Proceedings of ICFTBA 2024 Workshop: Finance in the Age of Environmental Risks and Sustainability
DOI: 10.54254/2754-1169/94/2024OX0185

A Research of Fine-Tuning CNN-LSTM Model for Gold Price Prediction

Yang Luo[1,a,*]

[1]School of Intelligent Systems Science and Engineering, Jinan University, Zhuhai, 519070, China
a. ly353@stu2021.jnu.edu.cn
**corresponding author*

Abstract: Gold price and foreign exchange price prediction are usually regarded as serialized data with strong time correlation. It is precisely because of this feature that the performance of traditional convolutional neural networks is often not as good as that of recurrent neural networks. This paper fine-tunes the CNN-LSTM model in two ways. The first is to simplify the input data, convert the time series into binary data, and change the model task from predicting prices to predicting price fluctuations. The second is to add a dropout layer and use the dropout layer to improve the model's overfitting problem. The results demonstrate that the improved model outperforms the first model, with prediction accuracy rising from 54% to 57%. Furthermore, the refined model is not limited to processing the gold price series; it can also be applied to other time series, and when paired with other models, it could yield even better outcomes.

Keywords: CNN-LSTM Model, Gold Price Prediction, Time Series Data.

1. Introduction

As one of the most important commodities in the world, gold has always been valued by the capital market. The fluctuation of gold prices is affected by many factors [1]. Investors can make investment activities based on gold prices, and governments can also formulate corresponding economic policies based on gold prices.

Faced with extreme market shocks and emerging markets, such as the global financial crisis that began in 2007, the nominal price of gold has risen by about 42% since the crisis [2]. Gold acts as a safe haven, especially for stock markets in developed countries. Holding gold can minimize losses when it is most needed. Therefore, predicting gold price fluctuations has become crucial.

With machine learning and deep learning technologies developing rapidly in recent years, people have started attempting to use these cutting-edge technologies to increase the accuracy of gold price prediction. Due to their benefits in processing time series data, deep learning models—especially those that include long short-term memory networks (LSTMs) and convolutional neural networks (CNNs)—have been widely used in the prediction of gold prices because of their advantages in processing time series data. These models can increase prediction accuracy by automatically extracting complicated features from gold price data and learning long-term correlations from time series.

However, applying deep learning to gold price prediction is not without challenges. The high dimensionality, noise, and many unpredictable factors of financial market data increase the difficulty of model prediction. In addition, the overfitting problem of complex models is also an issue that needs to be paid attention to in practical applications.

To overcome these challenges, researchers continue to explore new model structures and algorithm optimization strategies. For example, by introducing regularization techniques, using more complex feature engineering techniques, etc., the generalization ability and prediction performance of the model can be improved.

2. Literature Review

Over the past few years, the use of machine learning and deep learning models to predict the price of gold has become a trend and is being studied and used by an increasing number of academics. These research have also yielded a wealth of helpful techniques and insights. Shafiee and Topal examined the global gold market and prices from January 1968 to December 2008 [3]. The essay examined the relationship between the price of gold and other significant influencing factors and concluded that, in the long run, the prices of both gold and crude oil are comparatively stable.

Makala and Li used SVM model for predicting the price of gold and by comparing it with ARIMA model, they present the results of their study that SVM (polynomial kernel) outperforms ARIMA and other SVM models (linear kernel and radial basis function kernel) in predicting the price of gold [4]. In order to create a combined LSTM-XGBoost prediction model that is appropriate for short-term traffic flow prediction as well as other related multivariate time series prediction fields, Zhang and Zhang merged the attributes of the LSTM-XGBoost models [5].

Livieris et al. proposed a new deep learning model for accurately predicting gold prices and their changes [6]. The study integrated long short-term memory networks and convolutional neural networks (CNNs). CNNs were used to extract time series features and input those data into long-term memory networks. The experimental results showed that the hybrid model outperformed the traditional time series model in both regression and classification tasks, and pointed out that the SNN model can be used and improved in future prediction tasks [7].

3. Methodology

The main contribution of this paper is to improve the existing CNN-LSTM model, add a dropout layer to improve the problem of LSTM model’s easy overfitting, and obtain higher classification accuracy while simplifying the model task.

The advantage of convolutional neural network is that it can extract the features of input data, but its disadvantage is that it cannot handle time series well and is prone to overfitting when the amount of data is limited. LSTM is good at capturing long-term dependencies and can capture long-term dependencies in sequence data.

The fine-tuning model proposed in this paper consists of two parts: first, a dropout layer is added to the CNN-LSTM model to avoid overfitting; when compiling the model, a binary cross entropy loss function and accuracy index are used to perform binary classification predictions on gold prices. The preprocessing of the data and the convolutional, pooling, LSTM, and dropout layers—which make up the fine-tuning model's core—are then covered in this study.

3.1. Data Preprocessing

For the last five years, from 2019-08-14 to 2024-05-30, this study chooses gold price data from the World Gold Council for each trading week. Based on the date, it splits the dataset into a test set and a training set, totaling 211 data from 2023-05-31 to 2024-05-30. A train set consists of 920 data. The data's descriptive statistics, which can be used to characterize the distribution's characteristics, are displayed in Table 1 and include the maximum, minimum, median, mean, standard deviation (SD), skewness, and kurtosis.

Table 1: Descriptive statistics.

Item	Quantity
Maximum	19.9419
Minimum	11.65
Median	13.8269
Mean	14.2309
SD	1.6454
Skewness	1.0012
Kurtosis	0.6677

In addition, Figure 1 shows the trend chart of the opening price of gold. To begin with, the author only retains the weekly opening price of the gold price data for forecasting, using the first-order difference algorithm. A result greater than zero means that gold prices rise, with a value of one, and less than zero means that the price falls, with a value of zero. Then, a sliding window is defined, and the training set and test set are reconstructed according to the sliding window size as the input of the model. Figure 2 shows the reconstruction process using the training set as an example.

Figure 1: Weekly gold price from August 2019 to May 2024 (Photo/Picture credit: Original).

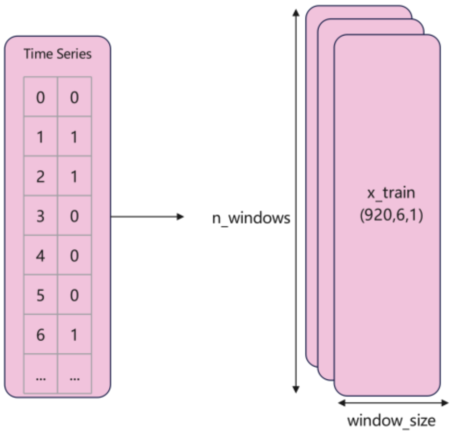

Figure 2: Training set after preprocessing (Photo/Picture credit: Original).

3.2. Key Layers

The convolutional layer is the core layer for feature extraction in CNN [8]. The layer is made up of numerous feature maps, each of which has several neurons. The convolution kernel connects each neuron to the local region of the preceding layer's feature map.

The introduction of the pooling layer can reduce the dimension of the feature map, thereby reducing the number of parameters. Because the feature map output by the convolution layer may contain a large number of features, this can easily lead to the problem of overfitting of the model.

LSTM neural networks are specially designed recurrent neural networks (RNNs) [9]. In the LSTM network, the forget gate, input gate, memory cell state, and output gate are the key components that make up the LSTM memory block, and they work together to process and store information. To be more specific, the information retained by the previous memory state C_{t-1} is determined by the forget gate f_t. The updated information is determined by the input gate $Input_t$ and passed to the new candidate cell information $\widehat{C_t}$ through the tach layer. Next, the previous memory cell C_{t-1} is updated, the new memory cell C_t is obtained, and the memory cell is finally output through the output layer Out_t. The basic LSTM operation procedure is illustrated by formulas (1) through (5).

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

$$C_t = f_t * C_{t-1} + i_t * \widehat{C_t} \tag{2}$$

$$Input_t = \sigma\left(W_i \odot \left[h_{t-1}, x_t\right] + b_i\right) \tag{3}$$

$$\widehat{C_t} = \tanh\left(W_C \odot \left[h_{t-1}, x_t\right] + b_C\right) \tag{4}$$

$$Out_t = \sigma\left(W_o \odot \left[h_{t-1}, x_t\right] + b_o\right) \tag{5}$$

Dropout technology can effectively solve the overfitting problem of deep neural network training with a large number of parameters [10]. The key lies in randomly deleting neural network units and connections during the training process of the model.

In this research, a model with two convolutional layers of size (2,) each, a maximum pooling layer, an LSTM layer with 200 units, an output layer with one neuron and a dense layer with 32 neurons, and a dropout layer with a dropout probability of 0.3 make up the suggested model. The proposed fine-tuning model's framework is depicted in Figure 3.

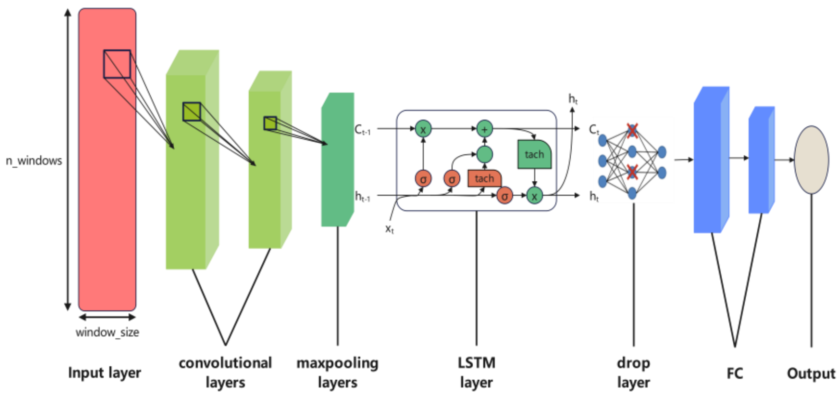

Figure 3: Proposed CNN-LSTM Fine-tuning model architecture (Photo/Picture credit: Original).

DOI: 10.54254/2754-1169/94/2024OX0185

4. Results

The author assesses the suggested CNN-LSTM model and the fine-tuned model experimentally in this section. Python 3.10.13 is used to write the implementation code on a laptop running Windows 11. Theano serves as the backend while the Keras library powers the deep learning model. The model parameter settings are shown in Table 2.

Table 2: Model Parameter settings.

Model	Parameters
$CNN-LSTM_1$	32-filter convolutional layer 64-filter convolutional layer Layer of LSTM with 100 units
$CNN-LSTM_2$	32-filter convolutional layer 64-filter convolutional layer Layer of LSTM with 200 units Fully connected layer with 32 neurons
$CNN-LSTM_3$	64-filter convolutional layer 128-filter convolutional layer Layer of LSTM with 200 units Layer of dropout with a drop probability of 0.2
$Fine-Tuned-Model_1$	32-filter convolutional layer 64-filter convolutional layer Layer of LSTM with 100 units Fully connected layer with 32 neurons
$Fine-Tuned-Model_2$	64-filter convolutional layer 128-filter convolutional layer Layer of LSTM with 200 units Layer of dropout with a drop probability of 0.3

All CNN-LSTM models and fine-tuning models have been trained for 50 epochs, with a batch size of 128, and the Adam optimization algorithm is used. The Adam optimizer adaptively modifies each parameter's learning rate and speeds up the model's convergence by combining the benefits of the Momentum and Root Mean Square Propagation methods (RMSprop).

This article evaluates each forecasting model's performance on the the classification task of estimating the direction of the gold price movement the next week. More specifically, the opening prices of the previous n weeks (i.e. window size) are analyzed and the next week's opening price is predicted to be an increase or decrease relative to the gold price of this week. Four performance metrics are employed for this binary classification problem: specificity (SPE), sensitivity (SEN), area under the curve (AUC), and accuracy (Acc).

The experimental results show that among the different models tested, $Fine-Tuned-Model_2$ performs the best with the highest accuracy (ACC) of 57.35%, followed by $CNN-LSTM_2$ with an accuracy of 54.13%. In terms of AUC metric, $Fine-Tuned-Model_2$ also performs the best with a value of 0.5678, while $CNN-LSTM_1$ and $CNN-LSTM_3$ have relatively low AUC values, 0.5282 and 0.5282 respectively. 0.5282 and 0.5290. Table 3 shows the results of the experiment.

In terms of sensitivity, $CNN-LSTM_1$ performs the best with a value of 0.6140, while $Fine-Tuned-Model_1$ has the lowest sensitivity of 0.5132. In terms of specificity, $Fine-Tuned-Model_1$ performs the best with a value of 0.54078 and $CNN-LSTM_3$ has relatively low AUC values of 0.5282 and

DOI: 10.54254/2754-1169/94/2024OX0185

0.5290, respectively. In terms of specificity, $Fine - Tuned - Model_1$ performs the best, with a value of 0.5407, while $CNN{-}LSTM_1$ has the lowest specificity, with a value of 0.4423.

Table 3: Model Parameter settings.

Model	ACC(%)	AUC	SEN	SPE
$CNN - LSTM_1$	53.211009	0.528171	0.614035	0.442308
$CNN - LSTM_2$	54.128440	0.538209	0.605263	0.471154
$CNN - LSTM_3$	53.211009	0.529015	0.596491	0.461538
$Fine - Tuned - Model_1$	53.080568	0.524887	0.513157	0.540740
$Fine - Tuned - Model_2$	57.345971	0.567821	0.571428	0.574626

Considering all the indicators, $Fine{-}Tuned{-}Model_2$ is more outstanding in overall performance, not only leading in accuracy and AUC, but also performing well in sensitivity and specificity. This may indicate that the model has been effectively optimised during the training process and is able to better balance various performance metrics. The two refined models' confusion matrix is shown in Figure 4.

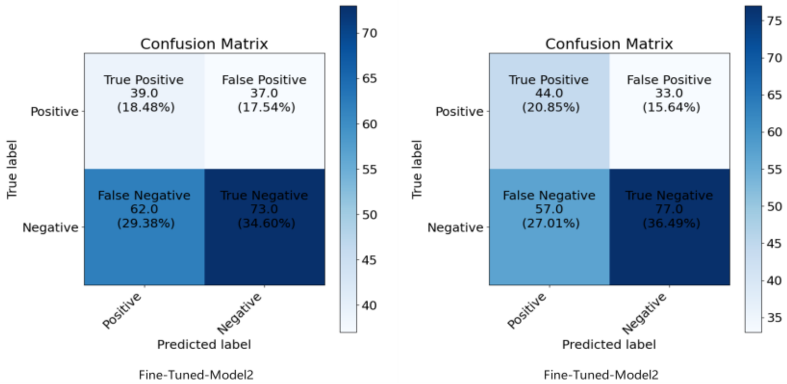

Figure 4: Models 1 and 2's refined confusion matrix (Photo/Picture credit: Original).

5. Conclusion

In this paper, the authors propose a dropout-based CNN-LSTM fine-tuning model and apply it to a classification problem. With three iterations of the original CNN-LSTM model—all of which contain two convolutional layers and varying numbers of filters—the author assesses and contrasts two iterations of the suggested fine-tuning model. Overall, the second fine-tuning model with the culling layer performed the best, with the highest accuracy and AUC, while the first fine-tuning model had the highest recall but performed poorly on the other three metrics.

Since the price of gold is also affected by a large number of unpredictable factors such as climate and natural disasters, in future work, the author will focus on exploring the application of spiking neural networks (SNNs) in financial NLP tasks to make the models operate closer to the human brain [10], and utilise the bio-inspired properties of SNNs not only to improve the ability of the models to comprehend the data of the financial market, but also to enhance their robustness in the complicated environment and enhance its robustness in complex environments.

DOI: 10.54254/2754-1169/94/2024OX0185

References

[1] Zhu, J., Han, W., & Zhang, J. (2023). Does climate risk matter for gold price volatility? Finance Research Letters, 58, 104544.

[2] Baur, D. G., & McDermott, T. K. (2010). Is gold a safe haven? international evidence. Journal of Banking & Finance, 34(8), 1886–1898.

[3] Shafiee, S., & Topal, E. (2010). An overview of global gold market and gold price forecasting. Resources Policy, 35(3), 178–189.

[4] Makala, D., & Li, Z. (2021). Prediction of gold price with ARIMA and SVM. In Journal of Physics: Conference Series (Vol. 1767, p. 012022). IOP Publishing.

[5] Zhang, X., & Zhang, Q. (2020). Short-term traffic flow prediction based on LSTM-XGBoost combination model. Computer Modeling in Engineering & Sciences, 125(1), 95–109.

[6] Livieris, I. E., Pintelas, E., & Pintelas, P. (2020). A CNN–LSTM model for gold price time-series forecasting. Neural Computing and Applications, 32, 17351–17360.

[7] AbouHassan, I., Kasabov, N., Bankar, T., Garg, R., & Bhattacharya, B. S. (2023). PAMET-SNN: Predictive associative memory for multiple time series based on spiking neural networks with case studies in economics and finance. Authorea Preprints.

[8] Li, Z., Liu, F., Yang, W., Peng, S., & Zhou, J. (2022). A survey of convolutional neural networks: Analysis, applications, and prospects. IEEE Transactions on Neural Networks and Learning Systems, 33(12), 6999–7019.

[9] Graves, A., & Graves, A. (2012). Long short-term memory. In Supervised Sequence Labelling with Recurrent Neural Networks (pp. 37–45).

[10] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. The Journal of Machine Learning Research, 15(1), 1929–1958.