Proceedings of CONF-CIAP 2026 Symposium: Applied Mathematics and Statistics
DOI: 10.54254/2753-8818/2026.CH31629

# Stock Price Analysis and Prediction Using Fully Connected Neural Networks

**Minxuan Li**

*Zhonghe High School, Chengdu, China*
*754326744@qq.com*

***Abstract.*** As the economy grows non-stop, stocks have become a part of people's daily lives. So, guessing stock prices correctly is a very important topic. If we can guess stock prices right, it can help people who invest money get more money back. It also helps the country's economy grow. To solve this problem, many people who study this stuff have used "neural networks" (a kind of computer tool) and got good results. In this study, we used a "fully connected neural network" as the tool to guess stock prices. The data we used is Coca-Cola's daily stock prices from October 23, 2012 to October 23, 2022. We got this data from Kaggle (a website with data). We compared many test results using two ways: MAE and MSE (two simple ways to check if the guess is good). Then we found the right settings for the tool. The study says: this tool is good for guessing stock prices when the data has some time-related links, some useful information, and is not a simple straight-line relationship.

***Keywords:*** Fully Connected Neural Network, Stock Price Prediction, Mid-Term Investment

## 1. Introduction

As the economy grows, guessing stock prices has become a more and more important topic. Market changes, the country's rules, and big social events all affect stock prices. This makes stock prices change a lot and not follow a simple pattern, so guessing them correctly is very hard. For many years, many researchers have studied this topic deeply and got good results.

From a real-life view, stocks are closely linked to many companies and people, and the stock market is very large. Numbers show that there are about 220 million stock investors in China. Making stock price guesses more correct can help investors get more money back and also help the country's economy grow. So, studying stock price prediction is really useful in real life.

This paper has six parts. Part 1 talks about the background of stock price prediction and why it is useful in real life. Part 2 looks at some results that earlier researchers got. Part 3 explains the basic ideas of neural networks. Part 4 shows the specific tests that were done and how to pick the right settings for the model. Part 5 analyzes the test results. Finally, Part 6 sums up the conclusions of this study.

## 2. Literature review

When dealing with guessing stock prices, some scholars have got some middle results.

DOI: 10.54254/2753-8818/2026.CH31629

In 2019, Qiao built different kinds of neural network models (a type of computer tool) based on deep learning methods. These models include Multilayer Perceptron (MLP), Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), Long Short-Term Memory Network (LSTM), and Gated Recurrent Unit (GRU). He used data from the Shanghai Stock Exchange Composite Index, which covers from July 1, 1991 to April 1, 2019. Qiao also added an Attention Mechanism (a small tool to make the model work better) to make the recurrent neural network models work even better. Comparing different tests showed that the Attention Mechanism not only made the guessing more accurate but also stopped the model from "overfitting" (meaning the model doesn't work well on new data). It got very good results. So, adding the Attention Mechanism to neural network models for guessing stocks can make the guesses more correct [1]. In 2020, Zhang and his team took out useful information from news and used a recurrent neural network to guess stock prices. Their tests found that the Price-News-CNN model had an average correct rate of 55.84%. Some single stocks even had a correct rate of 60% or higher. This shows the model is very accurate and this method works for guessing stock prices [2]. In 2022, Yan put forward a generative adversarial neural network model called SAR-GAN. This model combines a self-attention mechanism and residual networks (two small tools). The "generator" part of this model has LSTM layers, self-attention layers, and residual layers (different parts of the tool). Yan used the Shanghai Stock Exchange Composite Index as an example to analyze the index data. He also chose stock prices from big companies in important industries of different markets. The study found that compared with models trained with stock data from only one market, the SAR-GAN model had smaller guessing errors and worked better on different markets. So it is suitable for guessing stock prices in different markets [3]. In 2021, Zhang used a BP neural network and an ARMA-GARCH model to guess the closing prices of SAIC Motor Corporation. The results showed that when guessing the closing prices of the next 20 days, the BP neural network reduced the "mean absolute error" (a way to check how wrong the guess is) by 31.4%. When guessing the next 6 days, the ARMA-GARCH model reduced the mean absolute error by 7.4%. This means the BP neural network is better for guessing stock prices in the long term, while the ARMA-GARCH model is better for short-term guesses. Investors can choose different guessing methods according to their needs to get the most money back [4]. In 2023, Huang and his team improved the BP neural network and proposed the PSO-BP model. They compared it with an unoptimized LGB model and a BOA-LGB model. The PSO-BP model got an RMSE of 0.026615 and an MAE of 0.0371 (both are ways to check guess accuracy), which is the best. This shows the PSO-BP neural network model is very stable and suitable for guessing stock prices [5]. In the same year, Yuan and his team built a neural network model using MATLAB (a software). They used data from Zhongxing Telecommunications (stock code 00063) from March 2012 to December 2020 as the "training set" (data to teach the model) and data from March to September 2021 as the "test set" (data to check the model). The tests calculated the correct rate (T) and mean squared error (MSE, another way to check wrong guesses). When the correct rate was the highest, the MSE was 0.1632. By comparing the BP neural network before and after using Principal Component Analysis (PCA, a tool to simplify data), the study concluded that the PCA-BP model was more accurate and could help investors guess stock prices effectively [6].

In the years after that, some scholars did research using LSTM models (a type of computer tool for guessing stocks). In 2023, Wang made an MDB-BiLSTM model. He used LSTM, bidirectional LSTM (called BiLSTM for short), multi-delay embedding transformation (a small technique), and combined them with tensor methods (another small technique) to guess six kinds of stock prices. The results showed that this model was good at guessing stock trends and could work well on different data. When compared with BiLSTM, LSTM, CNN-LSTM models and different sets of

DOI: 10.54254/2753-8818/2026.CH31629

data, this model got an MAE of 0.0744. On average, this number was 0.0015 lower than other models. This means the model made fewer mistakes. It gives investors a reliable tool to guess stock prices [7]. In 2024, Zhan used the LSTM neural network model to guess stock prices. He used historical data of Tata Global Beverages from July 21, 2010 to September 28, 2018. He compared RNN and LSTM models. Finally, he proved that LSTM works for guessing stock prices [8]. In 2024, Yuan and his team put forward a mixed model for guessing stock prices. This model is based on deep learning and uses many kinds of data and investors' feelings. They took investors' comments from stock forums, used TextCNN (a tool) to analyze whether the comments are good or bad, and made a "sentiment index" (a number showing investors' feelings). Then they put this index, technical indicators (data about stock trading), and historical trading data into the BiLSTM model to guess stocks. After adding the Attention Mechanism (a small tool to make the model better), the MAE dropped by about 0.091 and the RMSE dropped by about 0.0101. The study said this mixed model is better than single models and models that don't use investors' feelings. It helps investors make decisions about buying or selling stocks [9]. In 2025, Zhu made a combined model called RVMD-CNN-BiLSTM-AM. He used minimum envelope entropy as a "fitness function" (a standard to check if the model is good). The Frost Ice Optimization algorithm (a tool) optimized the variational mode decomposition (a way to process data), and found the best number of intrinsic mode functions and penalty factors (settings for the model). Then he put the processed stock data into the CNN-BiLSTM model to get deeper data features (useful information from data), and passed these features into the Attention Mechanism to catch time-related characteristics (how data changes over time). The results showed that the model reduced MAE by 10.56% and MAPE by 18.05%. This means the model is more accurate than other comparison models. It gives useful references to financial workers for guessing stock prices and market trends [10].

The above is a summary of some existing research results. Because scholars keep studying, the accuracy of guessing stock prices has been getting better and better.

## 3. Methods

Neurons are fundamental components of neural networks, and their activation functions are typically chosen as the sigmoid function or the tanh function. Assuming the output vector of a neuron is $\vec{x}$, the weight vector is $\vec{w}$, and the activation function is the sigmoid function, the neuron's output y can be expressed as:

$$y = sigmoid(\vec{x} \bullet \vec{w}^T) \tag{1}$$

The sigmoid function is defined as:

$$\mathrm{sigmoid}(\mathrm{x}) = \frac{1}{1+\mathrm{e}^{-\mathrm{x}}} \tag{2}$$

Substituting this into the previous equation gives:

$$\mathrm{y} = \frac{1}{1+\mathrm{e}^{-\vec{\mathrm{w}}^{\mathrm{T}} \bullet \vec{\mathrm{x}}}} \tag{3}$$

A neural network is composed of multiple neurons connected according to certain rules. The leftmost layer is the input layer, responsible for receiving input data. The middle layers are hidden layers, and the rightmost layer is the output layer, where the network's output is obtained. Neurons

DOI: 10.54254/2753-8818/2026.CH31629

within the same layer are not connected. Each neuron in the n-th layer is connected to all neurons in the n−1-th layer, and each connection has an associated weight. These rules define the structure of a fully connected network.

When calculating outputs, the network functions as a mapping from an input vector $\vec{x}$ to an output vector $\vec{y}$, expressed as:

$$\vec{y} = f_{network}\left(\vec{x}\right) \tag{4}$$

To compute the output of a neural network for a given input, each element $x_i$ of the input vector xxx is first assigned to the corresponding neuron in the input layer. Then, according to Equation (1), the value of each neuron is computed layer by layer, moving forward through the network until all neurons in the output layer are evaluated. Finally, the output vector $\vec{y}$ is obtained by combining the values of all neurons in the output layer.

Assume each training sample is represented as ( $\vec{x}$ , $\vec{t}$ ), where $\vec{x}$ is the feature vector of the sample and $\vec{t}$ is the target value of the sample.

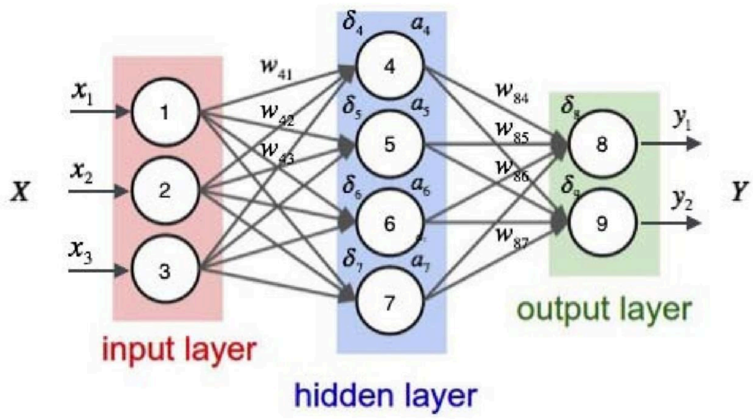


Figure 1. Architecture of a fully connected neural network

First, using the sample features $\vec{x}$ , the outputs $a_i$ of all hidden layer neurons and $y_i$ of all output layer neurons are computed. Then, the error term $\delta_i$ for each neuron is calculated as follows:

For output layer neuron i:

$$\delta_i = y_i(1 - y_i)(t_i - y_i) \tag{5}$$

where $\delta_i$ is the neuron's error term, $y_i$ is its output value, and $t_i$ is the target value corresponding to neuron i.

For hidden layer neuron:

$$\delta_i = a_i(1 - a_i) \sum_{k \in outputs} w_{ki} \delta_k \tag{6}$$

where $a_i$ is the output of neuron i, $w_{ki}$ is the weight connecting neuron i to neuron k in the next layer, and $\delta_k$ is the error term of neuron k.

Finally, the weight on each connection is updated as follows:

DOI: 10.54254/2753-8818/2026.CH31629

$$w_{ji} \leftarrow w_{ji} + \eta \delta_j x_{ji} \quad (7)$$

where $w_{ji}$ is the weight from neuron i to neuron j, $\eta$ is a constant called the learning rate, and $x_{ji}$ is the input from neuron i to neuron j.

The input value of the bias term is always set to 1.

This process describes the computation of each neuron's error term and the update of connection weights. To calculate a neuron's error term, the error terms of all neurons in the next layer to which it is connected must first be computed. Therefore, the error terms must be calculated in a backward sequence, starting from the output layer and moving layer by layer toward the hidden layers connected to the input layer. Once all neuron error terms are calculated, all weights can be updated according to their respective relationships.

## 4. Experimental results

In this study, the dataset consisted of Coca-Cola's daily stock prices from October 23, 2012, to October 23, 2022, obtained from Kaggle. The independent variables x included the opening price, highest price, lowest price, and closing price of the previous day, while the dependent variable y was the closing price of the following day. For the baseline control experiment, a sample spanning three years from October 22, 2019, to October 22, 2022, was selected. The period from June 13, 2022, to October 22, 2022 (130 days) was designated as the test set, with the remainder used as the training set. The neural network structure included a single hidden layer with 16 neurons. The results of the baseline experiment were: MAE = 0.029394, MSE = 0.00148. The predicted curve is shown in Figure 2.

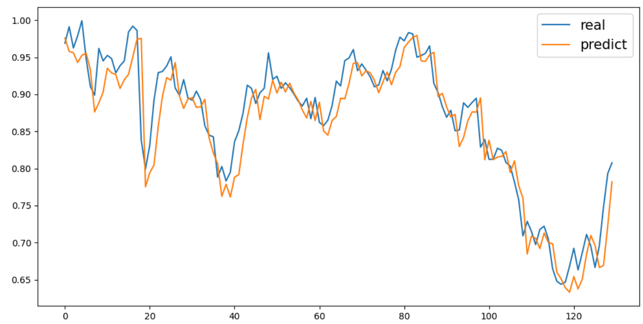


Figure 2. Predicted curve of baseline control group

Using a controlled-variable approach, the first experiment fixed all other parameters and the test set while varying the dataset time span to 2 years, 5 years, and 10 years. The results are summarized in Table 1:

Table 1. MAE and MSE for different time spans

| | 2 years | 3 years (baseline) | 5 years | 10 years |
|---|---|---|---|---|
| MAE | 0.046566 | 0.029394 | 0.075587 | 0.033027 |
| MSE | 0.003387 | 0.001480 | 0.007423 | 0.001584 |

DOI: 10.54254/2753-8818/2026.CH31629

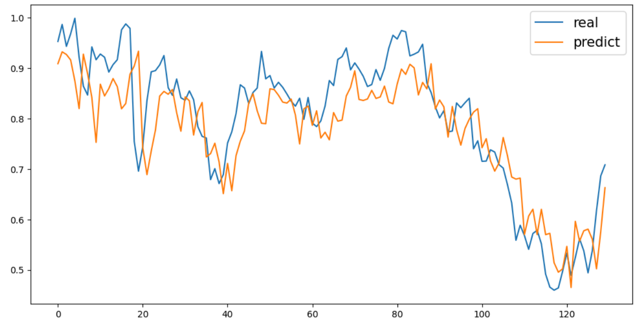


Figure 3. Predicted curve for 2-year dataset

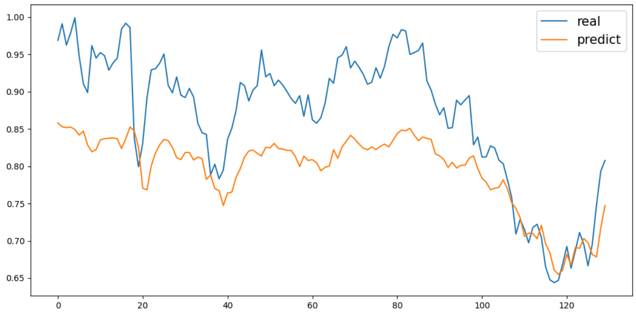


Figure 4. Predicted curve for 5-year dataset

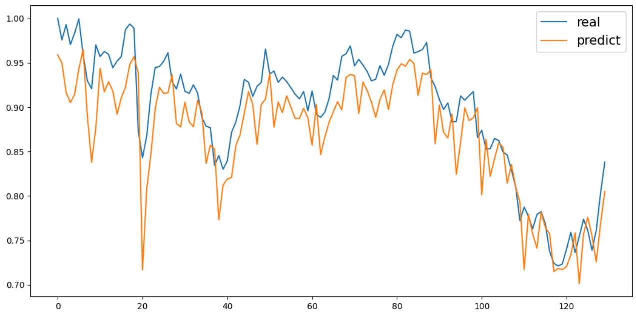


Figure 5. Predicted curve for 10-year dataset

Comparison indicates that a three-year time span yields more accurate predictions.

In the second experiment, all other parameters and the test set were kept constant while varying the number of hidden layer neurons to 8 and 32. The results are shown in Table 2:

Table 2. MAE and MSE for different numbers of neurons

| | 8 | 16 (baseline) | 32 |
|---|---|---|---|
| MAE | 0.057901 | 0.029394 | 0.046098 |
| MSE | 0.004408 | 0.001480 | 0.002901 |

DOI: 10.54254/2753-8818/2026.CH31629

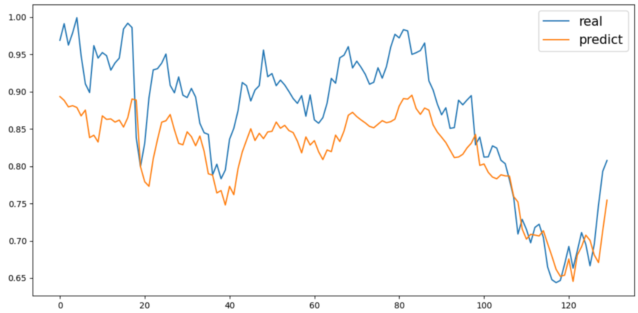


Figure 6. Predicted curve for 8 neurons

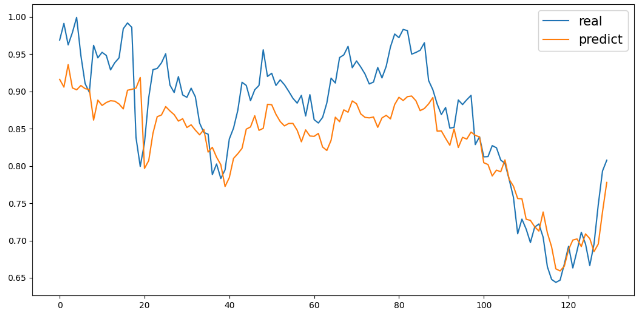


Figure 7. Predicted curve for 32 neurons

Comparison indicates that 16 neurons in the hidden layer produce the most accurate predictions.

In the third experiment, all other parameters and the test set were fixed while varying the number of hidden layers to two and three. The results are presented in Table 3:

Table 3. MAE and MSE for different numbers of hidden layers

| | 1 layer (baseline) | 2 layers | 3 layers |
|---|---|---|---|
| MAE | 0.029394 | 0.023940 | 0.049616 |
| MSE | 0.001480 | 0.001064 | 0.049616 |

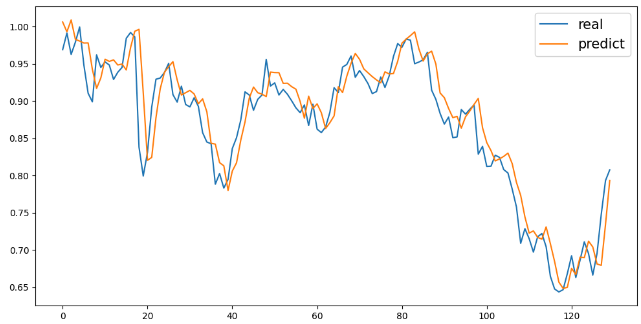


Figure 8. Predicted curve for two hidden layers

DOI: 10.54254/2753-8818/2026.CH31629

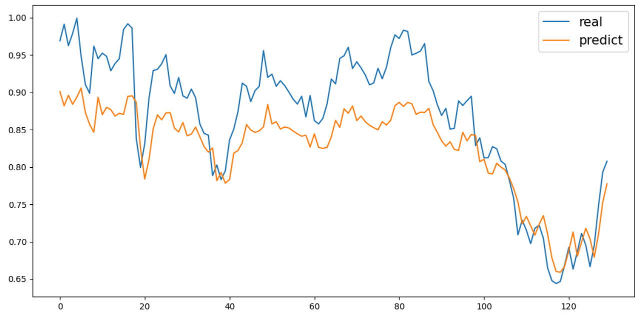


Figure 9. Predicted curve for three hidden layers

When we compare these two lines, we see that using two hidden layers gives the most correct guesses.

To sum up, the best model for guessing stock prices uses three years of stock data, two hidden layers, and 16 neurons (small parts inside each hidden layer) in every hidden layer.

## 5. Analysis

Looking at the time length we use for data, how correct our stock guesses are shows how much the target stock depends on time and how sensitive it is to time changes. The tests show that using data from three years gives the most correct guesses. This tells us that this stock is good for mid-term investment (holding the stock for a few years).

Looking at the number of neurons (tiny parts inside the model's hidden layers), the right number mainly depends on how much useful information is in the data and how much detail we need to get from the data. If there are too few neurons, we can't use all the useful information. If there are too many neurons, the information gets scattered, and the model will "overfit" (it works well on old data but badly on new data). The tests show that setting 16 neurons in each hidden layer is the best choice. This means the useful information in this stock's data is neither too much nor too little—it's moderate.

Looking at the number of hidden layers (parts inside the model that process data), this shows how "nonlinear" the stock problem is (nonlinear means the stock price doesn't follow a simple straight-line change). If the nonlinearity is higher, a model with more hidden layers usually works better. In our tests, a model with two hidden layers worked best. This means the nonlinearity of this stock prediction problem is moderate.

Summary: Coca-Cola stock has moderate dependence on time, moderate useful information in its data, and moderate nonlinearity. So, for investors, a mid-term investment strategy is good for this kind of stock. For building a model to guess this kind of stock's price, we suggest using three years of data, two hidden layers, and 16 neurons in each layer.

## 6. Conclusion

In the money and investment field, venture investment has always been a popular topic to study. As artificial intelligence and machine learning technologies have become more popular, research in this field has made big progress. Many researchers have come up with different ways and practical solutions to guess and analyze the trends of risky investment products. This study first looked at what earlier researchers had done. Then it used Coca-Cola's stock prices from October 22, 2012 to

DOI: 10.54254/2753-8818/2026.CH31629

October 22, 2022 as the data. The study used the controlled-variable method (a way to test things by changing only one part at a time). The tests changed the time length of data, the number of hidden layers, and the number of neurons in each layer in a planned way. The results show that this stock has moderate dependence on time, moderate useful information in its data, and moderate nonlinearity (prices don't follow a simple straight line). For this kind of stock, we suggest using a model with three years of data, two hidden layers, and sixteen neurons in each layer.

This study also has some weaknesses. The size of the tests was small. So it can't fully show the bigger market situation or all kinds of investment plans.

To fix these weaknesses, here are some suggestions for future research: first, make the research bigger and study mixed investment plans together. This can give investors more possible ways to invest. Second, do different studies based on different market situations. This can help create investment methods and prediction models that work better in more cases.

## References

[1] Qiao, R. (2019). Stock prediction model based on neural networks. Operations Research and Management, 28(10), 9.
[2] Zhang, Z., Huang, L., Chen, C., & Yan, H. (2020). Stock prediction method based on news feature extraction and recurrent neural networks. Journal of Documentation and Data Studies, (1).
[3] Yan, D., & Li, B. (2022). Stock prediction research based on generative adversarial neural networks. Computer Engineering and Applications, 58(13), 10.
[4] Zhang, R., & Zhang, H. (2021). Comparative analysis of BP neural network and ARMA-GARCH model in stock prediction. Journal of Normal University (Science Edition), (5).
[5] Huang, M., & Yao, C. (2023). Application of PSO-BP neural network model in stock prediction. Fujian Computers, 39(3), 36–40.
[6] Yuan, W., & Zhou, Z. (2023). Research on influencing factors of stock price fluctuation based on PCA-BP neural network. Commercial Exhibition Economy, (10), 99–101.
[7] Wang, J. (n.d.). Stock price prediction and portfolio research based on improved LSTM neural network (Doctoral dissertation, Xi'an University of Architecture and Technology).
[8] Zhang, X. (2024). Application research of LSTM model stock price prediction based on deep learning. Science, Technology and Innovation, (23), 23–26.
[9] Yuan, J., Pan, S., Xie, H., & Xu, W. (2024). S_AM_BiLSTM stock price prediction model integrating investor sentiment. Computer Engineering and Applications, 60(7), 274–281.
[10] Zhu, T., & Yan, Q. (2025). Stock price prediction research integrating improved variational mode decomposition and CNN-BiLSTM-AM. Journal of Liaoning University of Technology (Natural Science Edition), 45(2), 134–140.