Stock Price Prediction Using Machine Learning

By: Yixin Guo

Södertörn University | School of Social ScienceMaster

Dissertation 30hp

Economics Spring 2022

SÖDERTÖRN UNIVERSITY
STOCKHOLM

Abstract:

Accurate prediction of stock prices plays an increasingly prominent role in the stock market where returns and risks fluctuate wildly, and both financial institutions and regulatory authorities have paid sufficient attention to it. As a method of asset allocation, stocks have always been favored by investors because of their high returns. The research on stock price prediction has never stopped. In the early days, many economists tried to predict stock prices. Later, with the in-depth research of mathematical theory and the vigorous development of computer technology, people have found that the establishment of mathematical models can be very good, such as time series model, because its model is relatively simple and the forecasting effect is better. Time series model is applied in a period of time The scope gradually expanded. However, due to the non-linearity of stock data, some machine learning methods, such as support vector machines. Later, with the development of deep learning, some such as RNN, LSTM neural Networks, they can not only process non-linear data, but also retain memory for the sequence and retain useful information, which is positive. It is required for stock data forecasting. This article introduces the theoretical knowledge of time series model and LSTM neural network, and select real stocks in the stock market, perform modeling analysis and predict stock prices, and then use the root mean square error to compare the prediction results of several models. Since the time series model cannot make good use of the non-linear part of the stock data, can't perform long-term memory, and LSTM neural network makes better use of non-linear data and has better use of sequence data. Useful information in the long-term memory, which makes the root mean square error of the prediction result, the LSTM neural network needs smaller than the time series model, indicating that LSTM neural network is a better stock price forecasting method.

The time series for stock prices belong to non-stationary and non-linear data, making the prediction of future price trends extremely challenging. In order to learn the long-term dependence of stock prices, deep learning methods such as the LSTM method are used to obtain longer data dependence and overall change patterns of stocks. This thesis uses 5000 observations from S&P500 index for empirical research, and introduce benchmark models, such as ARIMA, GARCH and other research methods for comparison, to verify the effectiveness and advantages of deep learning methods.

Table of Contents

1. Introduction..1
2. Literature review..6
   2.1 Progress of stock price prediction..8
   2.2 Time series model..12
   2.3 Deep learning..13
3. Methodology..19
   3.1 Data source..20
   3.2 Methodology..20
4. Results and discussion..24
   4.1 Arima process..24
   4.2 GARCH process..25
   4.3 LSTM process..26
   4.4 Mixed model process..27
5. Conclusion and Implication..30
   5.1 Conclusion..30
   5.2 Implication..31
Reference..32
Appendix..34

1. Introduction

As a high-risk and high-return market, the stock market has always been closely watched by investors(Daubechies, I. 1992), and stock forecasting has always been a research topic of great concern to researchers. In addition, the stock market is an important part of my country's financial market, it reflects the operation of the national economy, and the operation of the stock market has an important impact on the operation of the national economy. Although the issue of predictability of stocks has always been controversial, the study of stock forecasts still helps us understand the laws of some market changes and development(Fama, E. F., & Blume, M. F. (1966)). With the advancement of science and technology, a large amount of financial data has been retained(Fama, E. F., & French, K. R. (1988)), providing a solid data foundation for the analysis of the stock market; at the same time, the continuous development and updating of algorithms has provided a powerful tool for people to analyze the stock market(Faria, G., & Verona, F. (2018), Ferreira, M. I., & Santa-Clara, P. (2011), Gençay, R., Selçuk, F., & Whitcher, B. (2002)).

As an important part of a country's economy, the stock market provides a financing and investment environment for the country's companies and investors. Predicting the future performance of the stock market can not only provide investors with investment advice, but also help companies formulate financing plans, thereby promoting the healthy development of the economy(Campbell, J. Y., & Thompson, S. B. (2008)). At the same time, establishing a stable investment portfolio based on the forecast results combined with the portfolio theory can help investors to further improve their investment returns. Therefore, it is a very meaningful problem for stock market forecasting and investment portfolio method research.

With the beginning of reform and opening up in 1978, real economy is advancing all the way and developing rapidly, which also makes the financial industry is booming. Investors pay more and more attention to the allocation of financial assets. In addition to savings and debt, relatively traditional investment and financial management methods such as securities, stocks, as a new method of asset allocation, have gradually become a new key for investors. The first stock in human history, the stock started in 1611 and was created in Amsterdam, the Netherlands. The subject of the transaction is the East India Company in the Netherlands, which was established in 1602 (Campbell, J. Y., & Vuolteenaho, T. (2004)).

Investors usually adjust the allocation of investment assets to reduce their own decision-making risks, this makes it very important for investors to predict the price of stocks or other financial assets. It is a challenging problem to accurately predict when and how to allocate asset budgets at that time, because there are many factors that can affect stock prices, such as the company's asset allocation or operating conditions, the impact of economic and political policies in related industries, and the occurrence of emergencies and currency exchange rates, etc.(Jiang, F., Lee, J. A., Martin, X., & Zhou, G. (2019)). Therefore, many investors have used technology and quantitative methods to try to predict the fluctuation of asset prices. These methods include finding a relatively suitable model from historical market data and at the same time finding the best time for accurate investment decisions. The issue of whether the stock market trend is predictable has been controversial for decades.

Stock data is a classic time series. Many researchers have used time series models for forecasting, such as ARIMA or GARCH models, but the assumptions of classic time series models are relatively high. For example, the series needs to be stable and linear. However, there are many factors that affect the stock price of stock data, which makes the stock data itself not stable and linear. Although the difference method can be used to smooth the sequence, the difference operation also causes data loss, which makes the traditional time series model have greater limitations in forecasting. With the development of computer science and artificial intelligence, more and more researchers choose machine learning models for prediction, such as support vector machines, perceptron models, etc. Because they can handle nonlinear data, especially support vector machines. This model has a non-linear kernel function, so it has been used by the majority of people in the industry for a period of time.

The stock market not only reflects the development of the country's economy, but also provides a basis for the country to formulate the next economic policy. The research on the stock market has a long history. At present, the representative stock investment theories include: random walk theory, modern portfolio theory, efficient market hypothesis, behavioral finance and evolutionary securities and so on. Among them, many empirical studies show that the efficient market hypothesis does not hold in emerging markets such as China. Therefore, many researchers began to use statistical models, such as differential integrated moving average autoregression (ARIMA), generalized autoregressive conditional heteroskedasticity (GARCH) and other models to predict stock prices and obtained some better prediction results. This has led many researchers to develop various statistical models to predict future changes in stock prices. However, because the statistical model itself has many assumptions, and many of these assumptions are not satisfied in practical applications, it has been difficult for statistical models to achieve good results. Subsequently, the wide application of the classic machine learning model has led many scholars to apply the model to stock price forecasting, and at the same time compare the traditional statistical model. The classical machine learning model avoids many assumptions of the statistical model and has efficient nonlinear learning ability, which makes the performance of the model much better than the statistical model in stock price prediction. Subsequently, people began to utilize classical machine learning models to further improve the out-of-sample performance of stock price prediction.

With the development of deep learning neural networks, people have gradually realized that neural networks can be used as a new predictive method: First, neural networks have low data requirements and do not require strict assumptions; at the same time, it can also choose non-linear activation. The function converts the linear mapping into a nonlinear mapping, and then through the processing of the hidden layer, it further enhances its ability to process nonlinear data. However, the general neural network does not make much use of the time sequence. Each network layer is performing calculations at the same time, ignoring the time sequence of the data. Therefore, the Recurrent Neural Network (RNN) was born. The connection of the same layer completes the task of extracting data sequence. Of course, RNN also has its own shortcomings. For example, if there are too many hidden layers, RNN will not have too much memory for information from a long time ago. Therefore, in 1997, Hochreiter and Schmidhuber proposed a long and short-term memory neural network model and introduced the concept of "gate", which solved this problem well.

With the rapid development of computer information technology, electronic trading is becoming more and more mature, making it possible to process massive high-frequency trading data. Due to the rapid increase in the amount of data that can be processed and the substantial enhancement of computer hardware, deep learning technology stands out from many machine learning models and performs significantly better than classical machine learning models. In recent years, deep learning technology has been successfully applied to fields such as natural language processing, speech recognition and image processing(Addison, P. S. (2002), Avramov, D. (2002), DeMiguel, V., Garlappi, L., Nogales, F. J., & Uppal, R. (2009)). As the core model of deep learning technology, the deep neural network model has attracted the attention of scholars and has become a research hotspot in recent years. Moreover, the model is completely data-driven and does not require preconditions, and at the same time has efficient nonlinear learning capabilities, which makes the model much better than the classical machine learning model. Therefore, some scholars have turned to various neural network models to predict the rise and fall of stock prices, and have achieved good results. At the same time, the model is also widely used in other financial fields, such as quantitative stock selection, algorithmic trading, high-frequency trading, etc. Then, continuing to study how to apply this model to stock market forecasting more efficiently is of great significance to both investors and researchers.

Thus, in this paper, we will introduce a mixed approach, combing traditional time series method and deep learning method to predict stock prices. The innovation of the article emphasizes the long-term dependence of LSTM on performance to improve accuracy, and the mixed approach can improve the robustness of the model.

Stock market forecasting is the act of trying to determine the future value of shares of companies listed on exchanges or other investment targets. The stock market has multi-scale properties. The so-called multi-scale refers to the existence of multiple data at different time intervals in the stock market. For example, taking stock price data as an example, there are not only short-term real-time price data per second, every minute, and hour, but also daily, weekly and even real-time price data. Monthly mid-term average stock price data, these prices may have different patterns of change at different scales. Stock market forecasting research is usually based on stock market data under a certain scale. After analysis, some patterns that recur in the data are extracted under a certain scale, so as to predict the movement trend of the stock market under this scale. Although data at different scales may have different changing laws, there is also a close interaction between them. If the data at different scales can be considered comprehensively, the state of the stock market can be described more accurately, and thus better forecasting the stock market. Traditionally, stock market forecasts need to be given by stock market researchers with deep knowledge and extensive analytical experience. They make predictions on the future development direction of the stock market and the degree of fluctuations based on multi-source heterogeneous data such as foreign exchange, policies, events, and stock prices in the global economic data.

Stock data is a classic time series, and many researchers have used time series models for forecasting, such as ARIMA or GARCH models(Goyal, A., & Welch, I. ,2003, Goyal, A., & Welch, I. 2008, Brock, W., Lakonishok, J., & LeBaron, B. 1992), but the assumptions of classic time series models are relatively high, such as the need for the series to be stationary and linear. However, the factors that affect the stock price of stock data come from many aspects, which makes the stock data itself not stable and linear. Although the difference method can be used to make the sequence stationary, the difference operation also causes data loss. , which makes the traditional time series model have great limitations in forecasting. With the development of computer science and artificial intelligence, more and more researchers choose machine learning models for prediction, such as support vector machines, perceptron models, etc., because they can deal with nonlinear data, especially support vector machines , the model has a nonlinear kernel function, so it has been used by the majority of people in the industry for a period of time.

The structure of this paper is described as follows. The first chapter is the introduction, which mainly introduces the research background, research significance, research development, and the main research content. The second chapter is the literature review including the traditional time series methods and the theoretical basis of the deep learning methods. Chapter 3 introduces data sources, preprocessing methods and model construction, as well as parameter optimization methods and evaluation guidelines. Chapter 4 is the case analysis, modeling according to the steps of the previous chapter, and making the forecast of the stock price sequence. Then, this paper compares the advantages and disadvantages of each model. Finally, the chapter 5 is the conclusion and implication of the paper.

2. Literature review

Research in finance has explored how stock markets are affected by their multi-source and heterogeneous data on some scales. Multi-source heterogeneous datain the stock market means that the data of the stock market includes data from different sources such as the stock market, the foreign exchange market and even the weather system, as well as the structure of stock prices, trading volumes, and stock news, announcements and social networks. and other unstructured data. In particular, the efficient market hypothesis believes that information from various sources in the stock market will have an impact on the stock market, while behavioral financebelieves that financial markets are explained, studied and predicted from the individual behaviors of traders and the motivations that produce such behaviors. the trend and extent of price fluctuations. These studies point out that the internal mechanism of the stock market is very complex, similar to Brownian motion. Combining the multi-source heterogeneous data in the stock market can more accurately classify and predict the stock market state. With the vigorous development of the stock market, it continues to generate a large number of multi-source heterogeneous data of various scales. The traditional idea of relying solely on experts to analyze and predict has been difficult to meet the needs of industry development(Guo, H. 2006 ,Haven, E., Liu, X., & Shen, L. 2012). In order to quickly analyze massive stock market data and assist or even completely replace investors in making stock market investment decisions, a large number of researches on stock market forecasting based on information technology have emerged. These studies have also contributed to the rapid development of quantitative funds that rely on automated computer analysis to execute and even make investment decisions entirely on their own(Chen, J., Jiang, F., & Tong, G. 2017).

Obtaining accurate stock price forecasts can more effectively avoid future risks for decision makers; for regulators, it can strengthen the control of the stock market, regulate and guide the stock market in a timely manner, and contribute to the sustainable development of the economy. Development provides firm confidence and strong guarantees.

The so-called stock price forecast is to use various scientific methods to predict the development prospects of the stock market through the regularity of the development of the stock market and its history and status, relying on a large amount of stock market information and accurate statistical survey data(Dangl, T., & Halling, M. 2012). For decades, scholars have explored various forecasting methods. Therefore, reading about relevant research and summarizing and classifying these forecasting methods has certain positive significance for further research.

Stock data is a classic time series, and many researchers have used time series models for forecasting, such as ARIMA or GARCH models,(Inoue, A., & Kilian, L. 2004,Jaffard, S., Meyer, Y., & Ryan, R. D. 2001), but the assumptions of classic time series models are relatively high, such as the need for the series to be stationary and linear. However, the factors that affect the stock price of stock data come from many aspects, which makes the stock data itself not stable and linear. Although the difference method can be used to make the sequence stationary, the difference operation also causes data loss. , which makes the traditional time series model have great limitations in forecasting. With the development of computer science and artificial intelligence, more and more researchers choose machine learning models for prediction, such as support vector machines, perceptron models, etc., because they can deal with nonlinear data, especially support vector machines , the model has a nonlinear kernel function, so it has been used by the majority of people in the industry for a period of time.

Predicting stock prices or other financial asset prices is very important to investors because investors usually reduce their decision-making risk by adjusting the allocation of investment assets. It is a very challenging problem to accurately predict when and how to allocate the asset budget at that time, because there are many factors that can affect the stock price, such as the company's asset allocation or operating conditions, and the impact of economic and political policies in related industries. , the occurrence of emergencies and the exchange rate of currencies, etc. Therefore, many investors have used technical and quantitative methods to try to predict the volatility of asset prices. These methods include finding relatively suitable patterns from historical market data, as well as pinpointing the best time to make investment decisions. The question of whether the stock market is predictable has been debated for decades, and there is still no conclusion.

2.1 Progress of stock price prediction

The research on stock behavior was first conducted by Bachelier in 1900. He used random walks to express stock price trends. Fama tested that stock price changes are characterized by random walks. Malkiel and Fama studied valid market assumptions in 1970 and found that all new information will be reflected in asset prices immediately without delay. Therefore, changes in future asset prices have nothing to do with past and present information. From their perspective, predicting future asset prices is considered impossible. On the other hand, many studies try to prove effective market hypotheses experimentally, and empirical evidence shows that the stock market can be predictable in some ways. In traditional time series models, parameter statistical models are used for forecasting, such as ARMA model, ARIMA model and vector autoregressive model, etc., to find the best estimate. Virtanen and Yliolli used six explanatory variables to estimate the Finnish stock market index, including the lagging index and macroeconomic factors in an econometric model based on ARIMA. Work(Clark, T. E., & West, K. D. 2007) proposed a stock price prediction system based on ARIMA in 2014, which has been tested in the listed stocks originated from the Stock Exchange in New York and the Stock Exchange running from the country Nigeria. Then the ARIMA model is regarded as a high potential model for forecasting short-term series.

Although econometric models mentioned above can easily describe and evaluate the relationship between large amount of variables through inference in the view of statistical, however these methods still have owned limitations for time series analysis in domain of finance. Firstly, they assume that the model structure is linear, and they cannot capture the non-linear nature of stock prices. In addition, these models all assume that the data as a constant value, although the actual time series for finance are full of noise and have time-varying oscillation. Because of its ability in nonlinear mapping and induction, it has been widely used. Many experts try to model financial time nonlinear models, such as multi-layer neural networks and support vector machines(SVM) with nonlinear kernel functions. They are differences from traditional economic models. Neural networks lack of a strict model structure and a series of apparent assumptions. As long as there is enough data, it can be modeled. Work from proposed two mixed models to predict, combining ANN with exponential generalized ARIMA, and later predicted the volatility for S&P500 index return for the year 2012. Their calculation results show that the mixed model has lower test errors and its performance is better than the non-mixed single model. Kristjanpoller et al. merged the generalized autoregressive conditional heteroscedasticity model (GARCH) and ANN in 2014, and proposed a prediction model for the volatility in the Latin American market, and showed that this model is superior to the GARCH model (its MSE is smaller). Work(Cochrane, J. H. 2007) from proposed a hybrid model of neural networks, random forest and support vector regression (SVR) in 2015 to predict the Indian stock market. Agarwal and Sastry combined the RNN neural network into two kinds of linearization models with ARMA and exponential smoothing function in 2015, and predicted stock returns. The experimental results show that the predictability has been greatly improved, and the improvement is mainly contributed by the RNN neural network.

For recent studies, LSTM neural networks that are properly built to learn temporal module have been widely used in various tasks of time series analysis. The reason why LSTM is advanced than traditional RNN is that it solves the problem that RNN neural network fails to solve, that is, the problem of gradient explosion and gradient disappearance, and it can learn effectively through storage units and "gates", and is useful for information for long-term memory. Therefore, many experts have used LSTM to conduct a lot of research on financial time series modeling. In experiments, LSTM is superior to support vector machines due to the addition of emotional features, so that the accuracy of predicting the opening price of the next day has been significantly improved (from 78.57% to 87.86%). The work from Dai, Z. F., Dong, X. D., Kang, J., & Hong, L. (2020b) used the textual data from the newspaper at Nikkei as the input of the LSTM neural network, and combined with the time series data in stock market to predict the opening price of 10 selected companies. A trading strategy based on the predicted results is simulated. The experimental results show that the model has a higher profit value than the trained model only with stock data.

When using deep neural networks (DNN) for financial time series analysis, researchers are more concerned about the problem of overfitting. Within a year, we can collect only about 252 data points per day. DNN has a good representation ability, because they learn the highly complex nonlinear relationship between variables, so the model has a high accuracy on the training set, but this makes the model prone to overfitting. In order to improve the generalization ability of the model, many researchers have conducted research on regularization methods such as L1 and L2 regularization, Dropout, early stopping, and reducing learning rate. These methods can avoid the problem of overfitting. For artificial neural networks, reducing the size of the neural network can also prevent overfitting, but since larger and deeper networks can solve more complex problems, there must be enough data to use deeper and larger networks. Therefore, the data enhancement method becomes a two-pronged method that can reduce the degree of over-fitting and improve the accuracy of generalization at the same time. However, this method is widely used in image processing problems. Unlike image data, data enhancement of financial time series is not a simple matter. Image data enhancement can be achieved through a variety of transformation techniques. For example, transformation-based data expansion methods will distort the original data to generate new composites. Therefore, although data enhancement of financial time series is of great significance in improving the performance and robustness of deep learning models, it has limited academic attention.

If the dimensionality of the input data is reduced and used as input in a neural network, information loss may occur. An important advantage of deep learning is that it can learn features from the input data itself. However, for RNN neural networks, as the number of network layers continues to increase, the earlier the input data, its influence on the output results will be more and more weakened due to the increase in sequence length, which leads to the RNN neural network Long-term memory of information is weak. Hochreiter and Schmidhuber in their 1997 paper "Long Short-Term Memory", they proposed the LSTM neural network for the RNN neural network's inability to solve the long-order dependence of the time series. They introduced the "gate" in the LSTM neural network. concept. Felix et al. optimized the LSTM neural network in 2000. Considering that the memory storage unit of the neural network will increase with the increase of the sequence length, which may cause the network to collapse, they proposed to add a forgetting gate inside the LSTM neuron. At this point, the prototype of the LSTM neural network has been completed. The LSTM neural network can control the transmission of input data through input gates, forget gates and output gates, and maintain the independence of the output of the memory storage unit and the result output, so that the sequence can retain important information during transmission and maintain it for a longer period of time memory. Therefore, the application of LSTM neural network in the prediction of financial time series has become more and more extensive.

2.2 Time series model

2.2.1 Stationary time series

Stationary time series are divided into strictly stationary time series and wide stationary time series. Below we introduce their definitions. Strictly stationary time series provide important theoretical significance, but it is difficult to obtain the joint distribution of random sequences in the actual research process. Therefore, in order to better use in practical applications, researchers have defined a relatively weak wide stationary time sequence. Researchers choose to use the characteristic statistics of the sequence to define wide stationarity, which can make the constraint conditions a little looser. By ensuring the stationarity of the low-order moments of the sequence to ensure that the sequence can be approximately stationary.

Time series analysis also belongs to the field of statistics. It can also analyze the population through samples like statistics. And from the statistical theorems, we can know that the number of random variables is directly proportional to the complexity of the analysis, and the sample size is inversely proportional to the accuracy of obtaining the overall information (obviously the sample information obtained when the population is selected as the sample is Overall information, but such an operation is obviously unrealistic). But time series data has its peculiarities. For a time series {⋯, X1, X2, · · ·, Xt, · · · }, its value Xt at any time t is a random variable, and since time is one-way, it cannot be repeated , So we can only get one sample value in this way, which leads to too little sample information for statistical analysis. But if we have the concept of stationarity, this problem will be solved.

2.2.2 Principles of the ARMA model

Autoregressive moving average model (ARMA model) is the most commonly used time series model. According to different conditions, it can be divided into the following three types of models: autoregressive (AR) model, moving average (MA) model and ARMA model.

Most of the data in real life is not stable. We need to smooth the data. Box and Jenkins proved that the difference method is an effective smoothing method. Therefore, applying the difference method to the ARMA model will result in the well-known ARIMA model. The following article will introduce the general modelingsteps of the ARIMA model.

(1). Sequence stationarity test: First, as with any data analysis, we should draw a time series graph and check whether the image has an obvious trend; then, we also need to draw a correlation graph, by observing whether the ACF image is rapidly reduced to 0 is used to judge the stationarity; finally, if the data does not have stationarity, then the difference method is needed to smooth the data.

(2). Determine the order of the model: After an appropriate degree of difference (d), we then need to determine the order p, q of the model, usually using ACF diagram and PACF diagram to determine the order, in order to make the result more accurate Using Akaike Information Criterion (AIC) or Bayesian Information Criterion (BIC), the smaller the value of AIC and BIC, the more correct the choice of model order.

(3). Model checking is mainly divided into two parts. The first part is the significance test of the model. It is usually chosen to test whether the residuals belong to the white noise sequence. The second part is the parameter test. We choose the quotient of the estimated coefficient and its standard deviation as the test statistic, which is usually compared with the critical value of the T statistic significance of 5% (that is, 1.96). If the absolute value of the test statistic is greater than 1.96, the null hypothesis is rejected, indicating that the coefficient is not significant, otherwise, the coefficient is significant and the model meets the requirements.

2.3 Deep learning

Although the various forecasting methods mentioned above can achieve a rough forecast of future stock price changes, they cannot achieve accurate forecasts. This is because my country's stock market is still in an immature state of development, and many factors such as national economic conditions, macroeconomic policies, and investors' psychological expectations in the short term will affect stock prices to some extent. Therefore, in future forecasts, various factors should also be considered comprehensively, such as the fundamentals of the operating enterprise, technical indicators, etc., in order to achieve the investment goal of maximum profit or avoidance of maximum risk.

The traditional methods to forecast stock include qualitative econometric methods and machine learning methods. The stock price series can be regarded as complex and time series with much nonlinearity, so using qualitative econometric models cannot achieve the higher forecasting ability. In the machine learning algorithm, due to the unique structure and learning mechanism of neural network, domestic and foreign scholars have gradually increased the research on using it to predict stock prices and trends. In recent years, with the continuous development of deep learning, deep neural network has gradually been applied to the fields of image, speech and finance. It can extract high-level abstract features from a large amount of original data without relying on prior knowledge, and has stronger learning ability and generalization ability. Especially the LSTM neural network, which is a kind of cyclic neural network in the deep learning algorithm, has a special gate structure, and has the characteristics of good selectivity, memory and internal influence of time series, which can process financial data sequences more effectively. Stock forecasts offer new ideas. This article attempts to use LSTM neural network for stock price prediction.

The essence of the BP neural network algorithm is the error gradient descent method. The core idea is: First, the input signal of the learning sample (normalization operation is usually performed) is sent to the input layer, and then passed to the output layer through the hidden layer, and after the calculation of the output layer, the corresponding predicted value is output. When the error between the predicted value and the true value (expected value) does not meet the preset target accuracy requirements, the network will feed back the error information from the output layer to the input layer, and adjust the weights and thresholds between each layer. Repeated loop iterations gradually reduce the error between the output value of the network and the expected output value of the sample until the set number of cycles or accuracy requirements are met. At this time, the learning process of the network ends, and the optimized weights and thresholds are obtained (Intrinsic relationship), and then based on the intrinsic relationship, extract the input information of the unknown sample to obtain the mapping (prediction) of the unknown sample(Conrad, J., & Kaul, G. (1998), Cowles, A., 3rd (1933), Dai, Z., Zhou, H., Wen, F., & He, S. (2020a)).

2.3.1 Fully connected network

Fully-connected network (fully-connected network, or feedforward network) is a kind of non- Linear model. It adds nonlinear functions (such as tanh, sigmod, ReLU, etc.) after the linear transformation and realize the function of non-linear function.

Figure 1 shows a simplest fully connected network structure, including input layer, hidden layer and output layer: the input x from the input layer to the hidden layer undergoes a linear transformation and then undergoes a non-linear transformation.

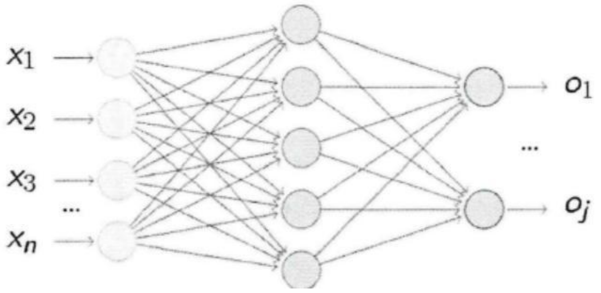


Figure 1 Schematic diagram of a fully connected network

$$h_1 = W_1 x + b_1; z = act\left(h_1\right); o = W_2 z + b_2 \tag{1}$$

Among them, $x \in R^n$ is input and o is output. W_1 , W_2 , b_1 and b_2 are the parameter to be learned. W_1 and b_1 are the parameter of the hidden layer and W_2 and b_2 are the parameter of the output layer. act is a non-linear activation function, such as tanh, sigmod and ReLU, etc.

Due to the introduction of the nonlinear activation function, the fully connected layer has the ability to fit the nonlinear function, and thus has a larger model capacity. A neural network with a wider stack (larger hidden layer dimensions) and deeper (more hidden layers) can fit more complex nonlinear functions.

2.3.2 Convolutional Neural Networks

Convolutional neural networks (convolutional neural network, CNN) are widely used in image processing related tasks (such as image classification, target detection, object recognition, etc.), has also been applied to natural language speech processing and speech processing tasks. The fully connected network requires corresponding parameters for each dimension of data. For image tasks, using a fully connected network will cause a lot of parameters and a huge model, which is not conducive to training and deployment use. The convolutional neural network uses a smaller tensor as a parameter (called the convolution kernel) in the input. The input height and width dimensions are sliding processing, and the input at different positions shares this parameter. This method is used to save province model parameters. Convolutional neural networks include convolution operations, nonlinear transformation and pooling operations. Processing the image information is an example to illustrate the calculation process of the convolution operation. For the input picture $I \in R^{H \times W \times C}$, where I is the picture, H is the height of the film, W is the width of the picture, C is the feature number of the picture, and its three primary colors (R, G, B) are generally used. The color value of as its characteristic, that is, C = 3, the whole picture is a three-dimensional tensor. Parameters of convolution operation, that is, the convolution kernel is $g \in R^{k \times k \times C \times C_{out}}$, where k is the size of the convolution kernel and C_{out} is the number of output features, also known as number is a four-dimensional tensor. Then, the convolution operation is

$$(I * g)(i, j) = \sum_{m=-\frac{k}{2}}^{\frac{k}{2}} \sum_{m=-\frac{k}{2}}^{\frac{k}{2}} I(i+m, j+n)\, g\left(m + \left(\tfrac{k}{2}\right), n + \left(\tfrac{k}{2}\right)\right) \tag{2}$$

The size of the convolution kernel and the number of output features need to be designed by the network designer, and there is also a step size (stride), void rate (dilation), filling method (padding) and other parameters can be designed/selected. The convolution kernel size is the size of the area that can be sensed by the convolution operation. When $k = H$, the convolution kernel sees the entire picture. It degenerates into a fully connected network. The step size indicates that the convolution kernel is slipping. The step length of each sliding in the dynamic calculation process. Filling means adding specific elements around the output image to control the size of the output.

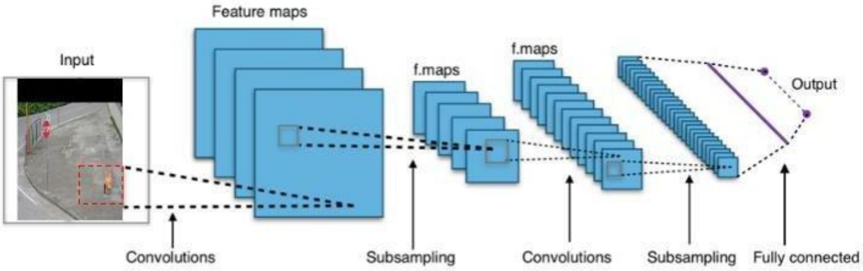


Figure 2 Architecture of CNNs

A topological structure of BP training, and good performance has been obtained in many experiments. A small part of the image called the local receptive area in CNNs is used as the bottom input of the hierarchical structure. Information is transmitted through different network levels, so the salient features of observation data that are invariant to translation, zoom, and rotation can be obtained at each layer.

Deep learning has been widely used into large amount of pattern classification tasks. Although this field is in the early stages of development, its development will undoubtedly give an boost to machine learning and AI domain. At the same time, there are still some specific tasks that are not suitable for processing, such as language recognition. The features extracted by generative pre-training can only describe potential voice changes, and will not contain enough distinguishing information between different languages; iris recognition, etc. The problem of pattern classification where the sample contains only a single sample is also a task that cannot be completed well. Deep learning still has a lot of work to be studied. In terms of models, whether there are other more effective and theoretically based deep model learning algorithms and exploring new feature extraction models are worthy of in-depth study. In addition, effective parallel training algorithms are also a direction worth studying.

2.3.3 Recurrent Neural Network

We treat RNN as a type of recursive neural network which takes sequential data as input, recursively in the direction of sequence evolution, and all nodes are connected in a chain, which aims to identify sequential features and use previous patterns to predict the next possible situation , The structure is shown in Figure 3. In order to solve the problems of RNN, LSTM is proposed, as shown in Figure 4. A special storage unit is designed so that it can remember the input historical information for a longer period of time. LSTM is composed of 3 gates, and the input gate You can control whether new inputs are allowed, and the forget gate controls which unimportant information is ignored, and finally the information is output through the output gate. The network can learn the long-term dependence of the input data well and remember the historical data information for a longer period of time. LSTM The forward propagation algorithm of LSTM is similar to RNN. It takes a time series of length T as input data. Each time the time step advances, the output result is updated. The backward propagation algorithm of LSTM is also similar to that of RNN. Beginning at the end, the gradient of each parameter is gradually calculated in the reverse loop, and finally the network parameters are updated with the gradient of each time step. Both RNN and LSTM can process time series data to learn time dependence, and have been widely used in the field of time-space sequence prediction research.

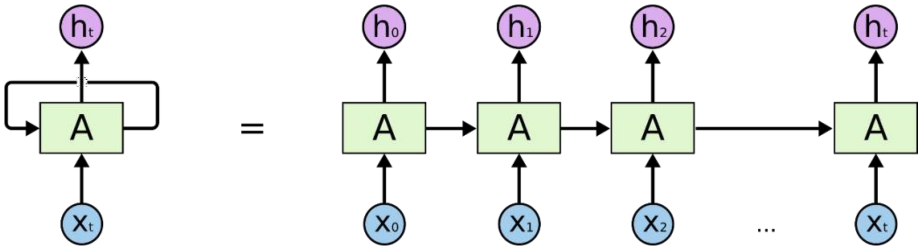
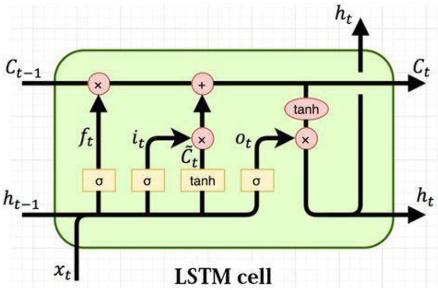


Figure 3 Structure form of standard RNN



$$i_t = \sigma\left(x_t U^i + h_{t-1} W^i\right)$$
$$f_t = \sigma\left(x_t U^f + h_{t-1} W^f\right)$$
$$o_t = \sigma\left(x_t U^o + h_{t-1} W^o\right)$$
$$\tilde{C}_t = \tanh\left(x_t U^g + h_{t-1} W^g\right)$$
$$C_t = \sigma\left(f_t * C_{t-1} + i_t * \tilde{C}_t\right)$$
$$h_t = \tanh(C_t) * o_t$$

Figure 4 Structure of LSTM

3. Methodology

Of course, stock market prediction technology has great economic value for stock market investors and investment institutions, helping investors and investment institutions to make profits and avoid investment risks. But the value of stock market forecasting technology is far more than that. From a social perspective, stock market forecasting technology can prevent systemic risks in the financial market, help rationally allocate social funds, and contribute to the harmonious and stable development of the economy. Stock data has its own characteristics, and the existing forecasting technology methods are not fully used, so its research brings new challenges to the technology. In particular, the multi-scale and multi-source heterogeneous prediction technology can not only be used for stock market prediction, but also has broad application prospects in many fields such as personal health state prediction, energy demand prediction and website traffic prediction. This research not only has important socio-economic value, but also has important academic research value.

The multi-scale property of stock market data refers to the existence of data at different time intervals, and the data at different scales will reflect the stock movement state of different time periods. Large-scale stock market data can reflect the long-term movement state of the stock market, and small-scale stock market data can reflect the short-term movement state of the stock market. Data of different scales have associated information and their own unique information. In order to describe the current market state more accurately, it is necessary to comprehensively consider stock market data of multiple scales. However, most of the existing researches only focus on the single-scale data of the stock market. This can lead to less-than-expected forecast performance due to an inaccurate description of the state of the stock market. How to effectively use multi-scale data is the key to accurately describe and predict the market state.

3.1 Data source

This article selects S&P500 index through yahoo finance, and the transaction data for each trading day from September 26th 2001 to September 24th 2021. The data includes 5000 observations. The selected data are divided into two parts. First part occupied 70% of the selected data to train the model , and the remaining observations are considered to test and validation.

3.2 Methodology

The volatility of stock prices is controlled by the trend of the stock, but is also sensitive to many other factors. Due to the relative stability and predictability of the intrinsic value of stocks, the factors that have impacts on the stock market price mainly include the following aspects: 1. Macro factors; 2. Industrial and regional factors; 3. Company factors; 4. market factors. This article predicts the closing index of the S&P500 rather than specific company stock price forecasts, so aside from the more microscopic industry and company factors, it mainly focuses on the influence of macroeconomic factors and market factors. Macroeconomic factors refer to the impact of macroeconomic environment and its changes on stock prices, including regular factors such as cyclical fluctuations in macroeconomic operations and policy factors such as monetary policy implemented by the government. This article predicts the daily data of the S&P500 closing index, mainly focusing on the impact of monetary policy and other policy factors on stock prices.

There are two types of stock price forecasting methods: qualitative analysis and quantitative analysis. The qualitative analysis method is the fundamental analysis method, which is a subjective analysis method relying on the experience of financial practitioners. This thesis is a numerical prediction of the daily closing index of the

S&P500 rather than a trend judgment of price fluctuations, so this thesis mainly focuses on the literature review of quantitative analysis methods.

Numerical data-based stock market forecasting research uses numerical data on a certain time scale in the stock market, such as sky-level index prices and stock price volume data, to predict specific stocks or other investments in the stock market on the same scale. Predict the future price of the underlying. According to the focus of the research, these studies can be divided into research on the characteristics of numerical data stock market forecasting and research on the numerical data stock market forecasting model.

In order to build our model, in addition to the traditional ARIMA model, this article will also use the LSTM model. The model in this article uses 70% of the data for training, and the remaining 30% of the data is used for testing. For training, we use Root Mean Square Error and Adam algorithm to optimize the model. This Article will use Stata12 to calculate the ARIMA and GARCH model and use Matlab for the training.

(1). ARIMA model

As the stock data is noisy, we must first perform stationarity test on the stock sequence. The test method is to observe the sequence diagram, autocorrelation diagram, and partial autocorrelation diagram of the sequence first, and then do a unit root (ADF) test to test its P If the sequence is non-stationary, then we choose the difference for smoothing. After determining the order of the difference, confirm that it is a stationary sequence, which can be used to determine the order of the model, that is, p, q. This article chooses to use the BIC value to determine Order. After the determination is completed, the model is tested, mainly the LB test, to confirm whether the residual is white noise. If it is, then the model passes the test and we can make predictions.

(2). Single Feature LSTM neural network

This model chooses the s&p500 return as the only input feature. First, it is necessary to test the stationarity of the closing price series: generally choose to draw a time series diagram first, and check whether the image has an obvious trend; then, we also need to draw a correlation diagram, and through observing whether the acf image is quickly reduced to 0 to judge the stationarity; then perform the ADF test; finally, if the data does not have stationarity, then the difference method is needed to smooth the data.

After smoothing the data, you can construct a single-feature LSTM neural network. This article chooses a three-layer LSTM network, that is, there is only one hidden layer, and the input layer has 20 neurons, so that it can process 20 days of stock prices , Because we are calculating the closing of the next day, so the output layer has only 1 neuron, which is used to output the stock price of the twenty-first day. 20 is chosen because after n-fold cross-checking, 20 is found to be the optimal parameter.

(3) GARCH model

In the 1980s, Engel proposed ARCH (auto regressive conditional heteroskedastic process) model, which is an autoregressive conditional heteroskedastic process model, can be used to make such predictions. The ARCH model defined by Engel.

The GARCH model hold an idea that the variance for the change of return can be predictable, not only the latest information, but also the previous conditional variance will have an affection on the conditional variance. In order to simplify the calculation, the risk metrics proposed by the JP Morgan Group's risk management company uses a simple and practical GARCH(1,1).

(4) Mixed model construction

The mixed model is constructed by ensembling three models including ARIMA model, Garch Model and LSTM model. The innovation of the article emphasizes the long-term dependence of LSTM on performance to improve accuracy, and ensemble can improve the robustness of the model.

(5) Estimator parameters

Since the ultimate goal of stock market forecasting is profit, how to correctly evaluate the model and select the model with the best profitability is very important in stock market forecasting. The current stock market forecast research generally adopts a two-stage model evaluation method: first, the performance of the model is evaluated, and then the model with the best performance is selected to evaluate the profitability of the model. The performance evaluation of the stock market forecasting model usually adopts the classification evaluation indicators, such as the accuracy rate and F1 value, and the profitability of the stock market forecasting model is estimated by various simulated trading algorithms. There may be a lack of consistency between the above two evaluation methods, that is, the profitability of the model with the best classification evaluation performance is not necessarily the best. This inconsistency can lead to the improvement of stock market forecasting models without valuable guidance. How to reduce this inconsistency and improve the validity of model evaluation is a difficult point in stock market research.

Mean Absolute Error (MAE)

Mean Absolute Error (MAE) is the most basic evaluation method, and its expression is as follows:

$$MAE = \frac{1}{N}\sum_{i=1}^{N} \left| observed_i - predicted_i \right| \tag{3}$$

Mean Square Error (MSE)

The mean squared error (Mean Squared Error) expression is as follows:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}\left(observed_i - predicted_i\right)^2 \tag{4}$$

The value of MSE is inversely proportional to the accuracy of the model. The larger the MSE, the worse the prediction effect of the model.

Root Mean Square Error (RMSE)

Root Mean Square Error (Root Mean Square Error) can be used to calculate the deviation between the observed value and the true value. Because the average index is non-robust, this makes the average error very sensitive to outliers. The expression is as follows:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(observed_i - predicted_i\right)^2} \tag{5}$$

4. Results and discussion

4.1 Arima process

The ARIMA time series model is a differential processing of the autoregressive moving average model. Its main methods are modeling, evaluation, verification and control, which are expressed as ARIMA(p, d, q). The main idea of the model is to regard the known data as a random sequence when it is formed in the order of time development, and then describe the random sequence by mathematical modeling. Time series values predict future values.

This project chooses the closing price sequence as the time sequence, and the sequence diagram of the closing price sequence is shown in Figure 5:

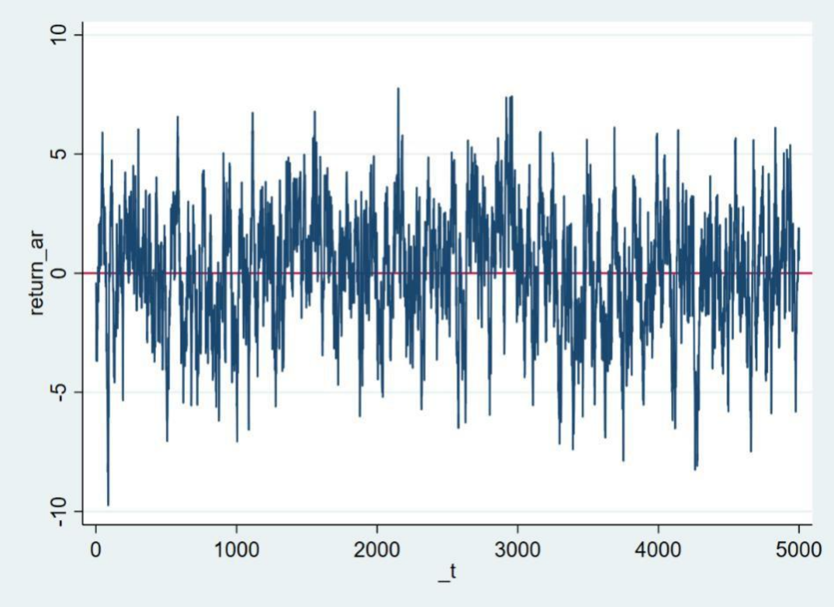
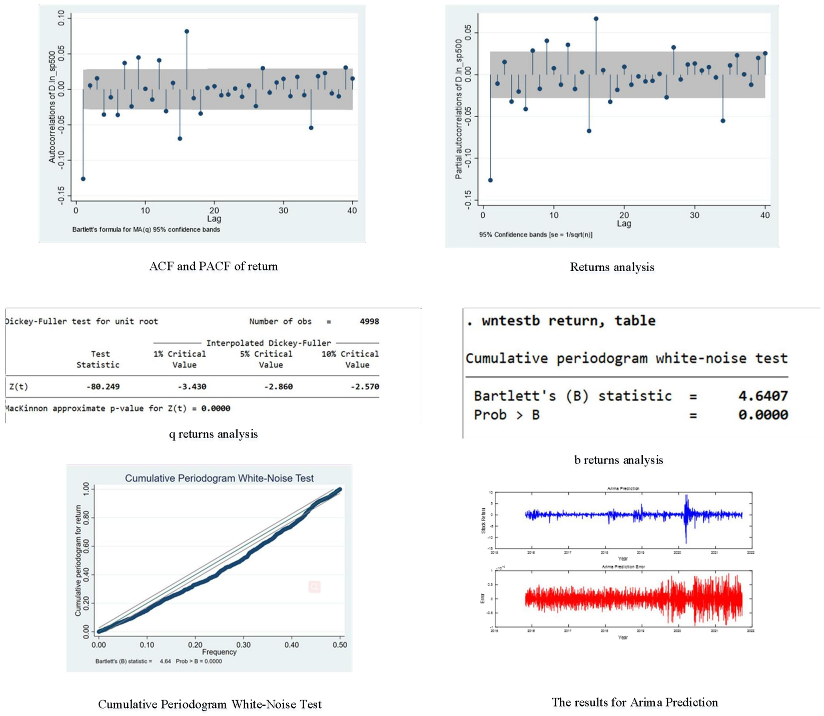


Figure 5 Timing chart of return series

As observed in the above figure, it can be seen that the stock price fluctuated to a certain extent during the period, and it was not stable.

Observing the sequence diagram of the sequence, we can consider the data to be non-stationary. In order to confirm our conjecture, we then draw the autocorrelation graph and the partial autocorrelation graph, as shown in Figure 6:



ACF and PACF of return

Returns analysis

```
Dickey-Fuller test for unit root                   Number of obs   =      4998

                               ---------- Interpolated Dickey-Fuller ---------
                  Test         1% Critical       5% Critical      10% Critical
               Statistic           Value             Value             Value
------------------------------------------------------------------------------
 Z(t)            -80.249            -3.430            -2.860            -2.570
------------------------------------------------------------------------------
MacKinnon approximate p-value for Z(t) = 0.0000
```

q returns analysis

```
. wntestb return, table

Cumulative periodogram white-noise test

 Bartlett's (B) statistic  =     4.6407
 Prob > B                  =     0.0000
```

b returns analysis

Cumulative Periodogram White-Noise Test

The results for Arima Prediction

Figure 6 The results for Arima Prediction

As can be seen from figure above, the ARIMA model is not very effective in predicting stock data, the error fluctuates between +7.2 and -2.6, and the upward or downward trend is basically the same as the change trend of the original data.

4.2 GARCH process

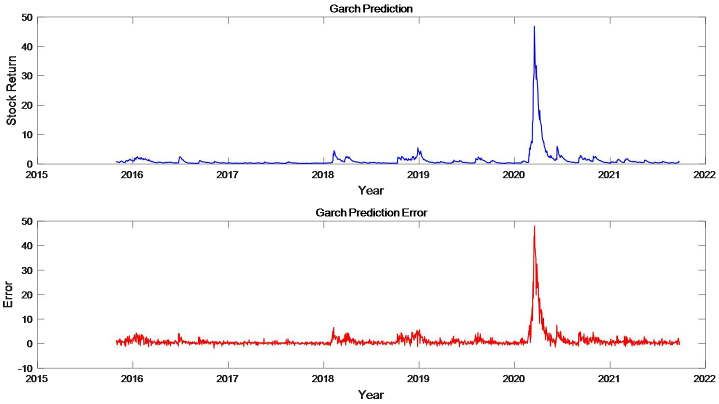


Figure 7 The results for Garch Prediction

As can be seen from Figure 7, the Garch model is better than the single ARIMA model in predicting stock data. For the Garch model, the error fluctuates between +2.1~-0.6, but the error is basically concentrated between +1~-1, the rising or falling trend is almost the same as the change trend of the original data , and even have overlapping intervals. It can be seen that the Garch model is stronger than ARIMA model's prediction of stocks in both the accuracy of stock prediction and the trend of stock changes.

4.3 LSTM process

In traditional neural networks, neurons in the same hidden layer are not connected to each other, and this structural defect directly leads to their poor performance in dealing with certain problems. This shortcoming becomes especially acute when dealing with time series and speech recognition problems where information is contextualized. The emergence of the recurrent neural network solves this problem very well. The neurons in the same hidden layer are connected to each other, which can effectively obtain the contextual information of the data. The output of the recurrent neural network is determined according to the input and the previous related information, so it can play its short-term memory when dealing with time series problems.

Although the effect of recurrent neural network in dealing with time series problems is very good, there are still some problems. The more serious one is that gradient disappears or explodes easily in the processing of long-term span problems. Causes the phenomenon of small memory value. After the cyclic neural network is expanded, it can be regarded as a multi-layer feedforward neural network with each layer sharing the same weight parameters. Although it keeps trying to learn the long-term dependencies of sequences, actual research finds this to be a difficult task indeed. Long-term reliance on signals tends to become very weak and highly susceptible to short-term signal fluctuations. There is a multiplier of the derivative of the activation function in time-based backpropagation, and the continuous accumulation will cause uncontrollable problems. Although it can be solved theoretically by adjusting the parameters, it is found that this problem is difficult to solve in practice, so it still needs to be optimized from the structure. This leads to its improved structure - the LSTM neural network.

Next, we can start the construction of the LSTM neural network. The first is the determination of several parameters. After n-fold cross-validation, we choose the hidden layer to have 10 neurons. The number of iterations is selected 50 times, and each 72 sample data is formed into a batch for training, that is, batchsize = 72, Adam algorithm is used as the optimizer of the model, the learning rate is 0.001, and the training set data is randomly scrambled. Use the MSE indicator as the loss function of the model for training. Figure 8 is the training diagram of the neural network. It can be seen from this diagram that after iteration, the loss function of the model decreases quickly and tends to converge. It can be seen that the prediction model is more reasonable.

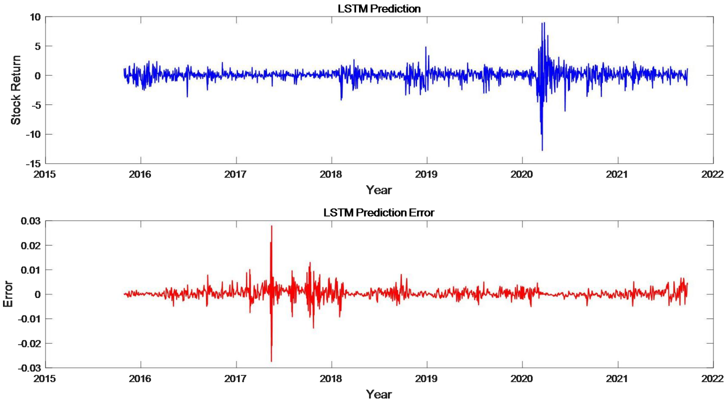


Figure 8 The results for LSTM Prediction

The code of LSTM can be seen in Apppendix.

4.4 Mixed model process

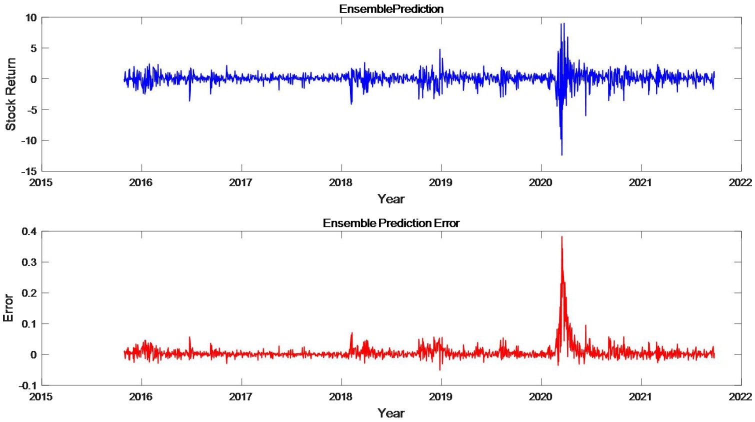


Figure 9 The results for Ensemble Prediction

This chapter chooses the traditional time series model and the LSTM neural network model to construct the stock price model and make predictions. First, the ARIMA model is used, and only the closing price sequence is stabilized, the model is determined, and the model is checked. Finally, the stock price forecast was made; then a new forecast of the stock price was made using the GARCH model; at this point, the application of the traditional time series model ended. Next, using the same data, the LSTM neural network with single-feature input and multi-feature input was constructed, the number of layers was selected and the neurons were determined, and when the model parameters were trained to meet the standards, predictions of closing prices were given. Finally, the mean square error (MSE) of each model was calculated separately to compare the models.

We are going to leverage the model of LSTM into traditional financial time series forecasting, and a stock forecasting model based on long short-term memory neural network (LSTM) is established. The absolute error and the coefficient of determination were evaluated, and a better prediction effect was obtained. It proves the feasibility of deep learning in financial time series forecasting, which can guide the investment behavior of institutions and individuals to a certain extent, and provides new ideas for stock forecasting research.

Firstly, an ARIMA model was established based on the closing index sequence. The input feature was the closing index of the previous day. The MSE of the initial test was 2.185, and the average error rate was 4.48%. After hyperparameter tuning, the prediction effect was significantly improved, and the MSE dropped to 1.213, and the average error rate is reduced to 3.19%. During the test period, the average yield fluctuation of the closing index was 0.62%, which was far lower than the optimal forecasting effect of the model. It can be considered that the forecasting effect of the ARIMA model on the S&P500 is extremely poor and has no practical significance. At the same time, to a certain extent, it shows that there are many factors affecting the fluctuation of stock prices, and the historical price of the S&P500 closing index cannot fully reflect the relevant information of the stock market.

Secondly, the optimal prediction effect of the GARCH model after hyperparameter tuning is MSE of 1.923 and an average error rate of about 1.65% in the test period, which are both better than the prediction evaluation indicators of the ARIMA model, indicating that the comprehensive index of historical transaction information (SH index)), Investor Sentiment Composite Index (IS Index) and Monetary Policy Composite Index (MP Index), compared with a single sequence of closing indices, they express more information on the S&P500, and the three-factor forecast model has good practical significance. However, the average error rate at this time is still higher than the average yield volatility of the closing index of 0.62%, and the applicability of the GARCH method to the forecasting of the Shanghai Composite Index still needs further investigation.

Finally, examining the predictive ability of the LSTM model established based on 13 original indicators after tuning the hyperparameters, it can be found that the MSE between the predicted value of the model and the actual value during the test period was 0.876, the average error rate was further reduced to 0.40%, and the predictive ability was significantly better. The MSE of the Mixed Model is 0.412 and the average error rate is 0.27%.

Table1 Accuracy Comparison For regression models

Model	MSE	Average Error Rate
ARIMA	1.213	3.19%
GARCH	1.923	1.65%
LSTM	0.876	0.40%
Mixed Model	0.412	0.27%

Base on Table1, the MSE and Average Error Rate of each model was calculated separately for comparison between the models. It can be seen that the MSE value and Average Error Rate of the Mixed Model is the lowest, so it can be clearly seen that the Mixed Model is better than the traditional time series model.

5. Conclusion and Implication

5.1 Conclusion

The importance of the stock market to a country's economy will make the types of stock price forecasting methods continue to develop and grow, and will continue to be derived from the development of other disciplines. In the development process of the follow-up forecasting method, it is necessary to continuously explore and deeply study the characteristics of the stock market, so as to make the model closer to reality, expand the applicability of the method, and obtain better forecasting accuracy.

Because stock data is affected by economic factors, political factors or environmental factors, the law of its change is elusive, and the cycle of the law of change is difficult to determine. Therefore, the model still needs a lot of historical data and selection of appropriate variables for analysis to obtain the desired results. In the traditional ARIMA model, when analyzing complex stock markets, its prediction results are not particularly ideal, and there are still certain errors in price prediction. As a technology in the field of deep learning, neural network can solve non-linear problems well. LSTM neural network is optimized on traditional neural network and introduces the concept of "gate", which enhances the long-term memory ability of the model , Which enhances its generalization ability. Therefore, the application of LSTM neural network in analyzing financial-related time series data is promising.

Based on the understanding of traditional time series analysis and RNN and LSTM neural network, this paper constructs a stock price prediction model based on LSTM neural network. For better comparison, we also established a traditional ARIMA model for comparison. As the neural network has a good predictive effect on nonlinear problems, this article chooses the optimized neural network-LSTM model, and also chooses the use of single-feature and multi-feature input models to seek better prediction results. The traditional time series model focuses on the role of time in stock forecasting. However, certain errors will occur when the model deals with complex nonlinear stock data, and the model does not consider other factors, such as economics and politics, so the prediction error of the ARIMA model will be large. Next, this thesis considers the ARIMA model, the GARCH model, and the single-feature input LSTM model that can handle nonlinear data, but they all have unconsidered problems, and their prediction results will also appear to be certain. The error. The multi-feature input LSTM model not only takes into account the influence of external factors, but also can process non-linear data, and its prediction performance is better. Through the result of the prediction, we can see that the prediction result of the mixed model is the best.

For the work of this article, the following points can be summarized:

(1) Carry out the steps of smoothing, model ordering, and model checking on our stock data, and finally establish the ARIMA model and predict the stock price; (2) Carry out the steps of smoothing, model ordering, and model checking on our stock data, and finally establish the GARCH model and predict the stock price; (3) Construct an LSTM neural network, determine the number of neural network layers and neurons, and train the parameters; (4) Construct a mixed model to predict stock prices. (5) Compare the prediction results of the above models.

5.2 Implication

The prediction model studied in this article is based on the LSTM neural network, and preliminary results have been obtained. However, due to objective factors such as research time and data sources, there is still a lot of room for research in this article. There is still much to be done for the model constructed in this article. Update and improvement, the follow-up work mainly includes the following aspects:

(1) Handling of abnormal values

Because stock market has a certain degree of speculation and is also susceptible to policy influences, there are often skyrocketing and plummeting situations. This leads to outliers in the stock data we obtain. There are many reasons for the occurrence of outlier points in stock data, which cannot be obtained by quantitative analysis. This makes the problem unable to simply use the LSTM neural network constructed in this article. Therefore, some methods to deal with outliers can be used to perform data processing. Noise reduction, such as wavelet transform, Fourier transform, etc.

(2) About feature selection

The number of features of the data set obtained in this article is not very large, and for real stock data, in addition to the features in the stock market, can you use other features, such as corporate financial reports; also, because the stock market is affected by policy Larger, using the application of LSTM neural network in text learning and text sentiment analysis, can we obtain some features from news and financial reports, so as to enable the model to make corresponding judgments on stocks in an economic sense, thereby improving the accuracy of our predictions rate.

(3)About model optimization

When constructing the neural network model, whether the number of hidden layers is small, and whether more hidden layers will have better prediction results, this is also the lack of research in this article.

Reference

Addison, P. S. (2002). The illustrated wavelet transform handbook. Napier University.

Avramov, D. (2002). Stock returns predictability and model uncertainty. Journal of Financial Economics, 64, 423–458.

Brock, W., Lakonishok, J., & LeBaron, B. (1992). Simple technical trading rules and the stochastic properties of stock returns. The Journal of Finance, 47, 1731–1764.

Campbell, J. Y., & Thompson, S. B. (2008). Predicting excess stock returns out of sample: Can anything beat the historical average? Review of Financial Studies, 21,1509–1531.

Campbell, J. Y., & Vuolteenaho, T. (2004). Bad beta, good beta. The American Economic Review, 94, 1249–1275.

Chen, J., Jiang, F., & Tong, G. (2017). Economic policy uncertainty in China and stock market expected returns. Accounting and Finance, 57, 1265–1286.

Clark, T. E., & West, K. D. (2007). Approximately normal tests for equal predictive accuracy in nested models. Journal of Econometrics, 138, 291–311.

Cochrane, J. H. (2007). The dog that did not bark: A defense of returns predictability. Review of Financial Studies, 21, 1533–1575.

Conrad, J., & Kaul, G. (1998). An anatomy of trading strategies. Review of Financial Studies, 11, 489–515.

Cowles, A., 3rd (1933). Can stock market forecasters forecast? Econometrica. Journal of the Econometric Society, 309–324.

Dai, Z., Zhou, H., Wen, F., & He, S. (2020a). Efficient predictability of stock return volatility: The role of stock market implied volatility. The North American Journal of Economics and Finance, 52, 101174.

Dai, Z., & Zhu, H. (2020). Stock returns predictability from a mixed model perspective. Pacific-Basin Finance Journal, 60, 101267.

Dai, Z. F., Dong, X. D., Kang, J., & Hong, L. (2020b). Forecasting stock market returns: New Technical indicators and two-step economic constraint method. The North American Journal of Economics and Finance, 53, 101216.

Dangl, T., & Halling, M. (2012). Predictive regressions with time-varying coefficients. Journal of Financial Economics, 106, 157–181.

Daubechies, I. (1992). Ten lectures on wavelets. Philadelphia, PA: SIAM (Society for Industrial and Applied Mathematics).

DeMiguel, V., Garlappi, L., Nogales, F. J., & Uppal, R. (2009). A generalized approach to portfolio optimization: Improving performance by constraining portfolio norms. Management Science, 55, 798–812.

Fama, E. F., & Blume, M. F. (1966). Filter rules and stock market trading. Journal of Business, 39, 226–241.

Fama, E. F., & French, K. R. (1988). Dividend yields and expected stock returns. Journal of Financial Economics, 22, 3–25.

Faria, G., & Verona, F. (2018). Forecasting stock market returns by summing the frequency-decomposed parts. Journal of Empirical Finance, 45, 228–242.

Ferreira, M. I., & Santa-Clara, P. (2011). Forecasting stock market returns: The sum of the parts is more than the whole. Journal of Financial Economics, 100, 514–537.

Gençay, R., Selçuk, F., & Whitcher, B. (2002). An introduction to wavelets and other filtering methods in finance and economics. Academic Press.

Goyal, A., & Welch, I. (2003). Predicting the equity premium with dividend ratios. Management Science, 49, 639–654.

Goyal, A., & Welch, I. (2008). A comprehensive look at the empirical performance of equity premium prediction. Review of Financial Studies, 21, 1455–1508.

Guo, H. (2006). Time-varying risk premia and the cross section of stock returns. Journal of Banking & Finance, 30, 2087–2107.

Haven, E., Liu, X., & Shen, L. (2012). De-noising option prices with the wavelet method. European Journal of Operational Research, 222(1), 104–112.

Inoue, A., & Kilian, L. (2004). In-sample or out-of-sample tests of predictability: Which one should we use? Economic Review, 23, 371–402.

Jaffard, S., Meyer, Y., & Ryan, R. D. (2001). Wavelets from a historical perspective. Philadelphia, PA: SIAM (Society for Industrial and Applied Mathematics).

Jiang, F., Lee, J. A., Martin, X., & Zhou, G. (2019). Manager sentiment and stock returns. Journal of Financial Economics, 132, 126–149.

Appendix

```
clear all; close all;clc
rng(20211004)
mode = 1;    %% mode=2 LSTM training,     mode = 1 Loading

if mode == 2
%% Data Process
filename = "sp500.xls";
sheet = "Sheet1";
[num,txt,raw] = xlsread(filename,sheet);

data = num;
date = datenum(raw(3:end,1));


figure
plot(date,data,'linewidth',1,'color',[0 0 1]);
xticks([datenum('2001/1/1'),datenum('2002/1/1'),datenum('2003/1/1'),datenum('2004/1/1'),datenu
m('2005/1/1'),                                                       datenum('2006/1/1'),
datenum('2007/1/1') ,datenum('2008/1/1') ,datenum('2009/1/1') ,datenum('2010/1/1')...
        ,datenum('2011/1/1'),datenum('2012/1/1'),datenum('2013/1/1'),datenum('2014/1/1'),datenu
m('2015/1/1'),datenum('2016/1/1')...
        ,datenum('2017/1/1'),datenum('2018/1/1'),datenum('2019/1/1'),datenum('2020/1/1'),datenu
m('2021/1/1'),datenum('2022/1/1')])
```

```
xticklabels(['2001','2002','2003','2004','2005','2006','2007','2008','2009','2010'...
                    ,'2011','2012','2013','2014','2015','2016'...
                    ,'2017','2018','2019','2020','2021','2022'])
datetick('x','yyyy','keepticks');
xlabel('Year','FontSize',12);
ylabel('Stock Return','FontSize',12)
set(gcf,'unit','centimeters','position',[10 5 30 15]);
axis normal;

numTimeStepsTrain = floor(0.7*numel(data));
dataTrain = data(1:numTimeStepsTrain+1);
dataTest = data(numTimeStepsTrain+1:end);

mu = mean(data);
sig = std(data);
dataTrainStandardized = (dataTrain - mu) / sig;
dataTestStandardized = (dataTest - mu) / sig;

stay_to_predict = 10;

Train_shift = [];
for i =    1 : 1 : stay_to_predict + 1
       data_temp = dataTrainStandardized;
       data_temp = circshift(data_temp,-(i-1));
       Train_shift = [Train_shift data_temp];
end
Train_shift(end - stay_to_predict:end,:) = [];
XTrain = Train_shift(:,1:stay_to_predict);
YTrain = Train_shift(:,stay_to_predict+1);

Test_shift = [];
for i =    1 : 1 : stay_to_predict + 1
       data_temp = dataTestStandardized;
       data_temp = circshift(data_temp,-(i-1));
       Test_shift = [Test_shift data_temp];
end
Test_shift(end - stay_to_predict:end,:) = [];
XTest = Test_shift(:,1:stay_to_predict);
YTest = Test_shift(:,stay_to_predict+1);

%% LSTM

numFeatures = stay_to_predict;
numResponses = 1;
```

```
numHiddenUnits = 300;

layers = [ ...
        sequenceInputLayer(numFeatures)
        lstmLayer(numHiddenUnits)
        fullyConnectedLayer(numResponses)
        regressionLayer];
options = trainingOptions('adam', ...
        'MaxEpochs',500, ...
        'GradientThreshold',1, ...
        'InitialLearnRate',0.01, ...
        'LearnRateSchedule','piecewise', ...
        'LearnRateDropPeriod',125, ...
        'LearnRateDropFactor',0.2, ...
        'Verbose',0, ...
        'Plots','training-progress');
net = trainNetwork([XTrain;XTest]',[YTrain;YTest]',layers,options);
[net,YPred] = predictAndUpdateState(net,[XTrain;XTest]');
YPred = sig*YPred(end - 1488:end) + mu;
YTest = sig*YTest + mu;
rmse = sqrt(mean((YPred-YTest').^2))
figure
plot(YTest','r')
hold on
plot(YPred,'.-b')
hold off
legend(["Observed" "Predicted"])

save('YPred.mat','YPred')
save('YTest.mat','YTest')
end
%% Results and Plots
if mode == 1
filename = "sp500.xls";
sheet = "Sheet1";
[num,txt,raw] = xlsread(filename,sheet);

data = num;
date = datenum(raw(3:end,1));
load('YPred.mat')
load('YTest.mat')
rmse = sqrt(mean((YPred-YTest').^2))
%%
figure
```

```
subplot(2,1,1)
plot(date(end - 1488:end),YPred,'linewidth',1,'color',[0 0 1]);
xticks([datenum('2015/1/1'),datenum('2016/1/1')...
        ,datenum('2017/1/1'),datenum('2018/1/1'),datenum('2019/1/1'),datenum('2020/1/1'),datenu
m('2021/1/1'),datenum('2022/1/1')])
xticklabels(['2015','2016'...
            ,'2017','2018','2019','2020','2021','2022'])
datetick('x','yyyy','keepticks');
xlabel('Year','FontSize',12);
ylabel('Stock Return','FontSize',12)
title('LSTM Prediction')
set(gcf,'unit','centimeters','position',[10 5 30 15]);
axis normal;

subplot(2,1,2)
plot(date(end - 1488:end),YPred - YTest','linewidth',1,'color',[1 0 0]);
xticks([datenum('2015/1/1'),datenum('2016/1/1')...
        ,datenum('2017/1/1'),datenum('2018/1/1'),datenum('2019/1/1'),datenum('2020/1/1'),datenu
m('2021/1/1'),datenum('2022/1/1')])
xticklabels(['2015','2016'...
            ,'2017','2018','2019','2020','2021','2022'])
datetick('x','yyyy','keepticks');
xlabel('Year','FontSize',12);
ylabel('Error','FontSize',12)
title('LSTM Prediction Error')
set(gcf,'unit','centimeters','position',[10 5 30 15]);
axis normal;
end
```