Comparison Between Logistic Regression, LSTM, and Random Forest in Chinese Stock Prediction

Shiming Zhou *

The Grainger College of Engineering, University of Illinois at Urbana-Champaign, Urbana 61803, USA

* Corresponding Author Email: shiming3@illinois.edu

Abstract. T Machine learning is transforming industries with its ability to derive insights and patterns within massive datasets. Among numerous algorithms available, certain foundational models stand out due to their efficiency and capability. This article compares three such models: linear models, decision trees, and neural network models. While all three has its unique pros and cons, this article aims to guide readers in choosing most fitting model with their tasks, by clarify the difference and future outlooks of these models.

Keywords: Stock Prediction, linear models, decision trees, neural network models.

1. Introduction

Stock trading has originated before the 1600s, but the oldest stock market was founded in 1602 when the Dutch East India Company was established [1, 2]. During 400 years of development, witnessing the rise and fall of countries and firms, stock markets played an important role between investors and investees, promoted economic developments, and experienced financial crises. Stock markets with complicated trading and investing characteristics are now the most important centers of modern businesses. Predicting stock market has its natural benefits. For economic entities, because the stock market is seen as an important factor reflecting the condition of the entity, it can help detect problems in the economy. For investors, it can help guide investments, manage risks, and allocate assets. The difficulty of stock market prediction has been revealed long before this century. Back in the 1930s, as indicated by Alfred Cowles 3rd in his article, despite the existence of professional forecasters and models with high sophistication, it was still unlikely that the stock market could be accurately predicted [3]. The major challenge is due to the randomness and complication of the stock market. The direct result of such challenge as revealed by Cowles was forecasters performing no better than random forecasters, that is, making decisions based on complete randomness. Reviewed and developed by Fama, Efficient Market Hypothesis (EMH) suggests the prices of assets can reflect information available on a certain level. This hypothesis gives support to the prediction of stock market using historical data, despite it arguing against profiting from the stock market [4].

Predicting stock prices or profits with computers is not a newly developed technique, in fact, such history can be traced back to the last century. The early examples of using neural to predict stock prices have begun around the 1990s. One of the earliest examples of such attempts available was conducted in 1990 by Eberhard Schöneburg, who constructed Adaline, Madaline, perception, and back-propagation neural networks to make predictions on the stock prices of three German stocks [5]. Schöneburg suggested that neural network has great potential for stock price prediction despite the fact that the training process was highly limited by computer performance back then.

With the development of computers and algorithms, approaches based on more sophisticated neural networks have been implemented and tested on bigger datasets. Chen et al. onducted their research on the stock return prediction with the LSTM model, using different levels of returns as categorical output [6]. They claimed to observe a significant improvement in the prediction accuracy compared with random prediction. The effectiveness is also supported by other researchers, e.g., Chen et al. have confirmed the effectiveness of deep learning on stock prediction tasks [7]. Other machine learning approaches are also applied in prediction. In 2016, Basak et al. have applied random forest on a binary prediction of rise or fall on stock prices [8]. In recent years, predictions based on more

sophisticated models were developed and tested. Ding et al. have used a transformer-based model for predictions on daily and 15-min scale data [9]. Other researchers trained their models on data with even higher frequency and corresponding turbulence. Some have applied different approaches. For example, Zhang et al. proposed a framework based on the inverse reinforcement learning method [10]. The researchers claimed their framework to promise profitability in the real stock market. The observation of forecasters being unable to outperform random methods made by Alfred has been no longer.

2. Data and Method

2.1. Data Preparation

The focus of this research is on the stock market in China mainland. There exist three Stock Exchanges in the mainland, the Shanghai Stock Exchange, the Shenzhen Stock Exchange, and the Beijing Stock Exchange. Beijing Stock Exchange started running in September 2021, indicating less than 2 years of data available, in comparison to the other two Exchanges, both started running in 1990. On the other hand, the Beijing Stock Exchange have less than 200 stocks in the market by the time the research is conducted, much less than the over 2000 stocks in Shanghai and Shenzhen Stock Exchange, it is considerably less important and representative for the research. Therefore, the stocks in the Beijing Stock Exchange were removed from both the training and testing datasets. Stocks with less than 1 year of running time were removed because of the general instability. Stocks with damaged data are also removed from the datasets. The models are desired to predict stock daily close percentage increases for specific stocks, namely, the relative difference of close price between two consecutive days. Because of data availability, accessibility, and relativity, the daily open, close, high, low price, volume, and amount are fetched to make basic predictions as a verification of the effectiveness. The dataset for each stock is collected from stock in the market until June 7^{th,} 2023. The data including volume and amount are processed to become 30-day clips. Then, the clips are processed through a min-max scaler for normalization to gain effectiveness. The scaler is applied after clipping to prevent the effect of future data in terms of the predicted day, and the effect of historical data. If all historical data in terms of the predicted day is taken to be processed with a min-max scaler, the price might be scaled to a position close to 1 because of inflation over many years. 220000 random 30-day clips with potential overlaps but no replacement was drawn from all the clips. The output desired from the model is categorical data. The daily close increases fall evenly in 7 intervals. On the level of distribution, training and testing dataset follows an 8:2 pattern, with 80% of stocks distributed into the training set.

2.2. Models

The independent variables, as the input of the models, are all six characteristics introduced in the data preparation section. The dependent variables, as the output of the model, are the prices of the stock on the next day of the input. Data is loaded to the model in shuffled order, the level of shuffling is on the level of stock. The implementation is based on PyTorch, and the models' details are described. The logistic regression model is implemented with a single linear layer with an input size of 180, that is the 6 features on 30 days, and an output size of 7. The dynamic LSTM model is implemented with an LSTM connected to a linear layer with a dropout rate of 0.3. The LSTM takes in 6 characteristics and outputs a tensor of size 64, and it takes a 30-day length on the time dimension, the layer number is 2. The FC layer has an input size of 64 and an output size of 4. Both of the models are trained with an Adam optimizer with a 0.01 learning rate. MSE loss was taken as the loss function for both models. The result of the logistic regression model and LSTM model is evaluated with average cross-entropy loss in the testing dataset. Training ends after 10 epochs of training. The random forest is implemented with Python using sklearn module. There are 100 decision trees in the random forest.

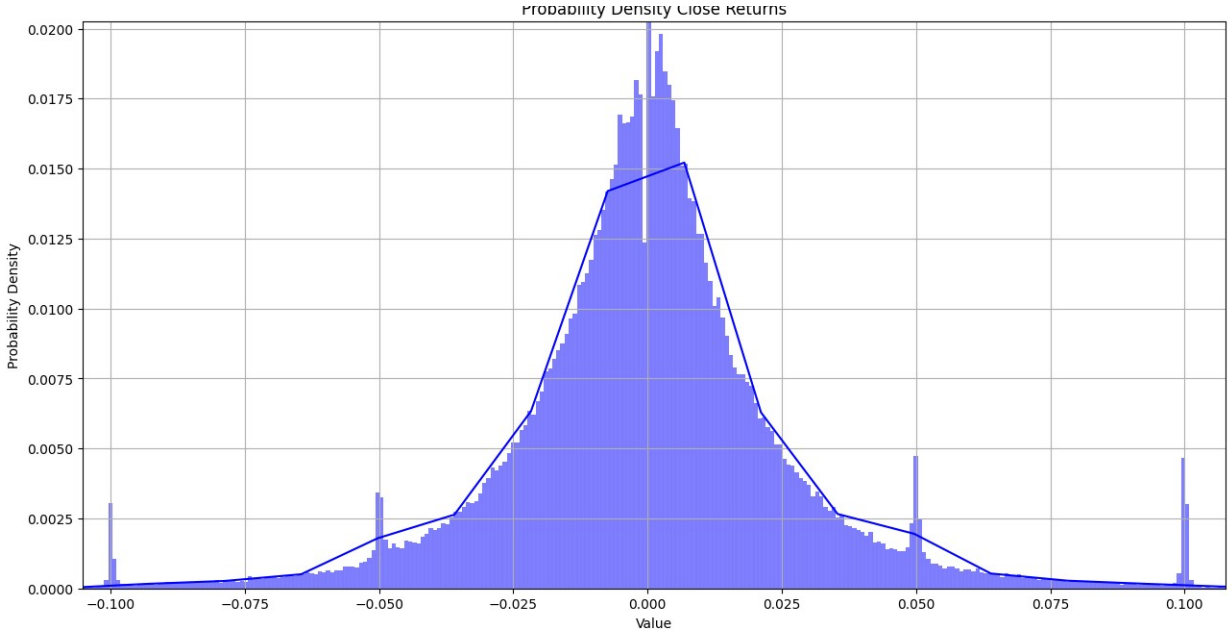


Figure 1. Probability Density of Close Percentage Increase (Photo/Picture credit: Original).

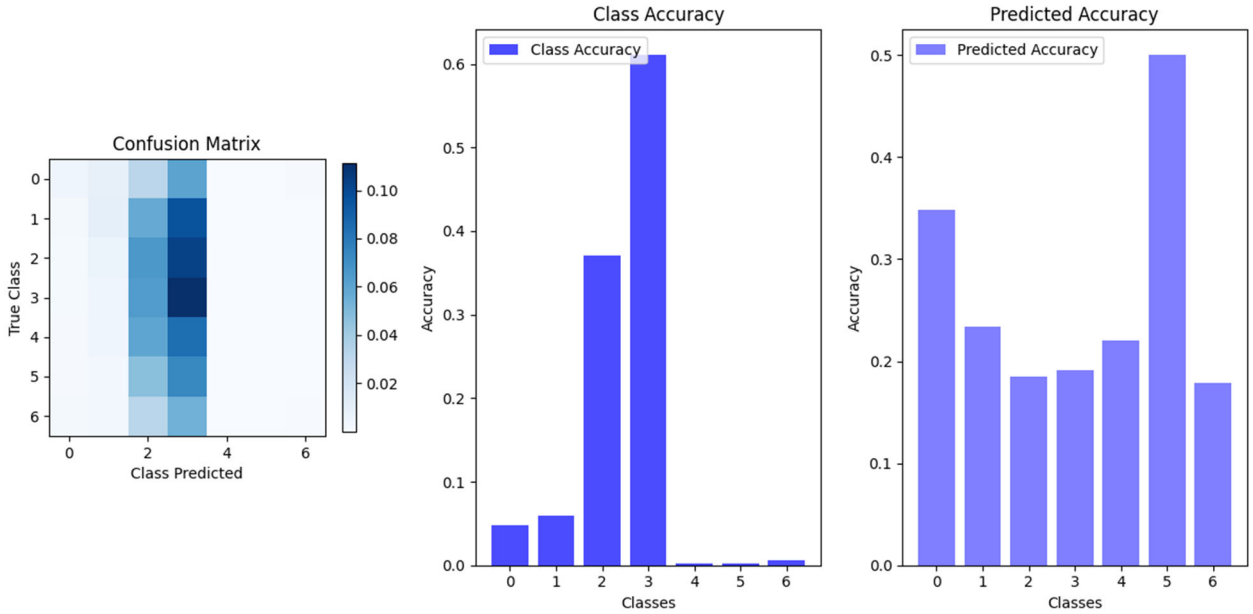
3. Results

The daily close increases are counted and computed, and thus the intervals are set to be (, -0.0225], [-0.0225, -0.0101], [-0.0101, -0.00255], [-0.00255, 0.00310], [0.00310. 0.0104], [0.0104, 0.0231], [0.0230,), here presented rounding to 3 significant figures. Because over 1 million daily increases were counted, a sample of 6000 increases was randomly selected without replacement as a representation of the whole. The progressive density curve of the results is presented in the figure below. It might be assumed that the distribution of increases follows a roughly normal distribution. The major difference between the normal distribution and that of the close increases is that the increases have peaks at ±0.05 and ±0.1, while the latter is the mechanism limit up and limit down of the market.

The running time of the model differs significantly. The shortest time cost is the logistic regression approach, taking approximately 0.5 minutes on model training and evaluating, the random forest takes approximately 5.7 minutes, and the LSTM takes the longest 20.4 minutes. The accuracy, precision, recall, and F1 score are shown as in Table 1.

Table 1. Accuracy, Precision, Recall, and F1 score of the models

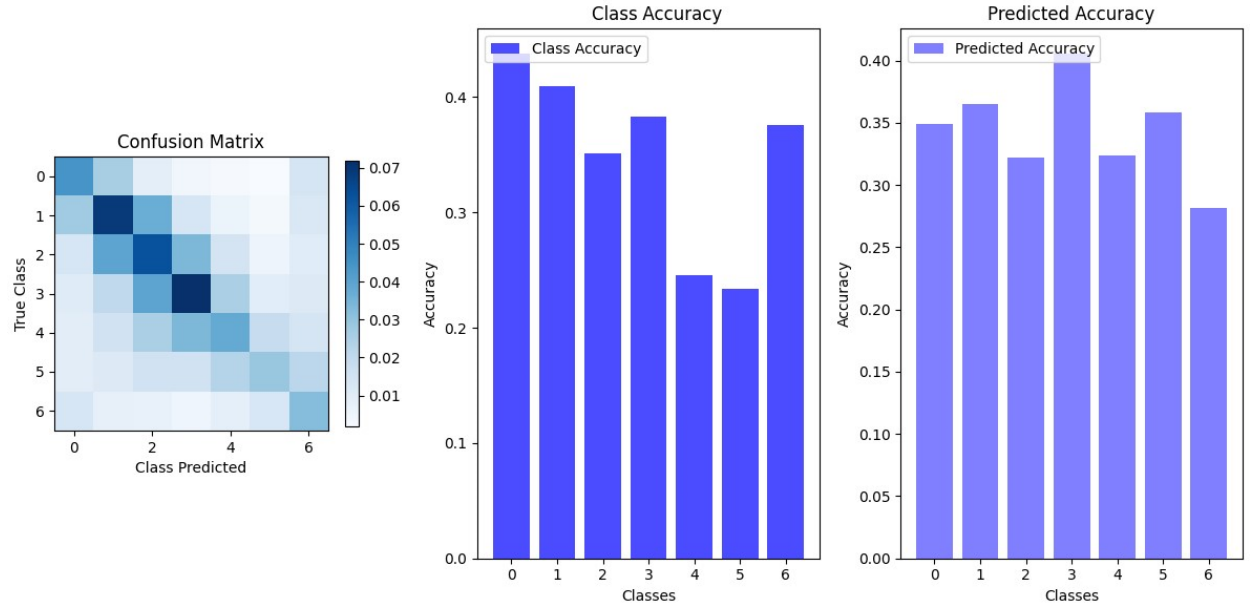
Model	Accuracy	Precision	Recall	F1 Score
Logistic Regression	0.194	0.242	0.158	0.107
LSTM	0.347	0.344	0.348	0.341
Random Forest	0.341	0.355	0.341	0.335



(a) (b) (c)

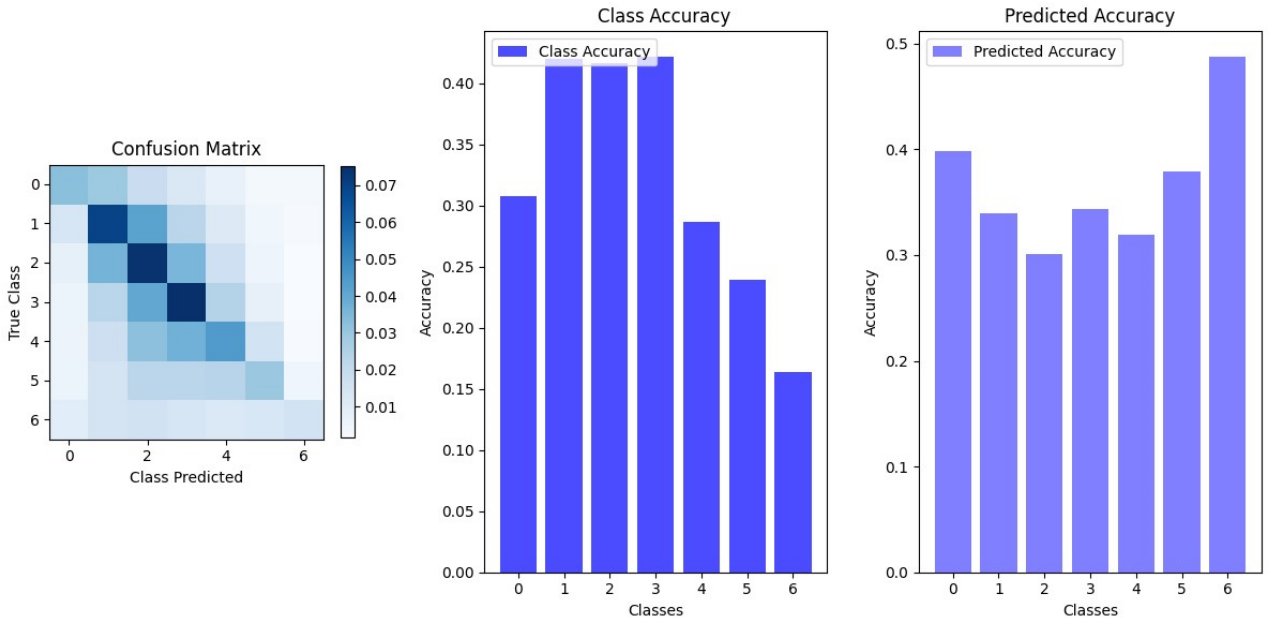
Figure 2. Visualized confusion matrix of the Logistic Regression model, its class accuracy, and predicted accuracy (Photo/Picture credit: Original).

As it is shown in Figure 2(a), the logistic regression model converges to predict almost every class into class 2 and class 3, that is, it predicted almost every close increase to fall between the [-0.0101, 0.00310] interval. Despite the accuracy of the model being higher than the expected accuracy of a random classifier, it is potentially not better than the random classifies considering the behavioral pattern. The LSTM model is shown to predict as the expectations in comparison to the logistic regression. As shown in Figure 3(a), the LSTM made a relatively more accurate prediction in comparison to the logistic regression model, and it does not show the same pattern as the former. As Figure 4 reveals, the random forest also predicts better than the logistic regression without the pattern of predicting every class to class 2 or class 3, and the general performance of the random forest is similar to that of the LSTM.



(a) (b) (c)

Figure 3. Visualized confusion matrix of the LSTM, its class accuracy, and predicted accuracy (Photo/Picture credit: Original).



(a) (b) (c)

Figure 4. Visualized confusion matrix of the random forest, its class accuracy, and predicted accuracy (Photo/Picture credit: Original).

4. Conclusion

In conclusion, LSTM showed significant improvements in comparison to logistic regression, which conforms to the common sense and reason for applying RNN to time series tasks. However, an LSTM is not performing significantly better random forest is not under the implementation of this research. Another potential approach might be implemented using a dynamic LSTM, taking the length of arbitrary running days as input. However, the data inputs have different lengths on the dimension of time series, therefore, the training might do unpadded and therefore non-batched. Results might be different using different batching and padding methods. Since the categories are ranked in their nature, a regression approach with mean squared error as its loss function might be preferred. Because this research aims to study the effectiveness of the models' ability to predict the close return of the stock on the following day given historical data, the LSTM approach took 30-day clips as input. While the min-max scaler might reflect the price level of the stock in history, the model might gain more information from applying the min-max scaler on all historical days before taking the clip, despite the drawbacks discussed in the model construction. The problem might be solved by dividing the price with a price indicator or adding the indicator as input to the model. For example, the Shanghai Securities Composite Index, or the Consumer Price Index (CPI).

In addition, as it might be observed from the confusion matrix of random forest, the performance is similar to a random predictor on the increases with true value lies in the interval. Further research might be conducted for examination. Due to time and resource reasons, are restricted on the following aspects. This research took daily stock data because of the inaccessibility, incompleteness, and irreliability of historical five-minute bar data, minute bar data, or higher frequency data, the result reviewed by this research may not be meaningful on other frequencies. Despite this result presented is approached with hyperparameters took the optimal from a few combinations, it is not based on highly explored state space of hyperparameters because of the lack of computational resources, and it does not represent the best termination of this approach. Therefore, the conclusions made by this research may not be entirely accurate, and conclusions drawn with the same approach but with other combinations might be different. The research might be improved by exploring previously mentioned probabilities.

To sum up, this research focused on the comparison of logistic regression, LSTM, and random forest on stock increase prediction. The result revealed the relatively poor performance of logistic regression and similar but better performance of LSTM and random forest under the implementation of this research. Due to resource restrictions, the implementation of LSTM might be improved, and one may reasonably make additional neural network attempts.

References

[1] F. Braudel and S. Reynolds, The wheels of Commerce (University of California Press, Los Angele, 1992) pp. 1 - 221.

[2] E. Chancellor, Devil take the hindmost: A history of financial speculation, (Plume, New York, 2005) pp. 10 - 24.

[3] A. Cowles, Econometrica 1(3), 309 (1933)

[4] E. F. Fama, The Journal of Finance 25 (2), 383 (1970).

[5] E. Schöneburg, Neurocomputing 2 (1), 17 – 27 (1990).

[6] K. Chen, Y. Zhou and F. Dai, "A LSTM-based method for stock returns prediction: A case study of china stock market," 2015 IEEE International Conference on Big Data (Big Data) p. 145.

[7] L. Chen, Z. Qiao, M. Wang, et al., IEEE Access 6, 48625 – 48633 (2018).

[8] S. Basak, S. Kar, S. Saha, L. Khaidem and S. R. Dey, The North American Journal of Economics and Finance 47, 552 – 567 (2019).

[9] Q. Ding, S. Wu, H. Sun, and J. Guo, "Hierarchical multi-scale gaussian transformer for stock movement prediction," Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence (2020) p. 17.

[10] W. Zhang, T. Yin, Y. Zhao, B. Han and H. Liu, IEEE Access 11, 14115 – 14127 (2023).