolutely accurate even if one gets wide varieties of relative data. Furthermore, many parameters that have great influence on stock price cannot be discovered or predicted like the reform of the tax law. In this paper, we use not only Linear Regression models but also LSTM neural network for the purpose of predicting the stock price of Amazon by only historical stock price of Amazon which is open to all investors. We thought LSTM neural network will perform better than Linear Regression model but the opposite is true which probably because LSTM neural network need more kinds of input to improve the training effect. But it doesn't mean Linear Regression models are better for the reason that it can only predict a very short period of stock price otherwise they will show you the stock price would tend to infinite or zero. In fact, stock price has nonlinear characteristics but Linear Regression performs great, we believe it is due to in test set, each prediction is only one step forward so that the nonlinear characteristics could be offset. In the future, we will focus on finding more available helpful dataset and using other models so as to improve the accuracy of prediction.

Reference

[1] Henrique, B. M., Sobreiro, V. A., & Kimura, H. (2018). Stock price prediction using support vector regression on daily and up to the minute prices. *The Journal of finance and data science*, *4*(3), 183-201.

[2] Patel, J., Shah, S., Thakkar, P., & Kotecha, K. (2015). Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques. *Expert systems with applications*, *42*(1), 259-268.

[3] Alkhatib, K., Najadat, H., Hmeidi, I., & Shatnawi, M. K. A. (2013). Stock price prediction using k-nearest neighbor (kNN) algorithm. *International Journal of Business, Humanities and Technology*, *3*(3), 32-44.

[4] Vijh, M., Chandola, D., Tikkiwal, V. A., & Kumar, A. (2020). Stock closing price prediction using machine learning techniques. *Procedia computer science*, *167*, 599-606.

[5] Yu, P., & Yan, X. (2020). Stock price prediction based on deep neural networks. *Neural Computing and Applications*, *32*(6), 1609-1628.

[6] Kumar, I., Dogra, K., Utreja, C., & Yadav, P. (2018, April). A comparative study of supervised machine learning algorithms for stock market trend prediction. In *2018 Second International Conference on Inventive*

[7] *Communication and Computational Technologies (ICICCT)* (pp. 1003-1007). IEEE.

[8] Roy, S. S., Mittal, D., Basu, A., & Abraham, A. (2015). Stock market forecasting using LASSO linear regression model. In *Afro-European Conference for Industrial Advancement* (pp. 371-381). Springer, Cham.

[9] Ismail, Z., Yahya, A., & Shabri, A. (2009). Forecasting gold prices using multiple linear regression method. *American Journal of Applied Sciences*, *6*(8), 1509.

[10] Selvin, S., Vinayakumar, R., Gopalakrishnan, E. A., Menon, V. K., & Soman, K. P. (2017, September). Stock price prediction using LSTM, RNN and CNN-sliding window model. In *2017 international conference on advances in computing, communications and informatics (icacci)* (pp. 1643-1647). IEEE.

[11] Roondiwala, M., Patel, H., & Varma, S. (2017). Predicting stock prices using LSTM. *International Journal of Science and Research (IJSR)*, *6*(4), 1754-1756.

[12] Sunny, M. A. I., Maswood, M. M. S., & Alharbi, A. G. (2020, October). Deep learning-based stock price prediction using LSTM and bi-directional LSTM model. In *2020 2nd Novel Intelligent and Leading Emerging Sciences Conference (NILES)* (pp. 87-92). IEEE.

[13] Zhuge, Q., Xu, L., & Zhang, G. (2017). LSTM Neural Network with Emotional Analysis for prediction of stock price. *Engineering letters*, *25*(2).

[14] Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*(8), 1735-1780.