# Stock Price Prediction Based On Neural Networks Incorporating Attention Mechanisms

XinRui Huang

School of Economics and Management,Xidian University,Xi'an, China

xinrui_huang@outlook.com

**Abstract.** Aiming at the nonlinearity of stock prices, this paper first utilizes deep learning neural networks, including Recurrent Neural Network, Long Short-Term Memory and Gated Recurrent Unit, which are known for their strong capability to fit nonlinear function relationships. These networks are used to construct the basic prediction model. Then, the Attention mechanism was introduced to further enhance the model by distinguishing the importance of different information for stock prediction. In this paper, I conduct experimental analysis on the historical stock data of Ping An Bank, spanning a period from from January 4, 2002 to December 30, 2022. Using LSTM, GRU, LSTM-Attention, and GRU-Attention models to predict stock prices. Experimental results show that the GRU model is better than the LSTM model in the three evaluation indexes, and the prediction effect of the model is further improved after adding the attention mechanism. It is proved that the GRU model has a better comprehensive effect, and the attention mechanism is effective and feasible for the optimization of the predictive model.

**Keywords:** Stock market prediction, RNN, GRU, LSTM, Attention Mechanism

## 1 Introduction

The stock market holds a significant position in the field of finance, playing a critical role in capital allocation and economic stability, and its price changes also reflect the changes in market supply and demand, the current economic situation, and business performances or overseas markets[1]. Therefore, grasping the trend of stock price movements is very important and necessary, and how to make more accurate forecasts of stock prices has been the concern of many scholars. However, stock price changes can be affected by many factors, and the stock price is a composite reflection of all available information about the listed company, which makes the stock data non-stationary, blaring and chaotic[2]. In that case, successful stock price forecasting is very difficult and challenging. Based on some past studies, we know deep learning neural networks have better fitting ability for nonlinear functional relationships. Considering the characteristics of nonlinearity and high complexity of stock data, the neural network model has some advantages in the problem of stock price prediction. In


Y. Jiao et al. (eds.), *Proceedings of the 3rd International Conference on Internet Finance and Digital Economy (ICIFDE 2023)*, Atlantis Highlights in Economics, Business and Management 1,
https://doi.org/10.2991/978-94-6463-270-5_57

this paper, I build stock price prediction models based on several neural networks and add an attention mechanism to improve the prediction accuracy.

## 2 Relative work

When conducting research related to stock price forecasting, researchers have applied many models to try to forecast stock price, mostly utilizing time series forecasting models and some machine learning algorithms. Among the time series forecasting models researchers have mainly adopted the Auto-Regressive Moving Average (ARMA) model[3] and the Auto regressive Integrated Moving Average (ARIMA) model[4]. However, time series forecasting models assume that there is some specific linear relationship between the data, which ignores the potential nonlinear relationship in the stock price series, leads to its not very accuracy prediction. Due to the nonlinear nature of stock prices, many scholars have chosen to use machine learning models that are good at handling nonlinear data for stock price prediction. Support Vector Machine (SVM)[5], Logistic Regression (LR), and Decision Tree (DT), Deep Learning (DL) and other machine learning algorithms are widely used in stock price prediction research, among which Deep Learning stands out for its use of deep neural networks and deeper understanding of data. Recently, because of the advancements in computer technology and other related fields, machine learning has emerged as a significant technique for forecasting stock trends. Neural networks are a common machine learning method used in predicting stock trends, which can identify nonlinear relationships in data and deliver better prediction results[6]. Several deep learning models such as artificial neural networks (ANN)[7], multilayer perceptrons (MLP), group methods for data processing (GMDH), convolutional neural networks (CNN)[8], recurrent neural network (RNN), as well as its variants Long Short-Term Memory (LSTM), and Gate Recurrent Unit(GRU) neural networks have become the focus of recent research in various fields, including the finance[9]. There are also many hybrid methods that have also been used to improve stock price prediction models by exploiting the unique advantages of each method [10].

## 3 Methodology

### 3.1 Neural Network

Recurrent Neural Network(RNN) is a kind of directed recurrent neural networks consisting of a chain of units connected together. Through the connection of the units, recursive neural networks can store the relationship between the input of a neuron at the current moment and the output of the previous moment, which is superior to other types of neural networks in handling time series data.

RNN is frequently used to analyze prediction sequence data, the most basic structure of an RNN is a sequence of hidden layers, each of which receives the output of the hidden layer from the previous time step and the input from the current time step. This structure allows the RNN to save and update its understanding of the sequence.

However, some research shows that RNN forgets the previous state information as time passes, so Long Short-Time Recurrent Neural Network(LSTM) is introduced. LSTM is highly adept at processing and forecasting significant events in time series data that exhibit long intervals and delays. It has shown exceptional performance across several fields in recent years.

The LSTM network structure uses the mechanism of control gates and consists of memory cells, input gates, output gates, and forgetting gates, as shown in Fig.1. Gate Recurrent Unit(GRU) is also a form of recurrent neural network. GRU is simpler and more efficient than LSTM, and it can solve the long dependency problem in RNN networks with good prediction results. In GRU, the reset gate dictates the degree of historical data retention and the amount of new data to be incorporated, thereby aiding in the capture of short-term patterns within time-series data. The update gate determines the amount of data to be discarded, acting as a filter to exclude irrelevant information from the sequence. When applied to stock data, the reset gate facilitates the capture of short-term dependencies in the stock time series, while the update gate enables the extraction of long-term dependencies.

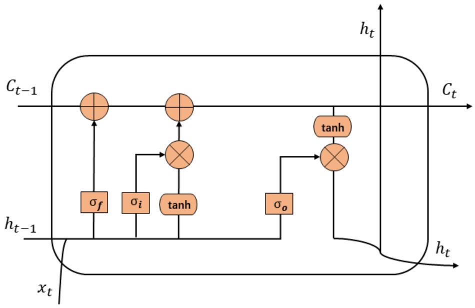


**Fig. 1.** LSTM neuron structure

### 3.2 Attention mechanism

Stock prices are affected by a variety of factors, and the impact of factor characteristics on the time series at different points in time may vary with each predicted value. However, the traditional recurrent neural network will give the same weight to the input values at all time points. So the researchers introduced the attention mechanism, which is a resource allocation mechanism similar to the attention process of the human brain. Through the autonomous learning of the model, the attention mechanism can create weight coefficients for the input variables and dynamically assign them to each region where the model receives information. By combining the weighting coefficients with the original input variables, new input variables can be obtained. There are various ways to combine the attention mechanism with the LSTM or GRU layer in the implementation of the attention mechanism[11]. This can improve the performance of the model in tasks where the input sequence is long and the output depends on a specific part of the input. This can help the model to better capture the long term dependencies in the sequence and enhance the prediction accuracy and performance of the model. This paper chooses to add an attention mechanism between two LSTM layers or two GRU layers to help the model better capture and focus on the parts of the input sequence that are more relevant to the predicted outputs.

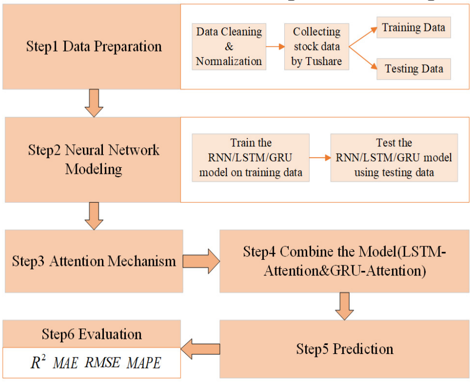


**Fig. 2.** Overall model flow chart

Fig.2 shows the complete process of stock forecasting in this article. Including data preparation, building a neural network model, and adding the attention mechanism on the basis of the built LSTM and GRU model to improve model performance. After that, different models are used to predict stocks separately, and finally four indicators, including $R^2$, MAE, RMSE, and MAPE are used to evaluate the prediction results of the models and compare the performance of different models.

## 4 Results and analysis

### 4.1 Data acquisition and processing

Historical stock data for Ping An Bank (000001.SZ) from January 4, 2002 to December 30, 2022 were obtained through Tushare, with a total of 4,936 valid data. The data include five indicators: opening price, high price, low price, closing price, and current day's volume. And all indicators are used to forecast tomorrow's closing price. In this paper, the data are divided into training set and test set, and the first 4336 data are selected as the training set, and the model is trained to use the last 500 data as the test set for prediction. For the data, the MinMaxScaler() function is used to normalize the data to between 0 and 1.

### 4.2 Experimental setup

In this paper, the past N days data are used to predict the N+1th day's data, and here the window size N is set to 48 days. Python is selected as a programming language to build a neural network based on the Keras framework to realize the proposed model, because is highly encapsulated, modular, simple, easy to extend, and its fine-tuning steps are simple.

To be more specific, this paper uses several recurrent neural network architectures to predict the N+1th day's closing price. The most basic model is the RNN, the model first processes the sequence input through a SimpleRNN layer containing 49 neurons. This layer uses tanh as the activation function and returns the complete output sequence for processing by the next layer. Next, a Dropout layer is added in order to avoid overfitting and to improve the generalization ability of the model. Next, the model passed through another SimpleRNN layer with 49 neurons and used tanh as the activation function, but this time only the final output of the sequence was returned. The model then passed through a second Dropout layer to further prevent overfitting. After passing through these sequence processing layers, the model passes the output to a fully connected layer (Dense layer) that contains 5 neurons and uses tanh as the activation function. Finally, the model passes a single neuron fully connected layer to produce the final prediction. This layer has no activation function and therefore can output predictions over an arbitrary range. The choice of the number of neurons in this model is based on Bayesian optimization. We used Bayesian optimization to search for the optimal number of neurons, resulting in a final configuration of 49 neurons.

Based on this base model, I also constructed more complex LSTM and GRU models. These models are structurally similar to the base model, but replace the SimpleRNN layers with more complex LSTM or GRU layers. These models also use the same number of neurons and the same optimization strategies and regularization methods are used during training. Next, based on the previously constructed LSTM and GRU models, the LSTM-Attention and GRU-Attention models were constructed by inserting an attention layer between the two hidden layers. Specifically, our LSTM-Attention and GRU-Attention models first process the input sequence using an LSTM or GRU layer with a Tanh activation function. Then, the models pass the output to the attention layer. The attention layer generates a weight for each time step that reflects the importance of that time step for the final prediction. These weights are then used to perform a weighted average on the output of the hidden layer to generate a new context vector. This context vector captures the most important information in the input sequence and is therefore used as input to subsequent layers. By adding an attention layer between the two hidden layers and assigning different weights to each time step of the input sequence, the model is able to pay more attention to the important features when learning the sequence information, which improves the prediction performance of the model.

## 4.3 Model Training

In the training part, this paper uses the Adaptive Moment Estimation(Adam) optimizer for updating the parameters of the model and setting the learning rate to 0.001. It is because Adam optimizer combines the advantages of AdaGrad, which works well with sparse gradients, and RMSProp, which is useful in online and non-stationary settings. In addition, the mean square error is chosen as the loss function.

The training set is used as input, and the amount of data fed to each batch is set to 64 for 100 epochs, and this paper also uses a callback function named 'ModelCheckpoint' provided by Keras. This function can save the model weights that achieved the lowest loss on the validation set, which allows us to use the best model for further prediction even if the model's performance deteriorates in later epochs due to overfitting.

### 4.4 Evaluation indicators

After the model is trained, we can use it to make predictions, and we also need to evaluate the performance of the model, considering that a single metric cannot be judged, so in this paper, the coefficient of determination ($R^2$), mean absolute error (MAE), root mean square error (RMSE), and mean absolute percentage error (MAPE) are used as indicators to evaluate the model.

### 4.5 Forecast

Firstly, the prediction models were constructed based on RNN, LSTM and GRU models, respectively. The plots are drawn using curves of different colors representing the predicted and true values, which show the prediction effect of different models more visually and facilitate observation and comparison.

According to the Fig.3, Fig4, Fig5 and Table 1, the predicted values, represented by the red curve, and the actual values, represented by the blue curve, have a similar overall trend, but the red curve is lagging.

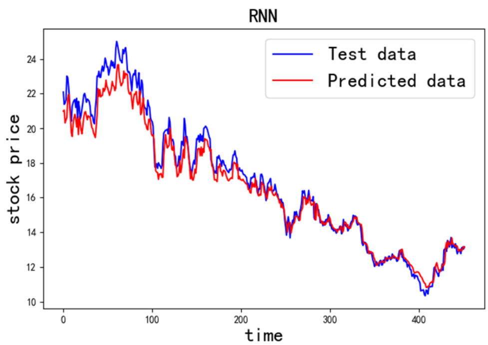


**Fig. 3.** Prediction by RNN Model

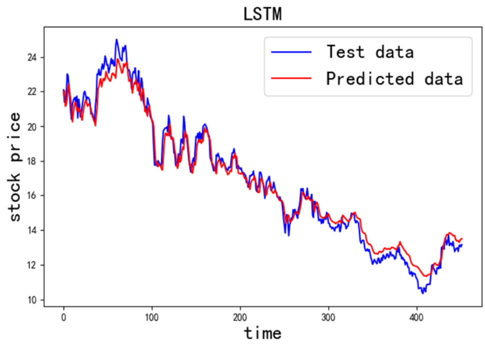


**Fig. 4.** Prediction by LSTM Model

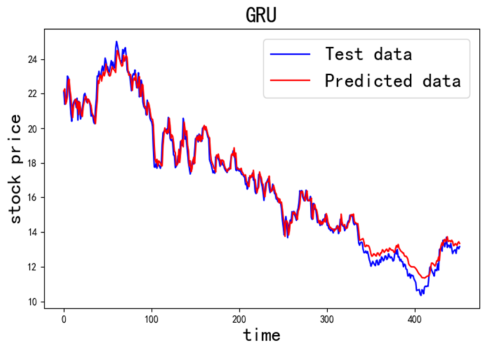


**Fig. 5.** Prediction by GRU Model

The attention mechanism is added to the original LSTM model and GRU model as shown below.

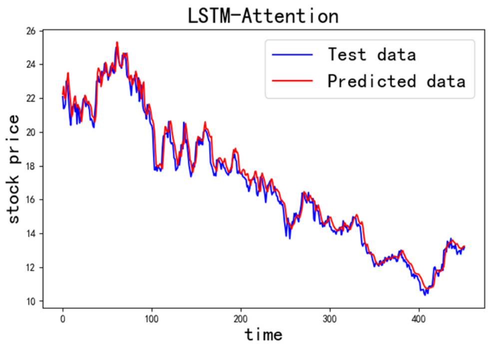


**Fig. 6.** Prediction by LSTM-Attention Model

It can be seen in Fig.6, using LSTM-Attention for prediction outperforms using the model when the attention mechanism is not added. And as evident from Fig.7, GRU also performs optimally when the attention mechanism is incorporated.

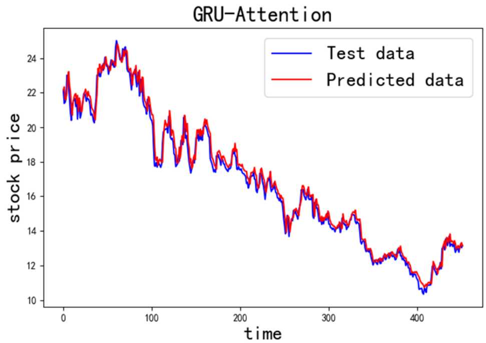


**Fig. 7.** Prediction by GRU-Attention Model

**Table 1.** RNN, LSTM, GRU, LSTM- and GRU-Attention Model Results

| Evaluation indicators | RNN | LSTM | GRU | LSTM-Attention | GRU-Attention |
|---|---|---|---|---|---|
| $R^2$ | 0.956 | 0.960 | 0.975 | 0.981 | 0.986 |
| MAE | 0.523 | 0.450 | 0.389 | 0.400 | 0.353 |
| RMSE | 0.707 | 0.565 | 0.480 | 0.525 | 0.454 |
| MAPE | 0.029 | 0.028 | 0.023 | 0.023 | 0.021 |

The high $R^2$value of 0.9859 indicates the model accurately predicts the variance. The small MAE and RMSE values, 0.3525 and 0.4538 respectively, highlight the model's minimal error rate. Lastly, the MAPE value of signifies a low average percentage deviation from actual values, further confirming the model's robust performance. From the above experimental results, it is obvious that the prediction effect of GRU model is better than that of LSTM model without adding the Attention mechanism, which is mainly due to the fact that GRU model optimally combines the forgetting gate and update gate in LSTM into a reset gate, which requires less training parameters and can prevent overfitting. And when the Attention mechanism is added, the prediction results are better than those of the neural network model alone. This is because the model pays the same attention to the features at different times when the Attention mechanism is not added, and the addition of the Attention mechanism enables to assign different attention weights to the features at different locations to advance the accuracy of prediction.

## 5 Conclusions

This paper constructs several neural network models based on RNN, LSTM and GRU for stock prediction, and conducts experiments with Ping An Bank's stock as an example. The experiments demonstrated that the model yields highly accurate predictions, particularly when we integrated the Attention mechanism into the original neural network. This allowed the model to focus on the most critical features in the time series data, thereby substantially increasing the accuracy of its predictions. Although the prediction model constructed in this paper has a good prediction effect, it is strongly dependent on stock data and does not combine with techniques such as text sentiment analysis to quantify stock market news sentiment information and stockholders' emotions, however, Stock price fluctuations are often influenced by a variety of factors, and predictions of future stock prices based solely on historical data have low credibility. For future work, I plan to refine the model by incorporating additional types of data and using multi-level strategies to improve the accuracy of future stock price predictions. This could include incorporating quantitative data on market news sentiment and stakeholder emotions, among other factors, to create a more robust and comprehensive prediction model.

## References


1. R. Akita, A. Yoshihara, T. Matsubara and K. Uehara, "Deep learning for stock prediction using numerical and textual information", 2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS), Okayama, Japan, 2016, pp. 1-6.
2. S. Selvin, R. Vinayakumar, E. A. Gopalakrishnan, V. K. Menon and K. P. Soman, "Stock price prediction using LSTM, RNN and CNN-sliding window model", 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Udupi, India, 2017, pp. 1643-1647.
3. M. F. Anaghi and Y. Norouzi, "A model for stock price forecasting based on ARMA systems", 2012 2nd International Conference on Advances in Computational Tools for Engineering Applications (ACTEA), Beirut, Lebanon, 2012, pp. 265-268.
4. Jan G. De Gooijer, Rob J. Hyndman, "25 years of time series forecasting", International Journal of Forecasting, vol.22, Issue 3,2006, pp. 443-473
5. L. J. Cao and F. E. H. Tay, "Support vector machine with adaptive parameters in financial time series forecasting", in IEEE Transactions on Neural Networks, vol. 14, no. 6, pp. 1506-1518, Nov. 2003.
6. W. Chen, M. R. Jiang, W. G. Zhang, Z. S. Chen, "A novel graph convolutional feature based convolutional neural network for stock trend prediction", Information Sciences, vol. 556, 2021, pp. 67-94,
7. J. Patel, S. Shah, P. Thakkar, K Kotecha, “Predicting stock and stock price index movement using Trend Deterministic Data Preparation and machine learning techniques”,Expert Systems with Applications, vol. 42, Issue 1, 2015, pp. 259-268.
8. E. Hoseinzade, S. Haratizadeh. “CNNpred: CNN-based stock market prediction using a diverse set of variables," Expert Systems with Applications, vol. 129, 2019,pp. 273-285.
9. A. S. Saud, S. Shakya, “Analysis of look back period for stock price prediction with RNN variants: A case study on banking sector of NEPSE”,Procedia Computer Science, vol.167,2020,pp.788-798
10. J. J. Wang, J. Z. Wang, Z.G. Zhang and S.P. Guo, "Stock index forecasting based on a hybrid model", Omega, vol. 40, 2012, pp. 758-766
11. J. Y. Qiu, B. Wang, and C. Zhou, “Forecasting stock prices with long-short term memory neural network based on attention mechanism,” in PLoS ONE, vol. 15, no. 1, Jan. 2020.