Stock Market Forecasting and Analysis Using Artificial Neural Network Algorithms

by

LI Cheng

Submitted in partial fulfillment of

the requirement for the degree of

DOCTOR OF ENGINEERING

Intelligent Information System Engineering
Graduate School of Engineering
Fukuoka Institute of Technology
Fukuoka, Japan

March 2023

Acknowledgements

My completion of this thesis is under the guidance of Professor Song Yu, who has conducted many times of revision and put forward valuable guidance for me in the process of writing. Under his constant and patient help, I have finally completed this thesis. Here, I would like to extend my sincere appreciation to my supervisor, Professor Song.

As a person easily falling into indecision and regret after some matters, I always hope to face the coming things after making enough preparations. When just entering the doctoral stage, I often asked myself whether I was excellent enough to study for a doctor? My answer was obviously no. For a long time after that, I had fallen into the status of self-denial. Fortunately, I met a rigorous supervisor with a gentle personality, whose patience and inclusiveness helped me overcome the difficulties I met at that time. It was just his meticulous academic attitude and persistent teaching that made me able to achieve the improvement based on my own cognition. I have been aware that the study for a doctoral degree is not because I am excellent enough, it is because that I want to become a more outstanding person. I am deeply indebted to Professor Song for this chance he gives me to become his student and study in Fukuoka Institute of Technology. I would like extend the gratitude to him again.

Thank Professor Zhu Shijie, Professor Fujioka Hiroyuki and Professor Tajima Takuya for their advice and kindness. I am grateful to them for sharing their experience, expertise and wisdom in thesis revision and oral defense. It is my honor to have these three professors as my thesis review committee members. I would like to thank for all the professors on the eighth floor of Building B, who have patiently taught me in both Japanese and English to help me quickly adapt to my learning life when I was poor in my Japanese.

Besides, I want to express my appreciation to other teachers and school staff for their continued help and support to me over the years, as well as the seniors and juniors in the research room, especially my senior sister Qiu Mingyue, for her help and advice in previous research and career direction. I would also like to thank my family and friends. When I was in a foreign country alone, their care had given me much strength and warmth. Thank them very much for their concern. I miss all of them too.

During the study at my doctorate stage, what I harvested was the thinking about the academic philosophy and life attitude in addition to acquiring a doctoral degree. Although the way stretches endless ahead, I would like to still make constant exploration. I am convinced that the unremitting efforts and serious attitude will be the

most correct choice in my life at any time.

Last but not least, I would like to extend my gratitude to my mother for her meticulous care and nurturing on me. It's impossible for me to express my gratitude to her in words. She has given me the most sacred and selfless love in the world. This honor also belongs to you. I will love you forever, my dear mother.

Contents

List of Figures

List of Tables

Abstract

With the rapid advancement of machine learning, collecting data and using machine learning techniques to analyze and predict the data have become more and more important in industry development. It is widely used in research fields such as agriculture, medicine, military and finance. In particular, stock market index prediction is a hot field that attracts many researchers and investors due to its potential profitability.

In this thesis, we use machine learning technology such as neural network method and deep learning method to forecast financial time-series data. The datasets are mainly composed of Nikkei 225 and some individual stocks.

The original artificial neural network (ANN) model is improved by changing different combinations of activation functions. We applied the improved ANN model from stock market index to individual stocks and then proposed an investment strategy. However, the shallow neural network model is not very adaptable. The number of neurons expands with the increase of the dataset and the model becomes difficult to train. For this reason, we chose deep learning, which has a stronger learning ability and broader coverage, for stock market movement prediction.

When forecasting long time series data, the longer the series, the slower the convergence speed and the worse the results. Long short-term memory (LSTM) model was developed to overcome these difficulties. In forecasting Nikkei 225, we improved the structure of the LSTM model to realize the model from univariate forecasting to multivariate forecasting. At the same time, we computed some technical indicators as feature extraction and augmented the dataset. To compare the efficiency of different neural network layers on the improved LSTM model, convolutional neural network

(CNN) and encoder-decoder layer are combined with the model respectively. The results show that the improved LSTM model can process multivariate data efficiently after data augmentation. It is the best performer compared with CNN-LSTM model and encoder-decoder LSTM model.

The structure of this thesis is as follows. In Chapter 1, the background, motivation, significance, method and the structure of this thesis are introduced in detail. The research status is also presented. It includes traditional time series prediction methods, shallow neural network, some other machine learning methods and deep learning methods. In Chapter 2, a hybrid ANN model is applied to predict the Japanese stock market. We change the combination of activation functions to test the prediction performance of the ANN model. In Chapter 3, an LSTM model is improved to achieve the model from univariate prediction to multivariate prediction. Compared with the other variants, the improved LSTM model performs better. The conclusion is summarized in Chapter 4. The results are concluded and analyzed in general terms.

和文要旨

近年、IT の急速な発展に伴い、ビッグデータや機械学習理論もめざましい進歩を遂げている。データを大量に収集し、そしてそれを機械学習技術で分析することが徐々に産業の各分野に浸透し、農業、医療、軍事、金融などの研究分野で広く利用されるようになった。株式市場における株価予測もその潜在的な収益性から、多くの研究者や投資家を惹きつける注目の研究対象となっている。

本論文では、人工ニューラルネットワーク（ANN: Artificial Neural Network）、とりわけディープラーニングなどの機械学習技術を用い、株式市場で収集される時系列データの予測を試みる。

使用するデータセットは日経平均株価指数と一部の個別銘柄のデータで構成されている。まず、ANN モデルとメタヒューリスティクス手法を結合させるハイブリッド ANN モデルに対して、活性化関数の違いの影響を調査し、活性化関数を適切に選択することで、モデルの予測精度を効果的に向上させることができることを明らかにした。また提案したモデルに基づいて、株式の投資戦略を提案しシミュレーションを行った。

しかし、従来の ANN モデルは適応性が低く、データセットの増加に伴いニューロン数が増加し、モデルの学習が困難になったため、株価変動の予測にはより強い学習能力と広いカバー範囲を持つディープラーニングを取り入れた。

通常、長い時系列データを予測する場合、系列が長ければ長いほど、収束スピードが遅くなり結果が悪くなる。これを克服するために LSTM（Long Short－Term Memory）モデルが提案されているが、本研究では、日経平均株価指数を予測ために、モデルの構造を改良し、複数特性値による予測モデルを提案した。同時に、特徴を抽出したいくつかのテクニカル指標を取り入れ、データセットを拡張した。その結果、データ拡張後のデータセットを用いた場合、改良モデルで良い結果が得られた。また、CNN-LSTM モデルやエンコーダ・デコーダ LSTM モデルと比較し改良モデルが優位であることがわかった。

本論文の構成は以下の通りである。第 1 章では、本論文の背景、研究目的、方法、構成を説明し、関連する先行研究を紹介する。第 2 章では、ハイブリッド ANN モデルを日本の株式市場の予測に適用し、活性化関数を変化させ、ANN モデルの予測性能を向上させるための適切な活性化関数を選定した。また、予測結果に基づいて投資戦略を提案した。第 3 章では、LSTM モデルを改良し、単一特性値による予測モデルから多特性値による予測へのモデルを提案した。また、比較モデルとして、オリジナル LSTM モデルを基に CNN-LSTM モデル、エンコーダ・デコーダ LSTM モデルを構築した。第 4 章は上記の研究結果をまとめ、今後の課題などについて述べた。

Chapter 1 Introduction

1.1 Purpose of the Study

With the rapid advancement of information technology, data has shown an explosive exponential growth. Digitization has become the basic force for building a modern society and pushing us towards the era of big data. However, in such a massive amount of data, it is difficult to accurately query and analyze the potential information of the data itself, which requires more effective methods to process the data and make further decisions based on the results obtained from the processing. Machine learning is a very good method of processing data. It uses knowledge of computers, probability theory, statistics, etc. to learn by inputting data into computer programs, and finds the target function and gives predictions during the training process.

With the accelerated development of machine learning, collecting data and using machine learning technology to analyze and predict data have been promoted gradually. It is commonly utilized in research fields such as agriculture, medicine, military and finance. Machine learning is also an important means for enterprises to break through the development bottleneck in the future. Therefore, the previous traditional data information analysis system has been gradually eliminated. The new machine learning technology has become the main trend of future advancement.

Taking the stock market as an example, the stock market usually needs to generate hundreds of millions of transaction information and markets data every day. However, how to screen, extract and classify the truly valuable data among such huge data? The problem becomes one of the difficulties that current investors and trading institutions need to face. The application and promotion of machine learning can be effectively applied to such problems. Therefore, machine learning is of great significance to stock market prediction.

Traditional stock market forecasting methods include linear regression (LR), exponential smoothing (ES), autoregressive integrated moving average model (ARIMA model), etc. [60]. More recently, machine learning has been used to analyze and forecast time series data, which is currently a hot research topic, especially in the field of finance. Investors and researchers are interested in using machine learning to predict the stock market. However, some theories and hypotheses claim that since the stock market is dynamic, non-linear and affected by various factors, it cannot be predicted. Random walk theory proves that stock price changes are similar to the Brownian motion of molecules [1]. According to the efficient market hypothesis, the current price has accurately, completely and timely reflected all the information about the intrinsic value of assets [11]. It is impossible for any investor to make stable profits in the securities market by speculative behavior.

Although aforementioned scholars and researchers believe that forecasting stock prices are pointless because stock price fluctuations are random and irregular, more researchers and scholars believe that stock market movements can be forecasted to some extent. People can easily obtain a large amount of information about the stock market thanks to the advancement of computers and the Internet. Therefore, the point becomes to extract and analyze the massive information efficiently. With advances in neural networks, it has been demonstrated that certain network models can be used to extract the most valuable information from time series data, including financial data. Neural network has a powerful ability to fit complex nonlinear functions. These characteristics encourage scholars and researchers to investigate whether neural networks can forecast stock market movement or not. Machine learning methods such as support vector machine (SVM) [40], artificial neural networks [54], and other deep learning methods can also be used for predicting the stock market in the current hot artificial intelligence.

Before the vigorous development of deep learning, it is ineffective to optimize the model from the aspects of increasing the number of layers or neurons of a neural network model. Hence, we try to start from the perspective of changing the combination of activation functions. The experimental results show that changing the activation functions can improve the prediction accuracy of the model. It provides a new optimization method for optimizing the artificial neural network model that applied in stock market prediction.

With the rise of deep learning techniques, it has been proved that recurrent neural network (RNN) is an effective method of processing time series data. There is currently agreement that RNN is one of the most effective methods for efficiently processing time series data. Moreover, the long short-term memory (LSTM), which is one variant of RNN, can not only overcome the shortcomings of RNN to a large extent, but also has more advantages in processing time series data. Therefore, we applied this technique to stock market forecasting.

Most researchers have used a variety of machine learning methods to make predictions on univariate data. On this basis, we improved the LSTM model to achieve the model from univariate (single-feature) forecasting to multivariate (multi-feature) forecasting. Subsequently, we filtered the technical parameters and composed a new dataset to improve the prediction accuracy of the model in terms of changing the number of features and length of the input data. It provides a new optimization method for optimizing deep learning models applied to stock market prediction. In addition, we also built encoder-decoder LSTM [8] and CNN-LSTM [48] models based on the improved LSTM as a comparison. The encoder-decoder layer which does not perform well on time series prediction problems is more suitable for machine translation of Seq2Seq. The CNN-LSTM framework is to use to do feature extraction and description of images which are not applicable to the time series prediction problem. The experimental results show that the improved LSTM model has the best prediction performance.

1.2 Structure of the Thesis

In this thesis, we use machine learning technology such as neural network method and deep learning method to forecast financial time-series data. The dataset is mainly composed of Nikkei 225 and some individual stocks. The structure of this thesis is as follows.

In Chapter 1, the background, motivation, significance, methods and the structure of this thesis are introduced in detail. The data of Japanese stock market is processed and predicted by using different artificial neural network algorithms. Meanwhile, the literature review is also presented. The research status is organized by the development of time series problem and different prediction methods.

In Chapter 2, a hybrid ANN model is designed to forecast the Japanese Stock Market. The combination of activation functions is changed to improve the prediction performance of the hybrid model. After that, the improved model is also applied to individual stocks forecasting. Finally, we also present a preliminary investment strategy simulation experiment.

In Chapter 3, an original LSTM model is improved to achieve the model from univariate prediction to multivariate prediction. The experimental results indicate that the improved model performs well after augmenting the dataset. Based on the improved LSTM model, an encoder-decoder LSTM model and a CNN-LSTM model are established as comparison models. These models are applied to three individual stocks. The experimental results are measured by root mean square error. The complete experimental results are presented in the appendix.

The conclusion is summarized in Chapter 4. The results are concluded and analyzed in general terms, as well as new model structure design for future work.

1.3 Literature Review

In this section, the literature on stock prediction is summarized under four main categories. These categories are:

a. The traditional time series predictive analytics method: Time series forecasting is a statistical method of analyzing time series data that collected from the past period of time. The traditional methods of time series forecasting include moving average, exponential smoothing, ARIMA, etc.

b. Machine learning methods such as SVM, decision tree (DF) and random forest (RF): Most of these methods are supervised machine learning models. Whether it is a regression problem or a classification problem, the goal is to find specific associations in the input data to make effective predictions. In order to ensure the prediction accuracy of the model, the complexity of the prediction model tends to be high.

c. Shallow neural network: The structure of a shallow neural network is similar to that of logistic regression (LR), except that the number of layers of the neural network has one more hidden layer than that of LR. Back propagation and activation function are the keys to fit the objective function of the neural network. Usually the model will be more likely to obtain accurate prediction results when the following three conditions are satisfied (i) the hardware is fast enough to compute; (ii) sufficient training data; and (iii) the initial weights are set reasonably.

d. Deep learning methods: Hinton [18] introduced the concept of deep learning in 2006. Dimensionality reduction improved the flaws of BP neural networks and facilitated the development of artificial intelligence. With the rise of deep learning, neural network structures such as CNN and RNN are successful in several fields. Deep learning can express the relationship of input variables by utilizing a multi-layer neural network structure. It can fit arbitrary complex nonlinear functions and reveal a prevailing capability to analyze the crucial characteristics of input data from a given sample data. Due to the wide variety of open source deep learning frameworks, scholars are keen to use a variety of algorithm combinations to improve the prediction accuracy and convergence speed of neural networks. Appropriate network structure coupled with large datasets improves the application range and feature extraction ability of deep learning. Deep learning has achieved rapid development in various fields.

These four categories and the corresponding literature review are surveyed in more detail in following paragraphs.

Table 1 A tabular view of the stock price prediction literature

Traditional time series prediction methods	Gencay et al. [16] (MA), Abdulsalam et al. [1] (MA), Taylor et al. [55] (ES), Shiab et al.[52] (ARIMA), Ariyo et al. [2] (ARIMA),
Some other machine learning methods	Cristianini and Shawe [9] (SVM), Wang et al. [59] (DTs), Tüfekci [56] (LR, SVM and ANN), Pai and Lin [42] (ARIMA model and SVM)
Shallow neural network	Lapedes and Farber [32] (NN), Kimoto et al. [28] (NN), Yoon and Swales [62] (NN), Vui et al. [58] (ANN), Rubi et al. [48] (ARIMA and ANN), Wang et al. [61] (ESM, ARIMA and BPNN), Freisleben et al. [11] (NN), Kuo and Chen [30] (FNN), Hsu et al. [21], Qiu et al. [47] (ANN), Jigar et al. [44] (ANN, SVM, RF and naive-Bayes), Zhang and Wu [64] (NN)
Deep learning	Kim and Won [26] (LSTM), Jing et al. [24] (CNN and LSTM), Zhang et al. [63] (GAN), Hoseinzade and Haratizadeh [17] (CNN), Mehtab et al. [39] (LSTM), Bacanin et al. [4] (MLP), Mahmoodzadeh et al. [36] (LSTM)

From the perspective of traditional time series data forecasting, when predicting stock market trends, the opening and closing prices of the stock market, the fluctuations of the stock market (high, low) and trading volume of the day are usually used to train and analyze. Majority researches use the information that collecting from application programming interface or crawling data in website. In these researches, the stock price can be directly applied to different algorithm models usually. Therefore, the time series associated with prices is very important. In some articles, daily market transaction information including the price of open, high, low, close and trading volume are used as input data. Additionally, there are other studies using real-time stock market data for forecasting to support high-frequency trading.

Gencay et al. [16] used moving averages to study the linear and non-linear predictability of the Dow Jones Industrial Average. He claimed to have discovered the non-linear predictability of the stock market by using the moving averages to generate buy and sell signals for returns. Abdulsalam et al. [1] adopted moving average method to predict future values of other variables through time series data. The method was found to be capable of describing trends in stock market prices and predicting future stock market prices for three Nigerian banks. Taylor et al. [55] proposed a new adaptive approach to predict the volatility of financial returns. Shiab et al. [52] conducted a study of univariate ARIMA forecasting models using the Amman Stock Exchange general daily index and showed that Amman stock exchange most closely follows the fact that the weak form of the efficient market hypothesis. Ariyo et al. [2] used the ARIMA model to establish a stock price forecasting model, and the results showed that the ARIMA model has great potential in short-term forecasting.

Traditional time series prediction methods use statistics as the core and forecast only according to historical data, without considering the possibility of market changes.

These methods mainly use time series as the analysis factor and seldom consider the influence of other factors. Prediction results often have large deviations due to the nonlinearity of the data. These drawbacks motivate scholars to find more suitable time series forecasting methods.

With the development of machine learning, algorithms such as SVM, DT and RF have achieved better results in data processing. Compared with traditional methods such as LR and ES, these machine learning methods are effective in fitting data and prediction accuracy to a certain extent, which affirms the effectiveness of machine learning for stock market forecasting. The application of techniques such as SVM, DT and RF in stock market forecasting has been widely studied by the scientific community. these outstanding methods have shown reasonable prediction accuracy and strong robustness.

Cristianini and Shawe [9] gave a detailed introduction to SVM and give their applications. Wang et al. [59] designed a financial statement analysis using decision trees. They selected 50 financial ratios to predict the direction of earnings change. The results indicate that the bagging decision tree model works well in predicting stock returns. Tüfekci [56] accurately predicted the performance of the Istanbul Stock Exchange National 100 Index by using three machine learning methods, namely LR, SVM and ANN. Pai and Lin [42] proposed a hybrid method. They used the characteristics of ARIMA and SVM to valid the prediction performance of the proposed model through real stock price datasets.

In the early days when machine learning was introduced into the quantitative field, limited by computing power, simple models such as LR, SVM, decision trees and ANN were usually used. The references demonstrate that machine learning methods often perform better than the statistical and traditional time series analysis models. However, it is an obvious fact that the mechanism of the stock market is a nonlinear and constantly changing system that affected by various factors. It is basically impossible to rely on simple models to achieve the effect of beating the market. Hence more complex models have begun to be introduced into the quantitative field.

In 1982, Hinton proposed the BP algorithm for artificial neural networks. The BP algorithm achieves the purpose of classifying nonlinear data by backpropagating weight and threshold. The introduction of this algorithm has injected new vitality into machine learning, which in turn triggered the second frenzy of artificial neural network.

Lapedes and Farber [32] showed that nonlinear applications can be processed well by utilizing neural networks, which are concise and perform well in terms of time series data points compared to traditional methods. Kimoto et al. [28] developed some optimization algorithms and prediction approaches for the Tokyo Stock Exchange price index forecasting system based on modular neural networks. The forecasting system achieves accurate prediction results. In addition, the simulation results of stock trading show stable profits. Yoon and Swales [62] studied the ability of neural network methods to solve complex problems and compared their predictive power with that of various discriminant analysis methods. Vui et al. [58] established some specific procedure to use ANN for prediction. They surveyed some commonly used neural network hyperparameters, such as the type of neural network and the type of learning algorithm. Rubi et al. [48] applied ARIMA and ANN to the Dhaka Stock Exchange price prediction. The results showed that the estimation error of ANN is smaller than that of traditional methods. Wang et al. [61] developed a hybrid model that integrated with ES, ARIMA and BPNN to predict the stock index. The weighs parameter of the hybrid model is estimated by GA. The proposed model is superior to all traditional models, including ESM, ARIMA, BPNN, Equal Weight Mixture Model (EWH), and Random Walk Model (RWM). Freisleben et al. [11] focused on predicting stock market prices by the using BP neural networks.

In addition to use open, high, low, and close as training data for shallow neural networks, many scholars had calculated and screened out many technical indicators and economic indicators as data features and achieved good prediction results. Technical indicators can more intuitively display the information that contained in financial data. The use of technical indicators is more effective than original data. This method allows algorithms to be applied to the calculated indicator data rather than the stock price time series itself. Feature extraction is done by manual calculations, which improves data quality and simplifies machine learning models. Proponents of technical analysis believe that any other factors that can affect commodity future prices, such as fundamentals, politics and psychology are actually reflected in their prices. As introduced in the article by Bustos [7], the input data, forecasting methods, performance evaluation and technical indicators have been extensively studied and categorized and summarized in detail.

Kuo and Chen [30] proposed a fuzzy neural network optimized by genetic algorithm to simulate a knowledge base of fuzzy inference rules based on the Taiwan stock market. Hsu et al. [21] proposed extensive forecasting simulations using data from thirty-four financial indices. These simulation experiments prove that the prediction accuracy of machine learning methods is higher than the prediction accuracy of econometric methods. Qiu et al. [47] used several meta heuristic algorithms to optimize the parameters of the ANN model and predict the movement of stock market index.

Jigar et al. [44] applied four prediction models, ANN, SVM, random forest and naive-Bayes with two approaches to predict individual stocks and stock indices. The numerical experiment results show that trend deterministic data could improve the prediction accuracy of the models. Zhang and Wu [64] developed an improved bacterial chemotaxis optimization, which was then combined with a BP neural network to establish an effective predicting model for forecasting several stock indices.

However, affected by theoretical knowledge limitations and computer processing power, it cannot be proved that it is meaningful to extend neural network model to a multi-layer network. Disproportionate number of neurons and excessive number of hidden layers can lead to an overly complex model structure that cannot be trained. Moreover, when the size of the input data is too large or too complex, it can lead to overfitting and failure to converge. In the subsequent development process, researchers found that in the BP algorithm, the error will gradually disappear with the increase of the number of network layers. The problem of gradient disappearance will cause the parameter gradient of the shallow network to be 0. The discovery of this shortcoming is a fatal blow to neural network. The advancement of neural network has once again stagnated.

The publication of Hinton [18] brought deep learning back to the public's field of vision. Two novel point of opinions are proposed. (1) Deep learning models can extract data features more effectively, which has greatly contributed to the advancement of computer vision. (2) For the problem that it is hard to train the deep network structure to achieve optimum, it can be optimized by layer-by-layer training method. The result of the upper layer training is used as the initialization parameter in the lower layer training process. In addition, it proposes an approach to solve the gradient disappearance problem when training a BP neural network. The results show that neural networks with many hidden layers have very good learning ability in deep learning. In particular, Alex won the ImageNet competition in 2012 with in one fell swoop the deep learning model AlexNet [29]. This sparked a new deep learning revolution. At the same time, scholars are also trying to apply deep learning to stock market forecasting.

Kim and Won [26] combined LSTM models to several generalized autoregressive conditional heteroscedasticity (GARCH)-type models, which proved outstanding learning ability and increased the prediction accuracy of stock market. Jing et al. [24] adopted a convolutional neural network model to classify the hidden sentiment of investors extracted from major stock forums. They analyzed the technical indicators and sentiment analysis results of the stock market by applying the LSTM model. The results indicate that the hybrid method have a good performance in forecasting stock prices. Zhang et al. [63] developed an original generative adversarial network architecture with a multilayer perceptron (MLP) as the discriminator and an LSTM as the generator for predicting stock closing prices. The simulation results indicate that the generative adversarial network can achieve better performance in forecasting closing price than the other models. Hoseinzade and Haratizadeh [17] proposed a CNN-based neural network that can process and extract features from various sources datasets. They claimed that the predictive performance has been significantly improved.

Many researchers have developed different forecasting methods using machine learning for financial data prediction. Simulations are established to analyze the forecasting performance and efficiency of the models. Mehtab et al. [39] developed a LSTM model to build four deep learning-based regression models to predict the future NIFTY 50 opening price. By optimizing the hyperparameters of the model and adjusting the network structure, they enhanced the predictive capability of the model to process data with different structures. The results clearly illustrate that the LSTM-based univariate model is the most accurate model for predicting the next week's opening value of the NIFTY 50 time series using one-week prior data as input. Bacanin et al. [4] conducted a study using the Borsa Istanbul 100 index as an example and proposed a MLP to predict the movements of stock index. The study suggested that a hybrid algorithm of a MLP and a modified whale optimization algorithm are effective in improving the accuracy of the model. Mahmoodzadeh et al. [36] developed an LSTM model that improved by Gray Wolf Optimization. The effect of the length of input data on the model performance and LSTM model with multiple parameters were investigated using several datasets.

However, deep learning has been widely used with the advancement of model architecture, the development of deep learning is still limited by inherent shortcomings. The impact of these factors should be taken into account when using deep learning models for prediction and analysis. The shortcomings are summarized as follows:

① Lacking of effective data can lead to overfit;

② Gradient vanishing and exploding;

③ Neural network requires a great amount of parameters, such as network topology and weight initialization;

④ The learning method is a black box. It is inexplicable to explain which part can impact the reliability and acceptability of the model;

⑤ Excessive training time and the purpose of learning may not even be achieved.

Although scholars have applied various machine learning methods to stock market forecasting, not many people have combined different models and architectures to LSTM models to explore the impact of univariate or multivariate inputs on model prediction accuracy. The overarching goal of this study is to use several kinds of artificial neural network algorithms to process stock market data with univariate or multivariate and predict short-term stock prices. The effects of different activation functions on the original ANN model are explored, and the study shows that the selection of appropriate activation functions can effectively improve the prediction accuracy of the model. With the advancement of neural networks, LSTM has achieved excellent performance in the prediction of time series data. After improving the basic LSTM model, we establish the encoder-decoder LSTM model and the CNN-LSTM model as comparison and explored the performance of these models that trained with different features and length of the datasets.

Chapter 2 Predicting Direction of Individual Stock Price Movement Using a Hybrid Model

2.1 Introduction

Predicting stock market movements is a challenging and attractive task because predicting stock market movements plays a guiding role in formulating investment strategies. Since the stock market is dynamic and highly nonlinear, traditional time series forecasting methods are not suitable for stock market prediction, while artificial neural networks can detect and fit the relationship between nonlinear data. Consequently, this paper will study the problem of stock market movement prediction based on artificial neural networks. With advances in technology, scholars and researchers are trying to extract the most valuable technical indicators from verbose data to explain and forecast the movement of stock market. It also plays an important role to increase returns and reduce risks for financial institution and retail investors [14], [35].

Artificial neural network (ANN) is a powerful tool for nonlinear dynamic system predicting and modeling. It is possible to predict the stock market efficiently by using appropriate mathematical models. In practical application, various ANN models use error back propagation (BP) algorithm. However, in the process of finding the optimal solution, the neural network is apt to be stuck in a local optimal solution. To overcome these drawbacks, Qiu and Song proposed an ANN model to forecast direction of Nikkei 225 [47], [46]. The weighs and bias of this model are optimized by Genetic Algorithm (GA) to get over the drawbacks that mentioned above. Hence this ANN model is also called a hybrid model.

In this chapter, we conduct some simulation experiments about changing the combination of activation function to improve original hybrid model. As revealed in the results, it is efficient to improve the prediction accuracy by changing the combination of activation function. Then we expanded the application of the improved model from stock market index to individual stocks. Additionally, an investment strategy is proposed based on the prediction results of individual stocks.

This paper is organized as follows. The second section introduces the previous studies. The back propagation algorithm and genetic algorithm are described briefly. The structure and parameters of artificial neural network are illustrated next. In Section 3, we show the performance of different activation function in improving original model and we apply the improved model to some individual stocks and get some good results. An investment strategy based on the results of prediction is described in Section 4. Finally, we summarize the results and their possible consequences for this chapter in Section 5.

2.2 Prediction Model and Parameters

A. Back Propagation Algorithm

The BP algorithm is a multilayer feedforward neural network and it is one of the most widely used neural network models [25]. Its main feature is that the signal is propagated forward and the error is propagated backward.

First, the error between the actual value and the predicted value is calculated. Then the error generated by each layer of neural networks is diffused from the back to front. Next the weights and biases of the gradient are calculated and updated by BP algorithm.

Training stops when the value of the error function has become sufficiently small, or reached a prespecified number of epochs have expired.

B. Genetic Algorithm

BP network is the core part of forward network, but there are some shortcomings, such as learning convergence is too slow, convergence to global optimal solution is not guaranteed and the network structure is not easily determined [27]. These defects had been proved in previous studies. GA optimizes the BP neural network by three ways: network structure parameters, GA optimization and model prediction [37], [41]. The optimized initial weights and biases of the BP network are obtained by multiple iterations of the GA. After that, the training process of neural network model can avoid falling into local optimal solution.

The initial weights and biases in the BP algorithm are used as the GA's individual gene values. The length of the individual is the number of weights and biases in the BP neural network. Each gene represents a weight or biases, and the value on the gene is the real value of the connected weights or biases in the BP neural network. This constitutes a chromosome in the GA. A certain number of chromosomes are used as the initial population for the training of the GA and then an optimal individual is obtained after iterative processes such as selection, crossover and variation operations of the GA. Finally, the optimized individual is used as the initial parameter of the BP network for training. The BP neural network prediction uses the GA to obtain the optimal individual to assign the initial weights and biases to the network [23].

Simulation results demonstrate that using GA to optimize BP neural network can speed up the convergence of the model and reduce the operation time and the number of iterations. For these reasons, first we use the GA to optimize the initial parameters of the model and then input the improved parameters to the BP neural network model for stock prediction.

The flow chart of the hybrid model is shown in Figure 1.

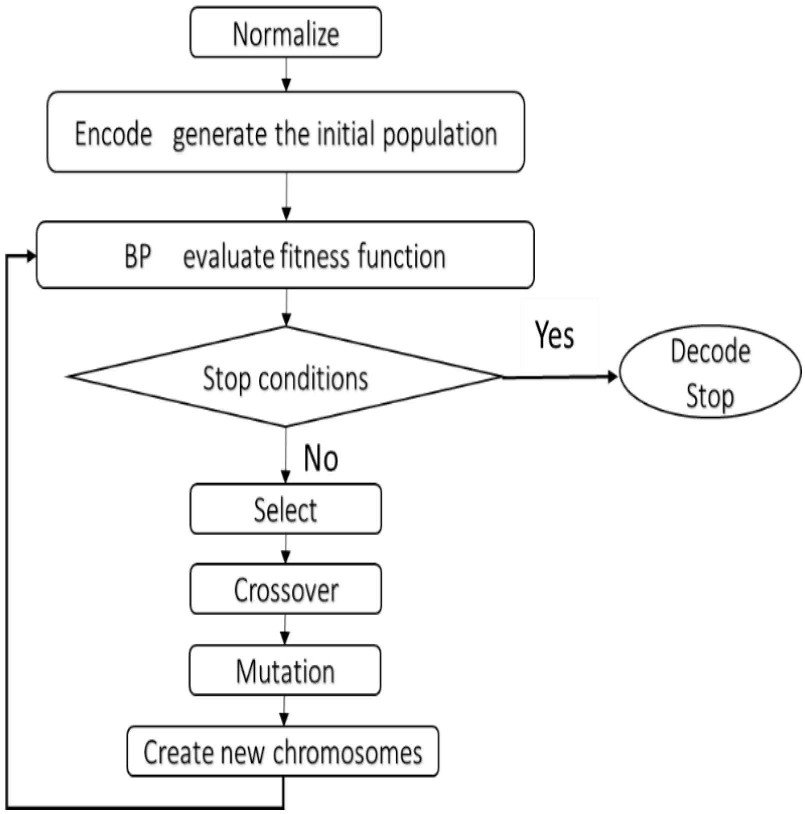

Figure 1 Flow chart of the hybrid model

C. The Design and Establishment of Experiment Model

By using the BP neural network, the primary prerequisite is that there are enough good typicality and high precision samples. To forecast the movement of some individual stocks by using ANN model, we collected the data from January 2008 to June 2018, including about 2400 trading days of sample data. We show the data of Panasonic company as example in Figure 2, which can illustrate the movement of Sony stock prices clearly. Horizontal axis is the daily closing price of Panasonic stock. Vertical axis is the data sequence.

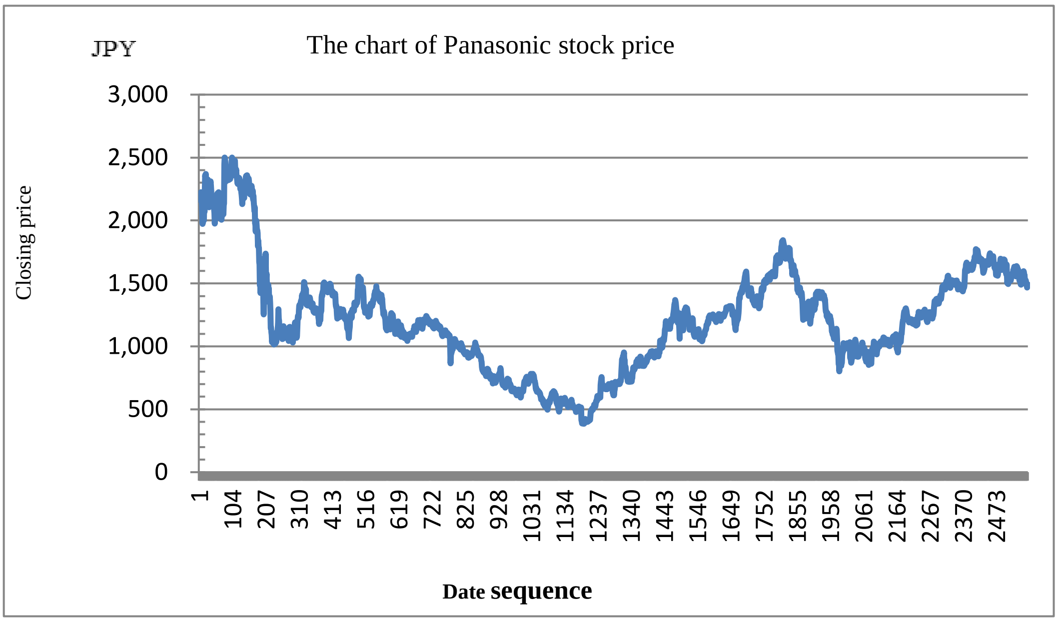

Figure 2 The historical data of Panasonic

In the light of previous researches, we use stock market information and a variety of technical indicators as input variables to predict stock market movements using neural networks. Hornik, Stinchcombe and White [20] have revealed that neural networks can fit any unknown function and that the number of layers of hidden layers need not exceed one.

Accordingly, the ANN model is established by utilizing BP algorithm which is most commonly studied now. The GA is used to optimize the initial weights in order to overcome the disadvantages of BP algorithm [3]. We reference this prior study [46] to select technical indicators as feature subsets.

On Balance Volume (OBV) measures buying and selling pressure as a cumulative indicator that adds volume on up days and subtracts volume on down days. It is also an indicators for measuring positive and negative volume flow. A rising OBV reflects positive volume pressure that can lead to higher prices. Conversely, falling OBV means that negative volume pressure can foreshadow lower prices.

BIAS_n measures the divergence of current stock price from an n day simple moving average of the stock prices. Normally scholars and investors choose n to be 6 days. The value of BIAS_n may be above or below the moving average when the closing price is far away from the average level.

ASY_n is the average return in the last n days.

We use OBV, MA_5, BIAS_6, BIAS_6, ASY_5, ASY_4, ASY_3, ASY_2, ASY_1 and the standardized closing value as input variables to predict the closing value of next period. The selected technical indicators' formulas and computational formulas are shown in Table 2.

Table 2 Selected technical indicators

Name of indicator	Formulae
OBV	$OBV_t = OBV_{t-1} + \Theta * V_t$
MA_5	$MA_5 = (\sum_{i=1}^{5} C_{t-i+1})/5$
BIAS_6	$\mathrm{BIAS}_6 = \left(\frac{C_t - MA_6}{MA_6}\right) \times 100\%$
PSY_{12}	$PSY_{12} = (A/12) \times 100\%$
ASY_5	$ASY_5 = (\sum_{i=1}^{5} SY_{t-i+1})/5$
ASY_4	$ASY_4 = (\sum_{i=1}^{4} SY_{t-i+1})/4$
ASY_3	$ASY_3 = (\sum_{i=1}^{3} SY_{t-i+1})/3$
ASY_2	$ASY_2 = (\sum_{i=1}^{2} SY_{t-i+1})/2$
ASY_1	$ASY_1 = SY_{t-1}$

The prediction model is consisted of an input layer, a hidden layer and an output layer. Therefore, in this experiment, the numbers of input layer, output layer and hidden layer are 10, 1 and 20. The architecture of the ANN model is shown in Figure 3.

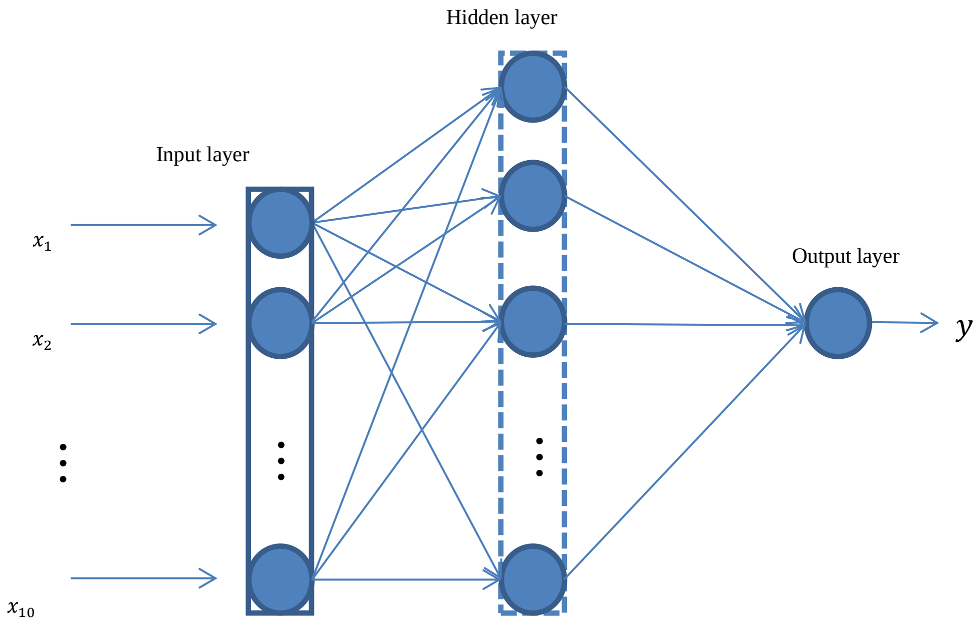

Figure 3 Architecture of the ANN model

Because the detailed value can't be predicted precisely, we use the value of next day to subtract the value of today to express the direction of individual stock price. If the result is positive, that means the stock is increasing. Otherwise, the stock is decreasing.

Through a prior study, we screened the technical indicators. These technical indicators and an array of stock market information were then combined to form the original data set. Before being fed into the model, the data had to be normalized to prevent too large or too small values from affecting the forecast results. We normalize the data as follows:

$$XN = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (1)$$

The new chromosomes are created by using GA (e.g., crossover, mutation). The iterations will stop when reach the stop conditions.

2.3 The Prediction of Individual Stocks

As mentioned before, first, we gather the experiment data from Yahoo! Finance website. Second, the original data is standardized before being applied to the ANN model. Because some values are too large and others are too small. By standardization, all of the value is limited in 0 to 1. Third, we select the training data (200 days) as input variables and select the testing data (30 days) as target data. When the neural network has been trained completed, we input the test data to test the performance of neural network models. We use value of today to minus value of the day before to see the stock market will increase or decrease. Consequently, the output variable can also represent the predicted direction of the daily stock market price. If predicted direction is same with true direction. That means we predict it correctly. Because the testing data is 30days, therefore the hit ratio is the times of predicting correctly divided by 30. Here, we present the results experiment in tables to read and understand data clearly.

In order to explore the instability of the ANN model, we have tried plenty of experiments about activation functions. Based on the results of numerical experiments, the activation function “logsig” is chosen as the best activation function in this hybrid model. The best hit ratio is 86% in the Japanese stock market. It is 6% higher than the original model, where the activation function is “Tansig/Logsig”. The combination of activation functions and results are shown in the Table 3.

Table 3 The results of different activation function

Training period (200 days)	Testing period (30 days)	Hit ratio of Sony Corporation	Hit ratio of Panasonic Corporation	Hit ratio of Toyota Corporation
2008.1~	2008.10	86%	76%	73%
2008.12~	2009.8	83%	80%	71%
2010.4~	2011.3	59%	54%	54%
2012.2~	2012.12	78%	79%	79%
2014.9~	2015.6	79%	80%	76%
	Average ratio	79.6%	77.6%	70.6%

After this experiment, we apply the improved model on some individual stocks. The procedures are same with the previous experiment. The results of individual stocks are shown in Table 4.

Table 4 The predicted results of individual stocks

Activation function	Hit ratio (%)	Feature
Tansig/Logsig	71.11%	Non-linear
Purelin/Purelin	51.22%	Linear
Logsig/Logsig	86.22%	Non-linear
Tansig / Tansig	62.77%	Non-linear
Logsig / Purelin	65.00%	Non-linear

The first column is the training period and the second column is the testing period. The length of the two periods is same and continuous. The last three columns are the hit ratios of Sony corporation, Panasonic corporation and Toyota corporation respectively. They are 79.6%, 77.6% and 70.6%. There is an unusual phenomenon. When the testing period is 30 continuous days from 2011.3, all of three stocks' prediction results are not exceeding 60%. We analyzed about the phenomenon and ascribed reason to the great east Japan earthquake occurred in 2011.3. This result proves that this experiment model maybe lose efficacy when the incidents happened

2.4 The Profit and Investment Strategy

We use Table 5 as an example to illustrate.

The first two columns are opening value and closing value of original data that we had collected from yahoo finance. The third column is output variable. Based on the prediction result, we can know the direction of the individual stocks. We use 1 to notate "the stock will increase". We use -1 to notate "the stock will decrease". If the stock will increase, we buy stocks at the opening value and sell stocks at the closing value. If the stock will decrease, we do a short selling, and cover the short at the closing value.

Table 5 The example of investment strategy

Open value (JPY)	Close value (JPY)	Actual direction	Predicted direction	Profit (JPY)
1131	1091	0	0	0
1085	1034	1	-1	51
1050	1049	1	1	-1
1064	1083	1	1	19
1092	1070	-1	1	-22
1089	1066	-1	1	-23

There are eleven groups of prediction experiment of this stock. The profit result of Panasonic is shown in Tables 6(a) and 6(b). The testing periods are among January 2008 to June 2018. These tables consists testing period, hit ratio, profit and return rate. When invest capital is JPY 500,000 in every testing period, the average profit and average return rate are JPY 79,463 and 15.9% respectively. The stock trading is daily trading and without considering the handling fee.

Table 6 The investment result of Panasonic

(a)

Period	2008.10	2009.8	2010.6	2010.11	2011.3
Hit ratio	72.3%	76.2%	82.2%	80.6%	53.9%
Profit (JPY)	80,015	85,252	77,489	87,384	72,297
Return rate	16.0%	17.1%	15.5%	17.5%	14.5%

(b)

Period	2012.12	2013.10	2014.8	2015.6	2016.4	2017.1
Hit ratio	73.2%	72.3%	79.6%	65.9%	70.1%	81.0%
Profit (JPY)	72,581	105,236	30,378	72,468	126,951	64,050
Return rate	14.5%	21.0%	6.1%	14.5%	25.4%	12.8%

2.5 Conclusion

In this paper, we applied an artificial neural network for predicting the direction of some individual stocks. In order to improve the accuracy of the model, experiments were conducted to found out which activation functions have a good performance to get a better hit ratio. In this experiment, the best performance of activation functions of the hybrid model is “Logsig/Logsig”. The hit ratio is 6% higher than original model.

As shown above, prediction accuracy of the hybrid model with new pair of activation functions outperforms the one that is trained by original activation functions. By utilizing improved model, the hit ratio of Sony corporation and Panasonic corporation and Toyota corporation are 79.6%, 77.6% and 70.6% respectively.

An investment strategy is established based on the numerical experiment results. Experimental tests indicated that the investment strategy can deliver returns while avoiding investment risk. To get a better neural network structure, we need to fine tune neural network artificially. It is a very laborious and time-consuming work. We propose to utilize CNN (Convolutional Neural Network), RNN (Recurrent Neural Network) or some others variations of neural networks to continue our research in the future work.

Chapter 3 Applying LSTM Model to Predict the Japanese Stock Market with Multivariate Data

3.1 Introduction

In real life, collecting data for a long or short period is a common activity. Time series data are abundant in research fields such as agriculture, medicine, the military, and finance. It is of great theoretical significance and practical application value to analyze these data to forecast long-term trends or to perform other forms of analysis. Traditional methods for stock market prediction include linear regression, exponential smoothing, autoregressive integrated moving average model (ARIMA model) and others [60]. Recently, machine learning is being used to analyze and forecast time series data, which is a hot research topic right now, particularly in the financial field. Investors and researchers are interested in using machine learning to forecast the stock market. However, some theories and hypotheses claim that the stock market cannot be forecasted because it is dynamic, nonlinear, and influenced by various factors [42]. The random walk theory demonstrated that the changes in stock prices are analogous to the Brownian motion of a molecule [1]. According to the efficient market hypothesis, current prices have accurately, completely, and timely reflected all the information about the intrinsic value of assets [11].

Although some scholars and researchers believe that forecasting stock prices are pointless because stock price fluctuations are random and irregular, many researchers and scholars believe that stock market movements can be forecasted to some extent.

People can easily obtain a large amount of information about the stock market thanks to the advancement of computers and the Internet. Therefore, extracting and analyzing the information to support making investment strategies have become difficult work. With advances in neural networks, it has been demonstrated that certain network models can be used to extract the most valuable information from time series data, including financial data. These characteristics encourage scholars and researchers to investigate whether neural networks can forecast stock market movement or not. Machine learning methods such as support vector machine (SVM) [40], artificial neural networks [50] and other deep learning methods can also be used for predicting the stock market in the current hot artificial intelligence. There is currently agreement that the recurrent neural network (RNN) is one of the most effective methods for efficiently processing time series data. Moreover, the long short-term memory (LSTM), one RNN variant, not only surmounts the defects of RNN but also improves prediction accuracy.

Many researchers have developed different forecasting methods using machine learning for financial time series forecasting. Experiments are conducted to analyze the forecasting performance, effectiveness and efficiency of the models. Mehtab et al. [39] proposed a LSTM network to build four deep learning-based regression models to predict the future NIFTY 50 opening price. By optimizing the hyperparameters of the model and adjusting the network structure, they enhanced the predictive ability of the LSTM model to handle data with different structures. The results clearly show that the LSTM-based univariate model is the most accurate model for predicting the next week's opening value of the NIFTY 50 time series using one-week prior data as input. Bacanin et al. [4] conducted a study using the Borsa Istanbul 100 index as an example and proposed a MLP to predict the movements of stock index. The study suggested that a hybrid algorithm of a MLP and a modified whale optimization algorithm are effective in improving the accuracy of the model. Mahmoodzadeh et al. [36] developed an LSTM model that improved by Gray Wolf Optimization. The effect of the length of the input time series on the model performance and LSTM model with multiple parameters were investigated using several datasets. Qiu et al. [45] used several meta heuristic algorithms to optimize the parameters of the artificial neural network model and predict the direction of stock market index movement. Jigar et al. [44] applied four prediction models, ANN, SVM, RF and naive-Bayes with two approaches to two stocks and two stock price indices. The experimental results suggest that the performance of the prediction models can be improved when these technical parameters are represented as trend deterministic data.

Although scholars have applied various machine learning methods to stock market forecasting, not many people have combined different models and architectures to LSTM models to explore the impact of univariate or multivariate inputs on model prediction accuracy. The overarching goal of this study is to develop LSTM model that can process input data with univariate or multivariate inputs and predict short-term stock prices. Furthermore, encoder-decoder LSTM and CNN-LSTM are established for comparison experiments based on LSTM model.

It is common in the machine learning process for the model to be unable to be effectively trained due to a lack of training data. Accordingly, we made two efforts to improve the dataset quality to improve the model prediction accuracy. The original dataset consisted of trading data from the Nikkei 225 index and some individual stocks obtained from Yahoo finance. We calculate and analyze several well-known technical indicators based on historical stock data before developing LSTM model, encoder-decoder LSTM model and CNN-LSTM model to forecast stock price trends in the future. We combine historical trading data and technical indicators as feature engineering and then adjust network structure and hyperparameters to achieve these models from univariate prediction to multivariate prediction. The results indicate that the encoder-decoder layer and CNN-LSTM framework do not perform well in stock market prediction problem.

The remaining parts of this chapter are divided into four sections. Section 2 surveys why LSTM can extract useful information from verbose data better than RNN. Section 3 describes the experiments and their outcomes. This section also includes information on data processing and feature engineering. Section 4 discusses the model's results. Finally, section 5 summarizes the contributions.

3.2 Methodology

This section first contrasts RNN and LSTM by using the external input of a single neuron as an example. Then, we briefly discuss how LSTM overcomes RNN's shortcomings. Subsequently, the internal structure, variables and equations of LSTM neurons are described in detail.

RNN is one of the neural networks that can retain information and pass it on to the next layer [13] [9]. RNN, unlike ordinary neural networks, is very effective for time series data. This type of information reflects the state or degree of change in a particular thing or phenomenon over time. The information extraction of time series data is realized through the sharing of parameters at different times due to the neuron's memory. Consider, as an example, a single-RNN neuron structure (Figure 4). x_t represents the input information at the current t moment; y_t represents the output information of the RNN neuron at the current t moment; h_{t-1} and h_t are the state information stored by the neuron at the previous moment and the neuron at the current t moment respectively.

We can see from equation (1) that the state information of neuron h_t is determined by both x_t and h_{t-1}. That is how the RNN remembers the previous information and applies it to the current output calculation.

$$h_t = tanh(w_h \cdot [h_{t-1}, x_t]). \quad (1)$$

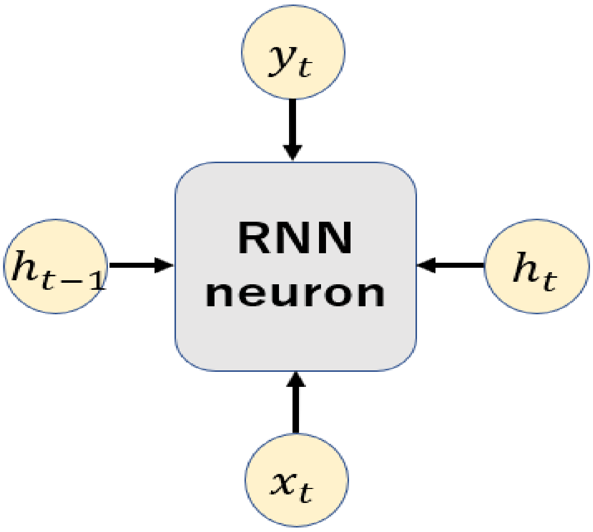

Figure 4 RNN neuron

However, this ability will be diminished when the outputs require more information from the past. Bengio, et al. [6] investigated this issue in-depth and discovered some fundamental reasons why RNN training is difficult. The RNN will include many accumulative multiplications of activation functions during the iteration. If the tanh or sigmoid function is used as the activation function, many decimals will be multiplied. With the increasing iteration numbers and time series, the accumulative multiplications of decimals cause the gradient to become smaller and smaller until it is close to 0, which is the phenomenon of the vanishing gradient; if the rectified linear unit (ReLU) is used as the activation function, the left derivative of the ReLU function is 0 and the right derivative is always 1, which avoids the occurrence of vanishing gradient. However, the derivative of constant 1 can easily result in an exploding gradient. By improving RNN, LSTM [15],[22], is introduced to eliminate the long-term dependency defects of the RNN.

A cell state C_t was added to LSTM neurons to overcome the vanishing gradient problem compared to input variables outside the RNN neuron. Figure 5 shows the internal structure of one LSTM neuron to better illustrate how information is transmitted through the LSTM neuron.

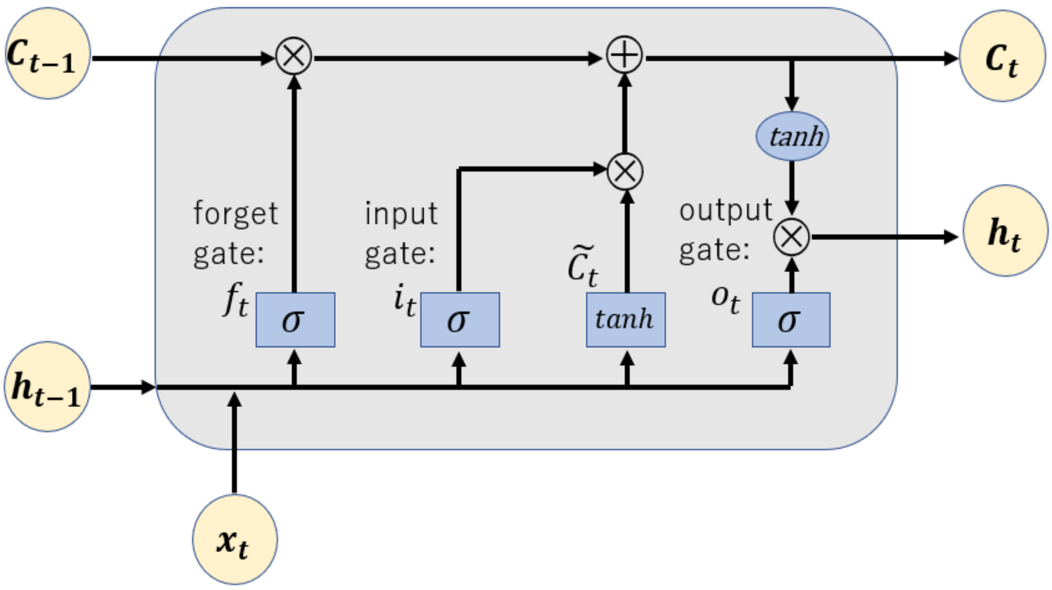

Figure 5 The internal structure of LSTM neuron

The following section goes into great detail about Figure 5. Three gates with sigmoid activation functions and one hidden layer with tanh activation functions replace the normally hidden layer in the RNN. The definitions and process steps are shown below. The functions are illustrated as follows.

x_t: input value (vector)
h_t: output value (vector)
C_t: neuron state
$\widetilde{C_t}$: the neuron state being updated
σ: sigmoid function
tanh: hyperbolic tangent function
f_t: a vector that determines how much information discard or remains in forget gate
i_t: a vector that determines what information updates in the input gate
o_t: a vector that determines what information output

Forget gate: The first step in an LSTM neuron is to determine what information must be discarded from the neuron state. This section of the operation is handled by a sigmoid unit known as the forget gate. According to the input information of h_{t-1} and x_t, it outputs a vector between 0 and 1 and the 0~1 value in the vector indicates how much information in the neuron state C_{t-1} is retained or discarded. A value of 0 indicates that no information is reserved, while a value of 1 indicates that all information is reserved. *W* and *b* are weight vector matrices and gate biases, respectively, in the equation (2).

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t]\right) + b_f. \tag{2}$$

Input gate and tanh layer: The following step is to decide what new information should be added to the neuron state. This section of the procedure is divided into two steps. First, h_{t-1} and $x_t$are used to determine which information is updated through a sigmoid unit known as the input gate. Then, through the tanh layer, use h_{t-1} and x_t to generate a new vector that can be updated to the neuron state. The equations (3~5) are shown below:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t]) + b_i. \tag{3}$$

$$\widetilde{C}_t = tanh(W_c \cdot [h_{t-1}, x_t]) + b_c. \tag{4}$$

$$C_t = f_t * C_{t-1} + i_t * \widetilde{C}_t. \tag{5}$$

Output gate: After updating the neuron state, determine what information is output based on h_{t-1}, C_{t-1} and x_t. The input information is routed through a sigmoid unit known as the output gate. Meanwhile, the updating neuron state C_t is processed by the tanh unit to produce a vector between −1 and 1. The vector obtained from the output gate is multiplied by this vector. Finally, the LSTM neuron's output h_t is obtained by equations (6) and (7).

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t]) + b_o. \tag{6}$$

$$h_t = o_t * tanh(C_t). \tag{7}$$

With the above three gates, the LSTM can process time series data by discarding or retaining input information.

The long-distance information gradient does not disappear during iteration using the backpropagation algorithm, and the gradient can be well transmitted in the LSTM, greatly reducing the likelihood of a vanishing gradient. Inspired by the LSTM neural network's superiority, we established an improved LSTM model for forecasting short-term stock prices. The two variants of LSTM, encoder-decoder LSTM and CNN-LSTM are set as comparison experiments. We conduct some numerical experiments to verify and evaluate the validity of these LSTM models by applying these models to the Nikkei 225 and several individual stocks for short-term forecasts.

Additionally, we supplement the original dataset with technical indicators. However, training data from both the original and augmented datasets are expanded. In these numerical experiments, we want to see if these two factors can improve the model's prediction accuracy while also improving data quality.

3.3 Experiments and Results

3.3.1 Preliminary Work

Yahoo finance was used in this study to collect publicly available market information on stocks. The dataset consists of the Nikkei 225 and several individual stocks, including open, high, low, close prices and trade volume. Data cleaning and data standardization are two major steps in data preprocessing before entering data into the model [38]. Data cleaning is primarily used to fill in missing values and remove outliers from a dataset. Data standardization is the scaling of data values into a specific period (0~1). Due to the different nature of each evaluation index in the multi-index evaluation system, it usually has different orders of magnitude and units. When the values of various indicators vary widely, if the original data are directly used for analysis, the indicators with higher values will play a larger role in the analysis, while the effects of indicators with lower values will be weakened. Therefore, standardizing the original data must ensure the reliability of the results.

After preprocessing the stock data, we compute some technical indicators that are commonly used in the stock market as feature engineering. Scholars generally agree on one point: data and features determine the upper limit of machine learning, and models and algorithms only approximate this upper limit. Therefore, feature engineering is critical for improving data quality in data preprocessing. Feature engineering is the process of extracting features from original data that can be used to better represent the actual problems dealt with by the prediction model. Good features can simplify the model while also improving its robustness and generalization. They can also improve the model's convergence speed and prediction accuracy at the same time. In this study, the 10-day moving average (MA10), 10-day relative strength index (RSI10), on balance volume (OBV), and price movement direction are calculated as features.

The MA10 is the average closing price of a stock in the market over the previous 10 days, and it is significant because it reflects the stock's 10-day average price. It is a critical indicator line for determining stock trends [53]. The stock trading operation reflects the aggregate results of various factors. Ultimately, price changes are determined by the relationship between supply and demand. The RSI indicator calculates the percentage of the total increase in stock prices to the average of the total change in stock prices over a given period using the supply and demand balance principle. The RSI value ranges from 0 to 100. The stronger the uptrend is, the higher the RSI value, and vice versa. The trend of changes in trading volume is used to infer the trend of stock prices. The direction of price movement is calculated by subtracting today's close value from the close value one day ahead. If the difference value is positive, we label price movement with 1. Conversely, if the difference value is negative, we label price movement with −1.

The prediction results of NIKKEI 225 and three individual stocks will be presented in tabular form in this section. We use root mean square error (RMSE) to estimate how well the model predicted for the model evaluation. It is the most popular measure for evaluating the accuracy of the model. RMSE is defined mathematically as follows:

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{n}\left(X_i - x_i\right)^2}{n}}\,. \tag{8}$$

3.3.2 Model

We conduct some numerical experiments to verify and evaluate the validity of these LSTM models by applying these models to the Nikkei 225 and several individual stocks for short-term forecast. First, we build a basic LSTM model on the Nikkei 225 and several individual stocks for short-term forecast. The structure diagram of the improved LSTM model is illustrated in Figure 7. These results are used as a benchmark for comparison. Among the time series data processing, the time series prediction problem with series as input and series as output is a more challenging prediction problem. The encoder-decoder LSTM has good performance in dealing with this type of problem, so one encoder layer and one decoder layer are added to the basic LSTM model to test the performance of encoder-decoder LSTM in predicting short-term stock market movements. We then combined the CNN with the basic LSTM model to test whether the CNN could improve the prediction results of the model.

Figure 6 is the loss function of the improved LSTM model during parameter tuning. This LSTM model is trained with Nikkei 225 and OHLCV+ dataset (See Sec. 3.3.3 Experiments for detail.). As shown in the figure, after epoch=100, there is no longer a significant decrease in the loss function, so the epoch is set as 100 for the efficiency of model training and preventing overfitting. The parameters of the basic part of three models are listed in Table 7.

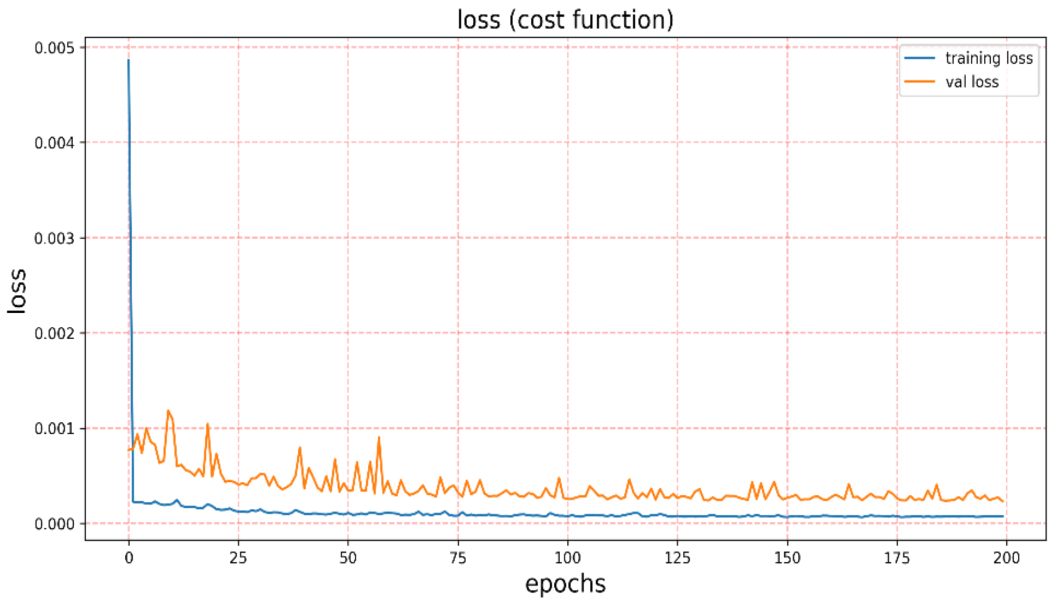

Figure 6 The loss function of NIKKEI 225

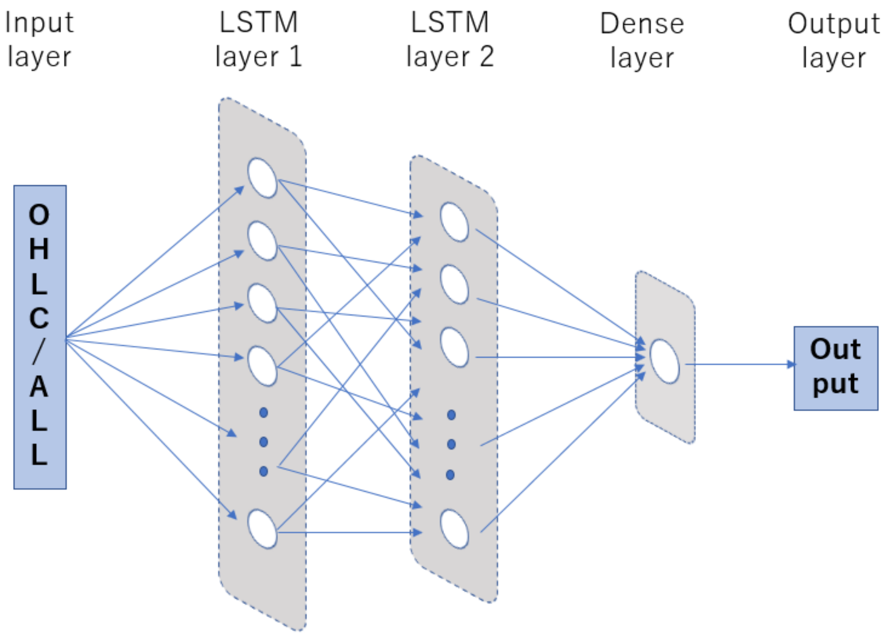

Figure 7 The structure diagram of the LSTM model

Table 7 Parameters of the basic part of models

Timestamp	20
Epochs	100
Batch size	16
Unit of the first LSTM layer	64
Unit of the second LSTM layer	32
Dense layer	1
Optimizer	adam

3.3.3 Experiments

The dataset is approximately 2,800 trading days long and was collected from January 2011 to June 2022. As an example, consider the NIKKEI 225 historical data chart (Figure 8). Since the LSTM network can extract useful information from verbose data without being constrained by long-term dependency defects, the dataset is divided into four parts to train this LSTM independently to forecast the short-term stock market. The training period is set at 600, 400, and 200 days respectively. The test period is a fixed 30-day period that follows each training period. These numerical experiments are used to investigate the effect of different length train periods on the performance of these LSTM models.

Figure 8 The historical data of NIKKEI 225

Furthermore, some contrast experiments are conducted to verify if these technical indicators are effective in improving the prediction accuracy of models. The detailed experimental procedure is as follows. As mentioned above, the RMSE is used as performance measure.

We begin to use the dataset with only univariate feature "close price" to train three models to verify the performance of these models in processing univariate input data. These results are used as a benchmark for comparison. Subsequently, we use the dataset which includes open, high, low, close and trading volume (OHLCV for short) as the input data to explore the performance and predictive ability of the models in processing multivariate input data. Finally, the OHLCV dataset and technical indicators are combined into a new dataset called “OHLCV+” for training and predicting. Based on these results, we use OHLCV+ dataset to explore whether the selected technical indicators can improve the prediction performance of the models.

In order to compare the performance of the models, we use the same dataset to train the LSTM model and its two variants (encoder-decoder LSTM and CNN-LSTM) separately. Except for the special layers of the model itself (encoding layer, decoding layer, convolution layer), the rest of the parameter settings are the same. The epoch is 100. The batch size is 16. The root mean squared error is chosen as loss function. The optimizer is adam [49]. The input data and prediction period are also set to be the same at the same period. These three groups of experimental results are summarized in section 3.4.

3.3.4 Results

The experiment results of Nikkei 225 are illustrated in Tables 8 to 11. The second column shows the length of the various training periods. The last three columns show the results of different LSTM models respectively. As previously stated, the test period is a fixed 30-day period that follows each training period. The model's prediction results using closing price as input data are shown in the row of "RMSE of closing price" in Table 8. The model's prediction results using OHLCV as input data and the model's prediction results when OHLCV+ is used as input data are shown in the rows of "RMSE of OHLCV" and "RMSE of OHLCV+" respectively.

Table 8 Results of first Prediction period (NIKKEI 225)

	Length of training period(days)	The improved LSTM	Encoder-Decoder LSTM	LSTM -CNN
RMSE of closing price	600	0.0125	0.0351	0.0618
	400	0.0172	0.0368	0.0649
	200	0.0233	0.0587	0.0541
RMSE of OHLCV	600	0.0198	0.0347	0.0698
	400	0.0200	0.0407	0.0758
	200	0.0246	0.0561	0.0554
RMSE of OHLCV+	600	0.0136	0.0224	0.0677
	400	0.0132	0.0429	0.0682
	200	0.0183	0.0346	0.0615

Table 9 Results of second Prediction period (NIKKEI 225)

	Length of training period(days)	The improved LSTM	Encoder-Decoder LSTM	LSTM -CNN
RMSE of closing price	600	0.0107	0.0346	0.0678
	400	0.0147	0.0364	0.0741
	200	0.0178	0.0413	0.0916
RMSE of OHLCV	600	0.0089	0.0363	0.0635
	400	0.0103	0.0388	0.0654
	200	0.0159	0.0391	0.0769
RMSE of OHLCV+	600	0.0138	0.0247	0.0406
	400	0.0150	0.0302	0.0456
	200	0.0172	0.0614	0.0496

Table 10 Results of third Prediction period (NIKKEI 225)

	Length of training period(days)	The improved LSTM	Encoder-Decoder LSTM	LSTM -CNN
RMSE of closing price	600	0.0138	0.0218	0.0283
	400	0.0158	0.0225	0.0366
	200	0.0185	0.0240	0.0495
RMSE of OHLCV	600	0.0145	0.0227	0.0287
	400	0.0168	0.0213	0.0333
	200	0.0191	0.0220	0.0425
RMSE of OHLCV+	600	0.0111	0.0196	0.0201
	400	0.0125	0.0183	0.0277
	200	0.0169	0.0193	0.0227

Table 11 Results of fourth Prediction period (NIKKEI 225)

	Length of training period(days)	The improved LSTM	Encoder-Decoder LSTM	LSTM -CNN
RMSE of closing price	600	0.0131	0.0162	0.0284
	400	0.0100	0.0191	0.0395
	200	0.0114	0.0209	0.0563
RMSE of OHLCV	600	0.0106	0.0271	0.0655
	400	0.0116	0.0451	0.0704
	200	0.0473	0.0288	0.1134
RMSE of OHLCV+	600	0.0083	0.0167	0.0224
	400	0.0098	0.0183	0.0262
	200	0.0100	0.0206	0.0432

Due to limited space, we summarize the prediction results of the experiments of Itochu, Toyota and Sony respectively and present the RMSE values in the form of periods for the experimental results, as shown in Tables 12~14. The first column is the length of training period. The second column is the training model. The prediction results of different features are shown in last three columns. We have summarized all the prediction results of individual stocks in the appendix.

Table 12 Prediction Results of Itochu

Length of training period(days)	Model	Features		
		CLOSE	OHLCV	OHLCV+
600	LSTM	0.0054~0.0086	0.0060~0.0082	0.0048~0.0069
	Encoder-Decoder LSTM	0.0079~0.0164	0.0089~0.0168	0.0090~0.0164
	CNN-LSTM	0.0190~0.0334	0.0163~0.0317	0.0153~0.0259
400	LSTM	0.0057~0.0089	0.0050~0.0097	0.0062~0.0072
	Encoder-Decoder LSTM	0.0067~0.0157	0.0071~0.0164	0.0086~0.0166
	CNN-LSTM	0.0189~0.0328	0.0162~0.0308	0.0168~0.0273
200	LSTM	0.0056~0.0120	0.0057~0.0110	0.0064~0.0097
	Encoder-Decoder LSTM	0.0085~0.0211	0.0098~0.0178	0.0091~0.0174
	CNN-LSTM	0.0144~0.0327	0.0151~0.033	0.0178~0.0289

Table 13 Prediction Results of Toyota

Length of training period(days)	Model	Features		
		CLOSE	OHLCV	OHLCV+
600	LSTM	0.0117~0.0943	0.0104~0.0242	0.0084~0.0153
	Encoder-Decoder LSTM	0.0095~0.1247	0.0141~0.0467	0.0133~0.0506
	CNN-LSTM	0.0486~0.0608	0.0217~0.0859	0.0190~0.0982
400	LSTM	0.0144~0.1030	0.0114~0.0241	0.0110~0.0192
	Encoder-Decoder LSTM	0.0169~0.1301	0.0119~0.0751	0.0148~0.0525
	CNN-LSTM	0.0404~0.0919	0.0267~0.1818	0.1458~0.0527
200	LSTM	0.0111~0.1155	0.0107~0.0167	0.0078~0.0135
	Encoder-Decoder LSTM	0.0208~0.1817	0.0090~0.0583	0.0133~0.0684
	CNN-LSTM	0.0521~0.1101	0.0373~0.1158	0.0337~0.1658

Table 14 Prediction Results of Sony

Length of training period(days)	Model	Features		
		CLOSE	OHLCV	OHLCV+
600	LSTM	0.0030~0.0090	0.0033~0.0096	0.0028~0.0086
	Encoder-Decoder LSTM	0.0043~0.0148	0.0038~0.0187	0.0034~0.0122
	CNN-LSTM	0.0076~0.0366	0.0075~0.0304	0.0064~0.0294
400	LSTM	0.0027~0.0076	0.0036~0.0091	0.0025~0.0081
	Encoder-Decoder LSTM	0.0042~0.0137	0.0037~0.0148	0.0039~0.0138
	CNN-LSTM	0.0080~0.0481	0.0102~0.0401	0.0073~0.0344
200	LSTM	0.0046~0.0117	0.0040~0.0122	0.0038~0.0098
	Encoder-Decoder LSTM	0.0055~0.0237	0.0070~0.0245	0.0051~0.0184
	CNN-LSTM	0.0059~0.0484	0.0080~0.0320	0.0062~0.0394

3.4 Discussion

These datasets are divided into four training periods and prediction periods. The four periods prediction results of NIKKEI 225 are shown in Tables 8 to 11 in order. The results demonstrate the effect of different length train periods on the performance of different input from the top down. Comparing the rows of “RMSE of closing price”, “RMSE of OHLCV” and “RMSE of OHLCV+” in the four tables, we can see that these additional features are effective at improving the prediction accuracy of these models. At the last three columns of the tables, we can explicitly compare the performance of the improved LSTM, encoder-decoder LSTM and CNN-LSTM models from left to right.

As revealed in the results, the RMSE of the model trained with OHLCV+ is slightly better than that of the model trained with OHLCV or closing price in most cases. This suggests that adding MA10, RSI10, OBV, and direction as features to the threes LSTM models will improve them. Although the encoder-decoder LSTM and CNN-LSTM occasionally perform better than the improved LSTM model, most of the time, the improved LSTM is the best performing one. The prediction results of the fourth prediction period have the best performance in NIKKEI 225 dataset. Table 11 displays that the best performance is 0.0083. Furthermore, the three LSTM models have significantly improved performance while increasing the training period.

Since RMSE is sensitive to outliers, the prediction error will be amplified. Hence, we conducted multiple rounds of experiments to reduce the influence and took the average of five experiment results as the final results. In tables 12 to 14, although the prediction results are displayed simply, the findings also hold true. These results can be extended to the prediction of individual stocks, as the experiment results show that the three LSTM models also have good prediction performance for the three individual stocks. Regarding the prediction of individual stocks, the best prediction results for Itochu, Toyota and Sony are 0.0048, 0.0078 and 0.0025 respectively. These prediction results are all derived from the LSTM model trained with the OHLCV+ dataset.

Lai and Chen used an LSTM model to analyze some Taiwan stocks [34] and the best result is 0.013. Li and Shen apply an attention-based multi-input LSTM to the Chinese stock market [33]. Their model's prediction result is 0.996. To forecast the S&P500 and the Korea Composite Stock Price Index 200, Baek and Kim developed an overfitting prevention LSTM [1]. The model's best performance is 0.6928. The prediction accuracy of the improved LSTM model in this chapter is superior to other studies.

Combining all the results, the LSTM model outperforms the encoder-decoder LSTM and CNN-LSTM models. This may be due to the fact that the encoder-decoder framework is more suitable for machine translation of Seq2Seq, while the output variables of all models in this chapter are one-step prediction, which does not involve the multi-step semantic sequencing problem and does not maximize the role of encoder layer and decoder layer. The basic idea of CNN-LSTM is to use convolutional neural network to do feature extraction of images and LSTM is used to generate the images description, while the model prediction in this chapter can be regarded as the prediction of regression problem. Therefore, compared with encoder-decoder LSTM and CNN-LSTM models, the prediction results of improved LSTM model are stable and perform well when predict NIKKEI 225 and other individual stocks. However, in terms of overall results, even with the same model and dataset, the RMSE will vary to some extent when the model predicts different periods. We believe this is due to the stock market's volatility.

3.5 Conclusion

LSTM, encoder-decoder LSTM and CNN-LSTM models for predicting short-term stock prices were proposed in this chapter. The experiment results lead to the following conclusion.

First, as the length of the training data increased, the RMSE decreased significantly. We can conclude that, while the information density of the dataset used in this chapter is low, it can be compensated by increasing the number of sample data.

Second, as the number of data features increased, the RMSE decreased. It can be concluded that in addition to increase the amount of data, increase the number of features (data quality) will also have an impact on the training results of these models. Although RMSE does not decrease linearly as the number of features increases, it performs well in both Nikkei 225 and individual stocks. The MA10, RSI10, OBV and the direction of stock movement demonstrated that these indicators can effectively improve the performance of these models.

Third, LSTM model performs better than encoder-decoder LSTM model and CNN-LSTM model in most cases. In addition, one of the characteristics of the stock market is volatility, and RMSE will vary greatly with different training periods. The experimental results show that CNN-LSTM model which has a good performance in image processing and encoder-decoder layer which has a good performance in text translation are not suitable for dealing with time series prediction problems.

Chapter 4 Conclusion

This thesis starts with a hybrid algorithm ANN model, where GA is used to train the initial parameters and BP is used to train the neural network. With the development of ANN at that time, when designing a neural network, the number of layers and the number of neurons should not be too many. One or two hidden layers are enough to complete the training for most of the problems. Adding more hidden layers will only increase the training time without improving the prediction performance of the model.

Therefore, we set up numerical experiments to compare the prediction performance of the model after changing the activation function and improving the quality of the data set. The results show that the Logsig/Logsig activation function can effectively improve the prediction accuracy of the model. It has good performance in both Nikkei 225 and individual stocks prediction. Based on the prediction results, we propose an investment strategy for individual stocks trading simulation and get good profit.

With the advancement of neural networks, deep learning has achieved excellent performance in several fields with its outstanding performance. We start from improving an LSTM model and use deep learning approach to perform stock market movement prediction. After improving the LSTM model into a multivariate prediction model, we combine the encoder-decoder layer and the CNN layer to the improved LSTM model respectively. These models are applied to Japanese stock market index prediction and individual stocks prediction.

The results show that in most cases, the model trained using the OHLCV+ dataset outperforms the model trained by the other two datasets. It suggests that adding MA10, RSI10, OBV, and direction as features to the threes LSTM models will improve the prediction accuracy. Furthermore, the three LSTM models have significantly improved performance while increasing the training period.

Despite the encoder-decoder LSTM performs well on text translation problems and the CNN-LSTM model has a good performance on image processing problems, for time series prediction problems, the experimental results demonstrate that in most cases, the improved LSTM model is the best performer.

Regarding future research work, we intend to use attention mechanisms to design deep learning architectures. This is based on the fact that when the human brain processes a large amount of input information, people focus on the important information of interest and ignore the secondary information. Similarly, neural networks built using the attention mechanism select only some critical information inputs for processing, thus improving the efficiency of the model.

In 2017, a paper titled "Attention is all you need" revolutionized the NLP field [57]. Neural network models based on attention mechanisms were initially applied to sequence-to-sequence learning on text data, but now attention mechanisms have been extended to a variety of modern deep learning domains such as machine translation, computer vision, speech recognition and reinforcement learning, etc. Zhou et al. [65] proposed an improved model based on transformer for long sequence time-series forecasting problems. The informer outperforms the transformer both in terms of computational speed and prediction accuracy.

In the next step, we will apply the neural network model with attention mechanism to the stock market forecasting problem. Due to the large number of parameters in the informer architecture, it is difficult to optimize the model by manual tuning.

Accordingly, we use GA to automatically optimize some of the parameters of the model based on the informer model. The experiment is currently in progress.

Bibliography

[1] S. O. Abdulsalam, K. S. Adewole and R. G. Jimoh, “Stock trend prediction using regression analysis–a data mining approach”, *ARPN Journal of Systems and Software*, 1(4): 154-157, 2011.

[2] A. A. Ariyo, A. O. Adewumi and C. K. Ayo, “Stock price prediction using the ARIMA model”, in: *Proc. of UKSim-AMSS 16th International Conference on Computer Modelling and Simulation*, IEEE, 106-112, 2014.

[3] G. Armano, M. Marchesi and A. Murru, "A hybrid genetic-neural architecture for stock indexes forecasting", *Information Sciences,* 170(1): 3-33, 2005.

[4] N. Bacanin, M. Zivkovic, L. Jovanovic, et al., “Training a multilayer perception for modeling stock price index predictions using modified whale optimization algorithm”, in: *Proc. of Computational Vision and Bio-Inspired Computing*, 415-430, 2022.

[5] Y. Baek and H. Y. Kim, “ModAugNet: A new forecasting framework for stock market index value with an overfitting prevention LSTM module and a prediction LSTM module”, *Expert Systems with Applications,* 113: 457-480, 2018.

[6] Y. Bengio, P. Simard and P. Frasconi, “Learning long-term dependencies with gradient descent is difficult”, *IEEE Transactions on Neural Networks*, 5(2): 157-166, 1994.

[7] O. Bustos and A. Pomares-Quimbaya, “Stock market movement forecast: A systematic review”, *Expert Systems with Applications*, 156: 113464, 2020.

[8] K. Cho, B. Van Merriënboer, C. Gulcehre, et al., “Learning phrase representations using RNN encoder-decoder for statistical machine translation”, *arXiv preprint arXiv:1406.1078*, 2014.

[9] N. Cristianini and T. J. Shawe, *“An introduction to support vector machines and other kernel-based learning methods”*, Cambridge university press, 2000.

[10] J. L. Elman, “Finding structure in time”, Cognitive Science, 14(2): 179-211, 1990.

[11] E. Fama, "Efficient Capital Markets: A Review of Theory and Empirical Work", *Journal of Finance,* 25(2): 383–417, doi: 10.2307/2325486, 1970.

[12] B. Freisleben, “Stock market prediction with backpropagation networks”, in: *Proc. of Industrial and Engineering Applications of Artificial Intelligence and Expert Systems: 5th International Conference*, 451-460, 1992.

[13] K. Funahashi and Y. Nakamura, “Approximation of dynamical systems by continuous time recurrent neural networks”, *Neural Networks*, 6(6): 801-806, 1993.

[14] E. Guresen, G. Kayakutlu and T. U. Daim, "Using artificial neural network models in stock market index prediction", *Expert Systems with Applications,* 38, 10389-10397, 2011.

[15] F. A. Gers, J. Schmidhuber and F. Cummins, “Learning to forget: Continual prediction with LSTM”, *Neural Computation,* 12(10): 2451-2471, 2000.

[16] R. Gencay, “Non-Linear prediction of security returns with moving average rules”, J*ournal of Forecasting,* 15(3): 165-174, 1996.

[17] E. Hoseinzade and S. Haratizadeh, “CNN-based stock market prediction using a diverse set of variables”, *Expert Systems with Applications*, 129: 273-285, 2019.

[18] G. E. Hinton and R. R. Salakhutdinov, “Reducing the dimensionality of data with neural networks”, *Science*, 313(5786): 504-507, 2006.

[19] J. C. Horne and G. G. Parker, “The random-walk theory: An empirical test”, *Financial Analysts Journal,* 23(6): 87-92, 1967.

[20] K. Hornik, M. Stinchcombe and H. White, "Multilayer feedforward networks are universal approximators", *Neural Networks,* 2(5): 359-366, 1989.

[21] M. Hsu, S. Lessmann, M. Sung, et al., “Bridging the divide in financial market forecasting: machine learners vs. financial economists”, *Expert Systems with Applications*, 61: 215-234, 2016.

[22] S. Hochreiter and J. Schmidhuber, “Long short-term memory”, *Neural Computation,* 9(8): 1735-1780, 1997.

[23] K. Jadav and M. Panchal, "Optimizing weights of artificial neural networks using genetic algorithms", *International Journal of Advanced Research in Computer Science and Electronics Engineering,* 1(10): 47-51, 2012.

[24] N. Jing, Z. Wu and H. Wang, “A hybrid model integrating deep learning with investor sentiment analysis for stock price prediction”, *Expert Systems with Applications*, 178: 115019, 2021.

[25] T. Jo, "VTG schemes for using back propagation for multivariate time series prediction”, *Applied Soft Computing*, 13(5): 2692-2702, 2013.

[26] H. Kim and C. Won, “Forecasting the volatility of stock price index: A hybrid model integrating LSTM with multiple GARCH-type models”, *Expert Systems with Applications*, 103: 25-37, 2018.

[27] K. Kim and I. Han, "Genetic algorithms approach to feature discretization in artificial neural networks for the prediction of stock price index", *Expert Systems with Applications*, 19 (2): 125-132, 2000.

[28] T. Kimoto, K. Asakawa, M. Yoda, et al., "Stock market prediction system with modular neural networks", in *Proc. of International Joint Conference on Neural Networks*, IEEE, 1-6, 1990.

[29] A. Krizhevsky, I. Sutskever and G. E. Hinton, "Imagenet classification with deep convolutional neural networks", *Communications of the ACM*, 60(6): 84-90, 2017.

[30] R. Kuo, C. Chen and Y. Hwang, "An intelligent stock trading decision support system through integration of genetic algorithm based fuzzy neural network and artificial neural network", *Fuzzy Sets and Systems*, 118(1): 21-45, 2001.

[31] Y. Kara, M. A. Boyacioglu and Ö. K. Baykan, "Predicting Direction of Stock Price Index Movement Using Artificial Neural Networks and Support Vector Machines: The Sample of The Istanbul Stock Exchange", *Expert Systems with Applications,* 38: 5311-5319, 2011.

[32] A. Lapedes and R. Farber, "Nonlinear signal processing using neural networks: Prediction and system modelling", No. LA-UR-87-2662; CONF-8706130-4, *Office of Scientific and Technical Information Technical Reports*, 1987.

[33] C. Lai, R. Chen and R. E. Caraka, "Prediction stock price based on different index factors using LSTM", in: *Proc. of* International *Conference on Machine Learning and Cybernetics,* 1-6, 2019.

[34] H. Li, Y. Shen and Y. Zhu, "Stock price prediction using attention-based multi-input LSTM", in: *Proc. of Asian Conference on Machine Learning,* PMLR, 454-469, 2018.

[35] T. Lee and C. Chiu, "Neural network forecasting of an opening cash price index", *International Journal of Systems Science,* 33: 229-237, 2002.

[36] A. Mahmoodzadeh, H.R. Nejati, M. Mohammadi, et al., "Forecasting tunnel boring machine penetration rate using LSTM deep neural network optimized by grey wolf optimization algorithm", *Expert Systems with Applications*, 209, 118303, 2022.

[37] D. J. Montana and L. Davis, "Training feedforward neural networks using genetic algorithms", *International Joint Conference on Artificial Intelligence,* 89: 762-767, 1989.

[38] G. W. Milligan and M. C. Cooper, "A study of standardization of variables in cluster analysis", *Journal of Classification,* 5(2): 181-204, 1988.

[39] S. Mehtab, J. Sen and A. Dutta, "Stock price prediction using machine learning and LSTM-based deep learning models", in: *Proc. of Symposium on Machine Learning and Metaheuristics Algorithms, and Applications*, 2: 88-106, 2020.

[40] S. Mukherjee, E. Osuna and F. Girosi, "Nonlinear prediction of chaotic time series using support vector machines", in: *Proc. of IEEE Signal Processing Society Workshop,* 1997.
[41] B.B. Nair, S.G. Sai, A. Naveen, et al., "A GA-artificial neural network hybrid system for financial time series forecasting", *Information Technology and Mobile Communication Springer,* 499-506, 2011.
[42] M. F. Osborne, "Brownian Motion in the Stock Market", *Operations Research,* 7: 145-173, 1959.
[43] P. Pai and C. Lin, "A hybrid ARIMA and support vector machines model in stock price forecasting", *Omega*, 33(6): 497-505, 2005.
[44] J. Patel, S. Shah, P. Thakkar, et al., "Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques", *Expert systems with applications*, 42(1): 259-268, 2015.
[45] M. Qiu, "A study on prediction of stock market index and portfolio selection", doctor thesis, Fukuoka Institute of Technology, 2014.
[46] M. Qiu, Y. Song and F. Akagi, "Application of artificial neural network for the prediction of stock market returns: The case of the Japanese stock market", *Chaos, Solitons & Fractals,* 85: 1-7, 2016.
[47] M. Qiu and Y. Song, "Predicting the direction of stock market index movement using an optimized artificial neural network model", *PloS one* 11(5): e0155133, 2016.
[48] M. A. Rubi, S. Chowdhury, A. A. Rahman, et al., "Fitting Multi-Layer Feed Forward Neural Network and Autoregressive Integrated Moving Average for Dhaka Stock Exchange Price Predicting", *Emerging Science Journal*, 6(5): 1046-1061, 2022.
[49] S. Ruder, "An overview of gradient descent optimization algorithms", arXiv preprint arXiv: 1609.04747, 2016.
[50] E. Schneburg, "Stock Price Prediction Using Neural Networks: A Project Report", *Neurocomputing,* 2(1): 17-27, 1990.
[51] N. T. Sainath, O. Vinyals, A. Senior, et al., "Convolutional, long short-term memory, fully connected deep neural networks", in *Proc. of International Conference on Acoustics, Speech and Signal Processing,* IEEE, 2015.
[52] M. Shiab, "The predictability of the Amman stock exchange using the univariate autoregressive integrated moving average (ARIMA) model", *Journal of Economic and Administrative Sciences*, 22(2): 17-35, 2006.

[53] V. H. Shah, "Machine learning techniques for stock prediction", *Foundations of Machine Learning,* 1(1): 6-12, 2007.

[54] E. Turban and R. Trippi, "Neural networks finance and investment: using artificial intelligence to improve real-world performance", 2nd ed., McGraw-Hill, Inc., 1995.

[55] J. Taylor, "Volatility forecasting with smooth transition exponential smoothing", *International Journal of Forecasting*, 20(2): 273-286, 2004.

[56] P. Tüfekci, "Predicting the Direction of Movement for Stock Price Index Using Machine Learning Methods", in: *Proc. of the Second International Afro-European Conference for Industrial Advancement*, 477-492, 2016.

[57] A. Vaswani, N. Shazeer, N. Parmar, et al., "Attention is all you need", *Advances in neural information processing systems*, 2017.

[58] C. S. Vui, G. K. Soon, C. K. On, et al., "A review of stock market prediction with Artificial neural network (ANN)", in: *Proc. of International Conference on Control System, Computing and Engineering*, IEEE, 477-482, 2013.

[59] H. Wang, Y. Jiang and H. Wang, "Stock return prediction based on Bagging-decision tree", in: *Proc. of International Conference on Grey Systems and Intelligent Services*, IEEE, 1575-1580, 2009.

[60] J. Wang and J. Leu, "Stock market trend prediction using ARIMA-based neural networks", in: *Proc. of International Conference on Neural Networks*, 4: 2160-2165, 1996.

[61] J. Wang, J. Wang, Z. Zhang, et al., "Stock index forecasting based on a hybrid model", *Omega*, 40(6): 758-766, 2012.

[62] Y. Yoon and G. Swales, "Predicting stock price performance: A neural network approach", in: *Proc. of the Twenty-Fourth Annual Hawaii International Conference on System Sciences*, IEEE, 4: 156-162, 1991.

[63] K. Zhang, G. Zhong, J. Dong, et al., "Stock market prediction based on generative adversarial network", *Procedia Computer Science*, 147: 400-406, 2019.

[64] Y. Zhang and L. Wu, "Stock market prediction of S&P 500 via combination of improved BCO approach and BP neural network", *Expert Systems with Applications*, 36(5): 8849-8854, 2009.

[65] H. Zhou, S. Zhang, J. Peng, et al., "Informer: Beyond efficient transformer for long sequence time-series forecasting", in: *Proc. of AAAI Conference on Artificial Intelligence,* 35(12): 11106-11115, 2021.

Appendix: The Detailed Results of Individual Stocks

Table 15 Results of first Prediction period (Sony)

	Length of training period(days)	The improved LSTM	Encoder-Decoder LSTM	CNN-LSTM
RMSE of closing price	600	0.003032	0.0043	0.007612
	400	0.002732	0.004208	0.007958
	200	0.004624	0.005528	0.005946
RMSE of OHLCV	600	0.003306	0.003812	0.007498
	400	0.00355	0.003656	0.010166
	200	0.003838	0.00699	0.008
RMSE of OHLCV+	600	0.00284	0.003382	0.006366
	400	0.002514	0.003926	0.007274
	200	0.00401	0.00508	0.00616

Table 16 Results of second Prediction period (Sony)

	Length of training period(days)	The improved LSTM	Encoder-Decoder LSTM	CNN-LSTM
RMSE of closing price	600	0.006746	0.014834	0.036624
	400	0.007262	0.011854	0.048062
	200	0.00726	0.023742	0.04593
RMSE of OHLCV	600	0.009612	0.018718	0.030422
	400	0.008118	0.014844	0.040066
	200	0.009818	0.011658	0.032006
RMSE of OHLCV+	600	0.008642	0.012196	0.029388
	400	0.00782	0.010492	0.034386
	200	0.007372	0.01842	0.03418

Table 17 Results of third Prediction period (Sony)

	Length of training period(days)	The improved LSTM	Encoder-Decoder LSTM	CNN-LSTM
RMSE of closing price	600	0.0064	0.012726	0.013764
	400	0.007568	0.013676	0.011922
	200	0.007984	0.011638	0.013388
RMSE of OHLCV	600	0.006914	0.00873	0.01749
	400	0.007326	0.008826	0.011892
	200	0.007638	0.008178	0.011014
RMSE of OHLCV+	600	0.00711	0.008726	0.014836
	400	0.006628	0.009176	0.012082
	200	0.007566	0.008326	0.012722

Table 18 Results of fourth Prediction period (Sony)

	Length of training period(days)	The improved LSTM	Encoder-Decoder LSTM	CNN-LSTM
RMSE of closing price	600	0.007002	0.012114	0.025608
	400	0.006792	0.011102	0.031266
	200	0.011746	0.017586	0.048424
RMSE of OHLCV	600	0.007266	0.008338	0.016986
	400	0.00913	0.011974	0.024946
	200	0.009654	0.02445	0.023162
RMSE of OHLCV+	600	0.007624	0.01095	0.018168
	400	0.008124	0.01375	0.021884
	200	0.012176	0.014454	0.039444

Table 19 Results of first Prediction period (Toyota)

	Length of training period(days)	The improved LSTM	Encoder-Decoder LSTM	CNN-LSTM
RMSE of closing price	600	0.011724	0.00947	0.05006
	400	0.014354	0.01693	0.0539
	200	0.01105	0.020804	0.05209
RMSE of OHLCV	600	0.010422	0.017264	0.03146
	400	0.014524	0.019572	0.03627
	200	0.01197	0.014282	0.03733
RMSE of OHLCV+	600	0.015264	0.022566	0.03026
	400	0.011192	0.017934	0.03275
	200	0.00781	0.014652	0.03922

Table 20 Results of second Prediction period (Toyota)

	Length of training period(days)	The improved LSTM	Encoder-Decoder LSTM	CNN-LSTM
RMSE of closing price	600	0.094346	0.124654	0.04859
	400	0.10302	0.130052	0.09193
	200	0.115456	0.181652	0.11015
RMSE of OHLCV	600	0.024224	0.046734	0.08586
	400	0.024126	0.0751	0.18179
	200	0.016708	0.058334	0.11584
RMSE of OHLCV+	600	0.0137	0.050568	0.09818
	400	0.019196	0.052528	0.1458
	200	0.01186	0.068374	0.16582

Table 21 Results of third Prediction period (Toyota)

	Length of training period(days)	The improved LSTM	Encoder-Decoder LSTM	CNN-LSTM
RMSE of closing price	600	0.077182	0.066884	0.06083
	400	0.067108	0.050672	0.07218
	200	0.081602	0.048242	0.08815
RMSE of OHLCV	600	0.012388	0.015742	0.02856
	400	0.01522	0.022046	0.05103
	200	0.01614	0.01792	0.07812
RMSE of OHLCV+	600	0.012258	0.019186	0.0297
	400	0.013838	0.025984	0.05269
	200	0.013456	0.025766	0.11318

Table 22 Results of fourth Prediction period (Toyota)

	Length of training period(days)	The improved LSTM	Encoder-Decoder LSTM	CNN-LSTM
RMSE of closing price	600	0.04077	0.051462	0.05009
	400	0.044066	0.041234	0.04044
	200	0.049518	0.053988	0.05511
RMSE of OHLCV	600	0.010808	0.014136	0.02167
	400	0.011388	0.011876	0.02668
	200	0.010658	0.009042	0.03899
RMSE of OHLCV+	600	0.008392	0.013308	0.01904
	400	0.011008	0.014844	0.02123
	200	0.01247	0.01327	0.03367

Table 23 Results of first Prediction period (Itochu)

	Length of training period(days)	The improved LSTM	Encoder-Decoder LSTM	CNN-LSTM
RMSE of closing price	600	0.005562	0.009144	0.02632
	400	0.006038	0.01009	0.02558
	200	0.008204	0.009366	0.02051
RMSE of OHLCV	600	0.007006	0.013868	0.02603
	400	0.00789	0.014738	0.02518
	200	0.009926	0.014016	0.02313
RMSE of OHLCV+	600	0.00482	0.009068	0.02395
	400	0.006358	0.01005	0.02261
	200	0.007646	0.011202	0.01978

Table 24 Results of second Prediction period (Itochu)

	Length of training period(days)	The improved LSTM	Encoder-Decoder LSTM	CNN-LSTM
RMSE of closing price	600	0.00544	0.007868	0.02133
	400	0.005658	0.006658	0.01919
	200	0.005596	0.008454	0.01439
RMSE of OHLCV	600	0.00601	0.008864	0.02051
	400	0.004984	0.007126	0.01616
	200	0.005686	0.009762	0.01512
RMSE of OHLCV+	600	0.005918	0.010168	0.01899
	400	0.006174	0.009284	0.01952
	200	0.00643	0.010436	0.01778

Table 25 Results of third Prediction period (Itochu)

	Length of training period(days)	The improved LSTM	Encoder-Decoder LSTM	CNN-LSTM
RMSE of closing price	600	0.00841	0.016436	0.01902
	400	0.00801	0.015678	0.01885
	200	0.01202	0.019926	0.02976
RMSE of OHLCV	600	0.008194	0.016766	0.01628
	400	0.009658	0.016424	0.0194
	200	0.010384	0.017284	0.02749
RMSE of OHLCV+	600	0.006938	0.016396	0.01526
	400	0.007218	0.016582	0.01676
	200	0.009686	0.01743	0.02492

Table 26 Results of fourth Prediction period (Itochu)

	Length of training period(days)	The improved LSTM	Encoder-Decoder LSTM	CNN-LSTM
RMSE of closing price	600	0.008626	0.011766	0.0334
	400	0.008876	0.012646	0.03284
	200	0.008048	0.021148	0.03265
RMSE of OHLCV	600	0.007872	0.010766	0.03172
	400	0.006384	0.01119	0.03077
	200	0.011002	0.017786	0.03302
RMSE of OHLCV+	600	0.00691	0.00898	0.02591
	400	0.00651	0.008642	0.02726
	200	0.00813	0.009076	0.02893

List of Publications

査読付き学術論文

1. C. Li and Y. Song, “Applying LSTM Model to Predict the Japanese Stock Market with Multivariate Data”, Journal of Computers, (accepted).

国際会議

2. M. Qiu, C. Li and Y. Song, “Forecasting Direction of Individual Stock Price based on a Hybrid Model”, in: *Proc.of International Conference on Computer and Information Sciences* (ICICIS), 2016.
3. M. Qiu, C. Li and Y. Song, “Application of the Artificial Neural Network in predicting the direction of stock market index”, 10th International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS), IEEE, 219-223, 2016.
4. C. Li and Y Song, “Application of Artificial Neural Network in Predicting the Direction of Stock Price”, in: *Proc. of the 3th International Conference on Production Management* (ICPM), 2017.
5. C. Li and Y. Song, “An Investment Strategy for Individual Stock Based on Prediction Using ANN”, in: *Proc. of the 11th Triennial International Conference of the Association of Asia Pacific Operational Research Societies* (APORS), 2018.
6. C. Li and Y. Song, “Forecasting Individual Stock Price Movements Using LSTM Neural Network”, in: *Proc. of Asian Conference of Management Science & Applications* (ACMSA), 2019.
7. J. Pittayarugsarit, Y. Song and C. Li, “A Problem of Assigning Students to Nursery Schools”, in: *Proc. of Asian Conference of Management Science & Applications* (ACMSA) 2019.
8. C. Li and Y. Song, “A comparison of single-feature and multi-feature analysis for stock market based on LSTM model”, in: *Proc. of the 13th Triennial International Conference of the Association of Asia Pacific Operational Research Societies* (APORS), 2022.
9. X. Liu, H. Liu, C. Li and Y. Song, “A Study on Daily Shift Scheduling in Multi-task Call Centers”, in: *Proc. of the 13th Triennial International Conference of the Association of Asia Pacific Operational Research Societies* (APORS), 2022.
10. H. Liu, X. Liu, C. Li and Y. Song, “An Implicit Modeling for Multi-task Call Center Shift Scheduling”, in: *Proc. of the 13th Triennial International Conference of the Association of Asia Pacific Operational Research Societies* (APORS), 2022.

国内会議

11. C. Li, Y. Song, “Generating Candlesticks as Images for Financial Forecasting Using Convolutional Neural Networks”, 日本生産管理学会 第 51 回全国大会予稿集, 80-81, 2020.
12. C. Li, Y. Song, “Predicting short term stock prices based on different features using LSTM model”, 日本生産管理学会 第 56 回全国大会予稿集, 74-75, 2022.