



Gold Price Prediction Based on LSTM-Attention Combined Model

Junping Wu, Jihua Liu

Business School of Hubei University, Wuhan, China

Email: 1247220825@qq.com

How to cite this paper: Wu, J.P. and Liu, J.H. (2026) Gold Price Prediction Based on LSTM-Attention Combined Model. *Open Access Library Journal*, **13**: e14759. <https://doi.org/10.4236/oalib.1114759>

Received: December 16, 2025

Accepted: January 11, 2026

Published: January 14, 2026

Copyright © 2026 by author(s) and Open Access Library Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

This paper explores effective methods for predicting gold prices, proposing three modeling strategies: a standalone Long Short-Term Memory (LSTM) network, a Convolutional Self-Attention (CSA) Network, and a combination model of both. Through empirical analysis, we systematically compare the performance of these three models in gold price forecasting, with a focus on evaluating their predictive accuracy. The experimental results indicate that the LSTM-Attention combination model significantly outperforms the standalone LSTM and Convolutional Self-Attention (CSA) Network in terms of prediction accuracy, demonstrating a more comprehensive ability to capture the dynamic features of price fluctuations. This model not only showcases the effective integration of LSTM and attention mechanisms in time series forecasting but also provides a practical tool for financial decision-making, offering valuable insights for investors in a volatile market environment.

Subject Areas

Finance

Keywords

Gold, LSTM, Price Prediction, Convolutional Self-Attention (CSA) Network

1. Introduction

As one of the important indicators of the global financial market, gold prices not only reflect changes in economic conditions but also exert a profound impact on the decisions of investors and policymakers. Fluctuations in international gold prices are not only related to investors' asset allocation and risk management but also closely linked to the pulse of the global economy. As a precious metal, gold is widely regarded as a "safe haven" and often attracts substantial capital inflows

during periods of economic uncertainty and market volatility. Especially in environments of rising inflation or currency depreciation, gold's value-preserving attribute makes it a preferred choice for many investors. Interest rate decisions by central banks around the world, changes in money supply, and expectations of economic growth all indirectly affect the investment appeal of gold. In a low-interest-rate environment, the opportunity cost of holding gold decreases, prompting more investments to flow into the gold market. Coupled with the interplay of various factors such as technical analysis, market sentiment, and speculative behavior, the trend of international gold prices exhibits complex and volatile characteristics. Therefore, establishing a scientific prediction model and analytical framework is crucial for investors to achieve returns and risk control. In-depth analysis of gold price trends can not only help investors seize market opportunities but also provide effective decision-making references for policymakers, facilitating the optimization of economic policies.

2. Literature Review

Scholars at home and abroad have adopted various methods to predict gold prices. Traditional gold price prediction methods mainly rely on statistical models, such as the Autoregressive Integrated Moving Average (ARIMA) model, which performs excellently in handling linear time series. Si Jin [1] based his research on the monthly spot gold prices in London from January 1, 2015, to June 2, 2015, and conducted a predictive analysis of London gold prices in the first half of 2015 by establishing an ARIMA model. The study found that the predicted values had a high degree of fit with the actual data, and the prediction results were relatively accurate. Xu Jun [2] applied the ARMA model to the weekly gold futures data from the New York Stock Exchange (NYSE) spanning from January 2006 to January 2016. Xu Jingyi and Kong Mengqi [3] selected the New York COMEX gold futures prices from January 2, 2020, to December 31, 2021, as research samples, conducted empirical analysis by constructing an ARIMA model, and concluded that gold prices would maintain steady growth in the short term. Considering the trending and seasonal characteristics of gold prices, Wang Peizhi and Mei Zhan [4] improved the traditional ARIMA model and constructed a SARIMA model that eliminates trend and seasonal factors, finding that the SARIMA model improved prediction accuracy to a certain extent. Pan Xueyan [5] established an ARMA-GARCH model for gold prices in the London Gold Market based on time series theory and conducted empirical analysis, which revealed that the model had high prediction accuracy. Luo Hongben [6] proposed using a gray-ARIMA hybrid model for time series prediction, improving the gray model with the PSO algorithm and the ARIMA model with the genetic algorithm to enhance model precision.

However, the limitation of the ARIMA model lies in its inability to effectively capture the potential nonlinear characteristics and complex dependencies in gold price trends. In recent years, the rapid development of deep learning technology

has attracted widespread attention. Among various deep learning models, the Long Short-Term Memory (LSTM) network has gradually gained application due to its strong modeling capabilities and adaptability to time series data. Liu Lu *et al.* [7] compared the prediction effects of unidirectional LSTM, bidirectional LSTM, and ARIMA models on gold futures price series and concluded that the bidirectional LSTM achieved the best performance. Kim and Chang [8] used the LSTM model to predict stock price volatility, combining LSTM with GARCH-family models and achieving favorable results. Ouyang Hongbing *et al.* [9] integrated wavelet analysis with the LSTM model to predict the Dow Jones Industrial Average and compared its performance with four other models (multilayer perceptron, SVR, KNN, and GARCH), finding that the LSTM model had the highest prediction accuracy. Yang Chen *et al.* [10] constructed an EMD-LSTM model and found that this model exhibited better performance and higher accuracy in predicting international gold futures prices, enabling more precise forecasting of price trends. Jing Zhigang *et al.* [11] applied wavelet analysis to decompose gold prices and input the decomposed data into the LS-SVM-ARIMA model, which yielded more accurate predictions of gold price trends. In 2017, Vaswani A *et al.* [12] abandoned the use of RNN and CNN, solely adopting the Attention mechanism with extensive application of Self-attention to construct the Transformer model for learning text representations. Subsequently, the Transformer has been widely applied in fields such as computer vision, speech recognition, and natural language processing, achieving excellent results in all these domains. In the field of time series prediction, many scholars have combined Attention with other models in recent years to improve prediction performance. Jin *et al.* [13] constructed an Inter-Series Attention model for predicting COVID-19 data across different regions, achieving outstanding results in predicting multiple indicators. Li S *et al.* [14] first applied the Transformer to time series prediction problems and proposed several improvements, with the final experimental results outperforming many models such as ARIMA, DeepAR, and DeepState. Wu *et al.* [15] built a time series prediction model based on the Transformer and conducted a case study on influenza patient number prediction, ultimately demonstrating that the Transformer model outperformed other deep learning models. Cai L [16] applied the Transformer model to real-time traffic condition prediction, efficiently achieving accurate traffic forecasting.

Although deep learning technology has shown great potential in time series prediction, the advantages of combined models have become increasingly prominent, as they can help improve model performance. Examples of such effective combined models include the CNN-LSTM model. Considering the excellent characteristics of LSTM in time series prediction and the superior feature extraction capability of CNN, Li C and Zhang X *et al.* [17] constructed a CNN-LSTM model to predict stock prices and added the Attention mechanism to enhance the accuracy and scalability of price prediction. Based on the noise and volatility of the stock market, scholars such as Jin [18] incorporated investor sentiment as an in-

fluencing factor, adopted Empirical Mode Decomposition (EMD) to gradually decompose complex stock price series, and combined LSTM with the Attention mechanism for price prediction, thereby improving model accuracy. Xu Jiali [19] constructed a Shanghai gold futures prediction model based on the Attention mechanism combined with CNN-LSTM, conducting empirical analysis using the closing prices of the main Shanghai gold futures contracts on the Shanghai Gold Exchange and related indicators from January 9, 2008, to December 31, 2020. Wang Hong [20] proposed a relation extraction method that combines the LSTM model with the Attention mechanism, converting text information into vectors, extracting local text features, and calculating the importance of correlations between the input and output of the LSTM model through the Attention mechanism. The Attention mechanism can effectively capture complex dynamic patterns in data and achieve excellent prediction results in time series data.

Building on existing research, the construction of combined models still requires in-depth exploration. Although deep learning models have demonstrated outstanding performance, they often face issues such as large data requirements, long training times, and poor interpretability. Therefore, existing research can consider combining the advantages of traditional models to explore more robust hybrid methods. This paper will further conduct relevant research based on the combined model of LSTM and the Attention mechanism, taking gold price prediction as an example. The LSTM model excels at capturing long-term and short-term dependencies in time series data, while the Attention mechanism better handles correlations within time series. Thus, combining the advantages of these two models may yield higher prediction accuracy. In addition, we will compare the prediction accuracy with models using LSTM or the Attention mechanism alone to explore methods for improving prediction precision.

3. Model Introduction

3.1. LSTM Model

Long Short-Term Memory (LSTM) is a special type of Recurrent Neural Network (RNN) designed to address the problems of gradient vanishing and gradient explosion faced by standard RNNs when learning long-term dependencies in time series data.

The core of LSTM is the cell. Unlike traditional RNN cells, the LSTM cell introduces a “cell state,” which is a long channel running through the entire network to maintain long-term dependencies of information. The LSTM cell mainly consists of several important components: forget gate, input gate, cell state update, and output gate. Its structure is shown in **Figure 1**, where b_f is the bias of the forget gate, W_f represents the weight, and σ is the sigmoid activation function.

The forget gate is responsible for determining how much of the previous moment's cell state C_{t-1} is retained in the current moment's cell state, that is, deciding what information to discard from the cell state. This gate reads the hidden state h_{t-1} and the current input x_t . Through the sigmoid activation function,

the forget gate outputs a value between 0 and 1, where 0 indicates complete forgetting and 1 indicates complete retention. The calculation formula is as follows (1):

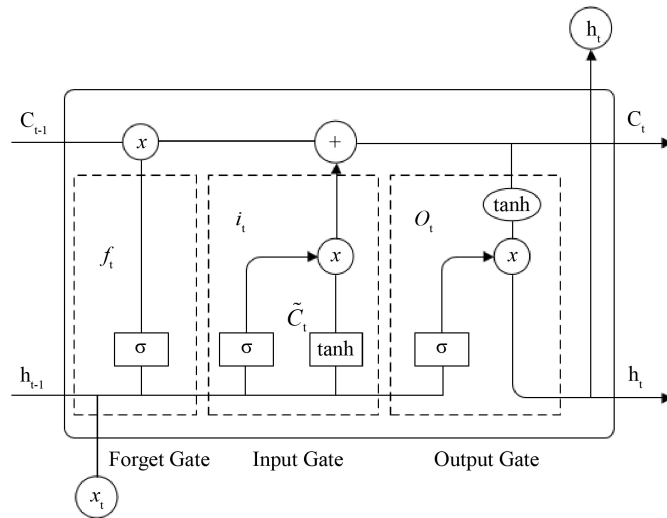


Figure 1. LSTM structure.

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

The input gate is responsible for determining how much of the current moment’s input is retained in the current moment’s cell state, consisting of two parts. The first part is the sigmoid layer, which decides which values to update. The calculation formula is as follows (2):

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \tag{2}$$

The second part is the tanh layer, which updates the information to be updated into the cell state. The tanh layer creates a new cell state value vector \tilde{C}_t , which will be added to the state. The calculation formulas are as follows (3) and (4):

$$\tilde{C}_t = \tanh W_c[h_{t-1}, x_t] + b_c \tag{3}$$

$$C_t = f_t C_{t-1} + i_t \tilde{C}_t \tag{4}$$

The output gate is responsible for determining how much of the current moment’s cell state is output. A sigmoid layer is used to determine which parts of the cell state will be output. The cell state is processed through tanh to obtain a value between -1 and 1 , which is then multiplied by the output of the sigmoid gate, ultimately outputting only the determined parts. The calculation formulas are as follows (5) and (6):

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

$$h_t = O_t \tanh(C_t) \tag{6}$$

3.2. Self-Attention Model

Self-attention is a mechanism widely used in the Transformer model. It allows

each element in the sequence to establish connections with itself and other elements, and assigns different weights based on these connections. In this way, the model can understand the relationships between elements in the sequence, thereby capturing long-range dependency information. The self-attention mechanism usually includes several steps: input representation, generating queries, keys, and values, calculating attention weights, outputting results, and multi-head attention. Its architecture is shown in **Figure 2**.

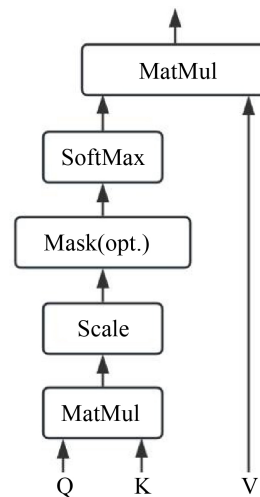


Figure 2. Self-attention architecture.

Input representation usually represents the input sequence as a matrix, where each row represents the embedding of an element (such as the word vector of a word).

Generating queries, keys, and values involves generating three vectors for each matrix X in the input sequence: Query (Q), Key (K), and Value (V). These vectors are obtained by multiplying the input vectors by different weight matrices. The calculation formula is as follows (7), where W_Q , W_K , and W_V are learnable weight matrices:

$$Q = XW_Q, K = XW_K, V = XW_V \quad (7)$$

Calculating attention weights is used to compute the similarity between queries and keys to determine the attention weight of each element in the input sequence to other elements. This is usually calculated through dot product, and then the result is scaled and normalized by Softmax to obtain normalized weights. The calculation formula is as follows (8), where d_k is the dimension of the key, used for scaling to prevent excessively large dot products:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (8)$$

After obtaining the attention weight matrix, the output result generates a new representation for each input element by weighted summation of the vectors in

the Value matrix using these weights. This process can be regarded as fusing the information of the input sequence, and the self-attention mechanism responds to the relationships between elements in the sequence.

3.3. Convolutional Self-Attention (CSA) Network

The Convolutional Self-Attention (CSA) Network proposed in this paper is a variant based on the Transformer encoder block, integrating convolutional layers and self-attention mechanisms. The design logic is as follows: gold price time series data simultaneously contains local short-term fluctuation features (e.g., daily price volatility) and long-range dependency relationships (e.g., monthly trend correlations). Convolutional layers can efficiently capture local spatial features and suppress noise interference in raw data, while the self-attention mechanism can model long-range dependencies between sequence elements. The combination of the two achieves synergistic optimization of “local feature extraction - long-range relationship modeling,” laying a foundation for improving prediction accuracy.

3.4. Model Evaluation Indicators

For the prediction results of gold prices, Mean Absolute Error (MAE), Mean Squared Error (MSE), and Coefficient of Determination R^2 are used as model evaluation indicators.

Mean Absolute Error (MAE), also known as L1 norm loss, can reflect the actual situation of prediction errors by calculating the proximity between prediction results and real data. The calculation formula is as follows (9):

$$\text{MAE} = \sum_{i=1}^n |y_i - \hat{y}_i| \quad (9)$$

Mean Squared Error (MSE) is a commonly used indicator to measure the difference between the prediction results of a prediction model and the actual observed values. The calculation formula is as follows (10):

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (10)$$

The Coefficient of Determination R^2 represents the proportion of the variability in the dependent variable that can be explained by the prediction model, and is a comprehensive evaluation indicator. It can measure the degree to which the independent variable explains the dependent variable. According to the formula, its value range is 0-1. The closer the value is to 1, the higher the contribution rate of the independent variable to the change of the dependent variable; otherwise, the explanation degree is weaker. The calculation formula is as follows (11):

$$R^2 = 1 - \frac{\sum_{i=1}^n (\hat{y}_i - y_i)^2}{\sum_{i=1}^n (\hat{y}_i - \bar{y})^2} \quad (11)$$

where n represents the number of samples, y_i represents the actual observed value, \hat{y}_i represents the predicted value (*i.e.*, the predicted value of the model at the i -th time point), and \bar{y} is the sequence mean.

4. Model Prediction Results

After the text edit has been completed, the paper is ready for the template. Duplicate the template file by using the Save As command, and use the naming convention prescribed by your journal for the name of your paper. In this newly created file, highlight all of the contents and import your prepared text file. You are now ready to style your paper.

4.1. Data Selection and Descriptive Statistics

This paper selects the LBMA gold price time series from the Goldhub platform. The LBMA gold price refers to the spot gold price announced by the London Bullion Market Association. This price is widely regarded as the benchmark price in the global gold market, especially in the financial and investment industries, and has important reference value. The unit of LBMA gold price is US dollars per ounce. The selected data spans from January 2, 2012, to March 13, 2024, totaling 3183 data points, as shown in **Figure 3**.

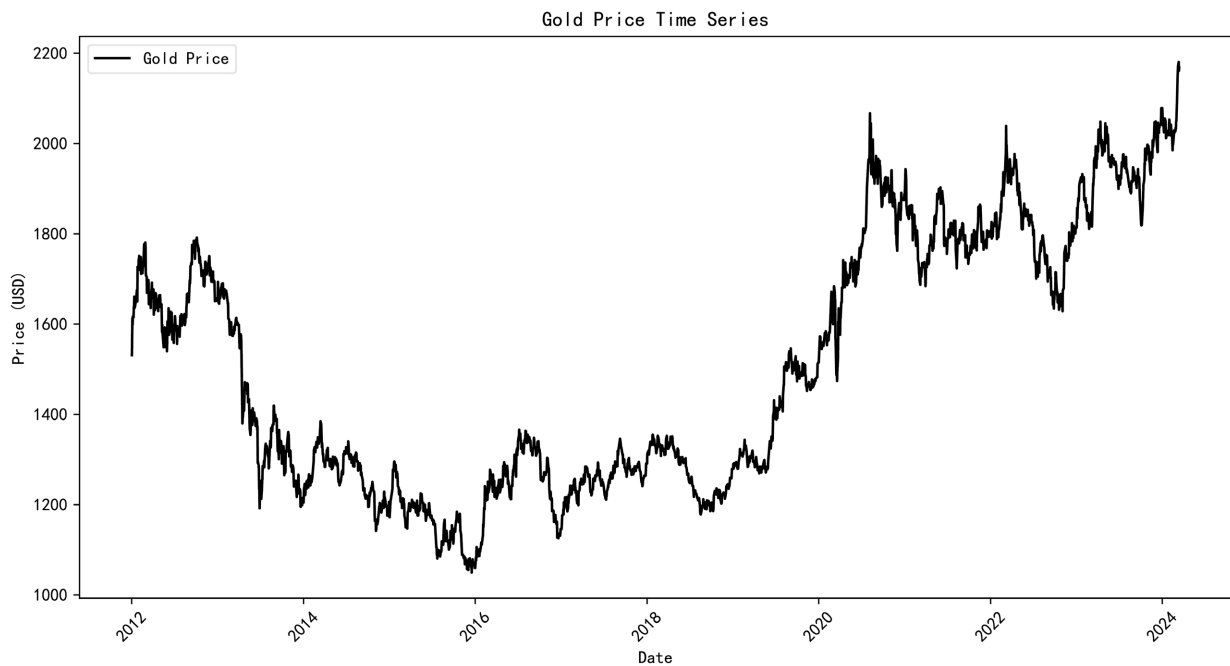


Figure 3. Time series of gold prices.

4.2. Data Preprocessing

First, confirm the sample data for modeling, retain only the time and price labels, remove other labels, and divide the data into a training set and a test set. 95% of the overall sample is used as the training set, and the remaining 5% is used as the test set for verification. Considering the temporal continuity of time series data, to avoid data leakage (*i.e.*, future information being learned by the model in advance), this study adopts a time-series splitting method to divide the training and test sets. The 95% training set is used for the model to learn long-term price

trends, while the 5% test set (covering August 2023 to March 2024) includes recent market volatility characteristics (such as economic recovery expectations and interest rate adjustment scenarios), which can effectively verify the model's generalization ability under different market environments.

4.3. Prediction Based on LSTM Model

After dividing the data into training and test sets, to eliminate the impact of dimensional differences between variables on the model, the data is first normalized to solve the comparability between data, accelerate the convergence of the training network, and improve the prediction accuracy of the model. According to the characteristics of the dataset and the requirements of the model, this paper adopts the min-max normalization method, as shown in (12), to convert the original data into the range [0, 1].

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (12)$$

where: x is the original data, and x' is the normalized data. At the same time, a function is defined to create input and output sequences, converting time series data into a format suitable for machine learning models. By setting the time step to 30, each input sample will contain data from the past 30 time steps when constructing the dataset.

To determine the optimal parameters of the LSTM model, a hyperparameter model LSTMHyperModel is first defined. The number of units in the first LSTM layer ranges from 32 to 256 with an increment of 32; the dropout rate of the first Dropout layer ranges from 0.2 to 0.5 with a step of 0.1; the number of units in the second LSTM layer ranges from 32 to 256 with an increment of 32; the dropout rate of the second Dropout layer also ranges from 0.2 to 0.5 with a step of 0.1; finally, the output layer is defined as a Dense layer with one unit; the Adam optimizer is used, and the learning rate is optimized in the range from $1e-5$ to $1e-1$, sampled logarithmically; finally, Mean Squared Error (MSE) is used as the loss function to compile the model.

After successfully defining the function model, hyperparameter optimization is performed. A RandomSearch object is created to randomly search for hyperparameters, with the optimization goal of reducing the validation loss. Finally, the hyperparameter search is started through the tuner.search method, inputting the training data X_train and target y_train, conducting training for 20 epochs, and performing validation at the same time, and finally outputting the optimal parameters. The output optimal parameters are: the number of neurons in the first LSTM layer is 256, the dropout rate of the first LSTM layer is 0.4 (used to prevent overfitting), the number of neurons in the second LSTM layer is 32 (usually used for stacking multiple LSTM layers), the dropout rate of the second LSTM layer is 0.2, and the learning rate is 0.003218.

The LSTM model is constructed using the output optimal indicators, and the previously imported Adam optimizer is used with a learning rate of 0.003218. The

loss function uses MSE output. After constructing the model, the model is trained using the training data X_{train} and target values y_{train} ; the training is conducted for 100 epochs with 32 samples per batch; at the same time, the training loss and validation loss at the end of each iteration are output. can basically predict the fluctuation of gold prices well, as shown in **Figure 4**.

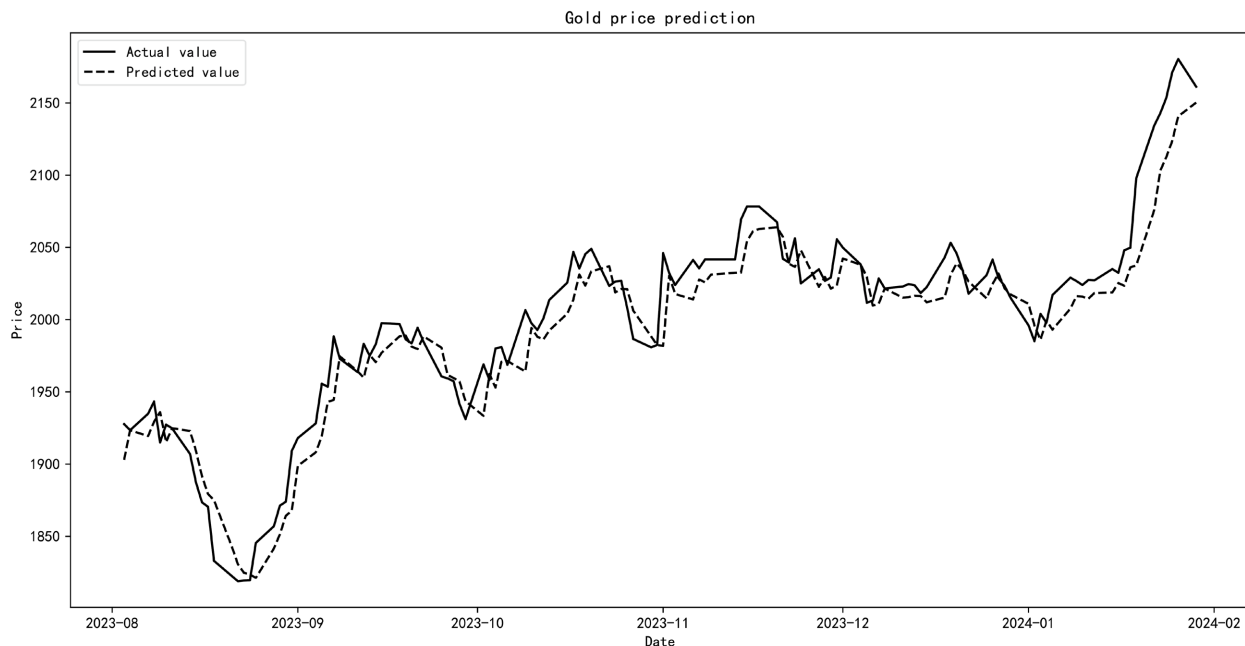


Figure 4. Comparison of LSTM model predictions.

4.4. Prediction Based on Convolutional Self-Attention (CSA) Network

When constructing the model, data processing is first performed. First, select the target column and normalize the data, then start dividing the dataset. The training set accounts for 95% of the overall data, and the test set accounts for 5%, consistent with the previous LSTM model. The time step is set to 30, and a function also needs to be defined to create input and output sequences, converting time series data into a format suitable for machine learning models.

The Convolutional Self-Attention (CSA) Network proposed in this study is based on the core architecture of the Transformer encoder block, with convolutional layers introduced for feature extraction. The design rationale is that time series data contains both local short-term fluctuation features (e.g., intraday price changes) and long-range dependency relationships (e.g., monthly trend correlations). Convolutional layers (64 filters, kernel size 3) can quickly capture local spatial features and reduce noise interference in raw data; the subsequent multi-head self-attention mechanism focuses on long-range dependencies between sequences, and the combination of the two achieves synergistic optimization of “local feature extraction - long-range relationship modeling.”

Consistent with the LSTM model, the CSA Network uses RandomSearch for

hyperparameter optimization, with the goal of minimizing validation loss. After 20 rounds of iterative search, the optimal hyperparameters are as follows: 64 convolutional layer filters, kernel size 3, layer normalization epsilon = $1e-6$, dropout rate 0.1; multi-head self-attention mechanism set to 4 heads, key dimension 64; learning rate optimization range of $1e-5$ to $1e-1$ (logarithmic sampling), optimal learning rate 0.0028; MSE as the loss function, Adam as the optimizer.

First, create an input layer whose shape is specified by the input shape parameter, then extract corresponding features through a convolutional layer, using 64 filters and a convolution kernel size of 3, perform layer normalization on the convolution output with the same padding, set the epsilon parameter, and apply a dropout rate of 0.1 to prevent overfitting. Next, use the multi-head self-attention mechanism to focus on features, using 4 heads and a key dimension of 64. Add the original input to the attention output, then reduce the feature dimension through a convolutional layer, using 32 filters and a convolution kernel size of 1, and finally use a global average pooling layer to aggregate the data. Finally, output a value through a fully connected layer, connect the input and output to construct a complete model and return it. After construction, the model is trained using the training dataset X_{train} and y_{train} , with epochs set to 100 and $batch_size$ set to 32. After training, prediction is performed, and it is found that the overall fitting accuracy is high, as shown in **Figure 5**.

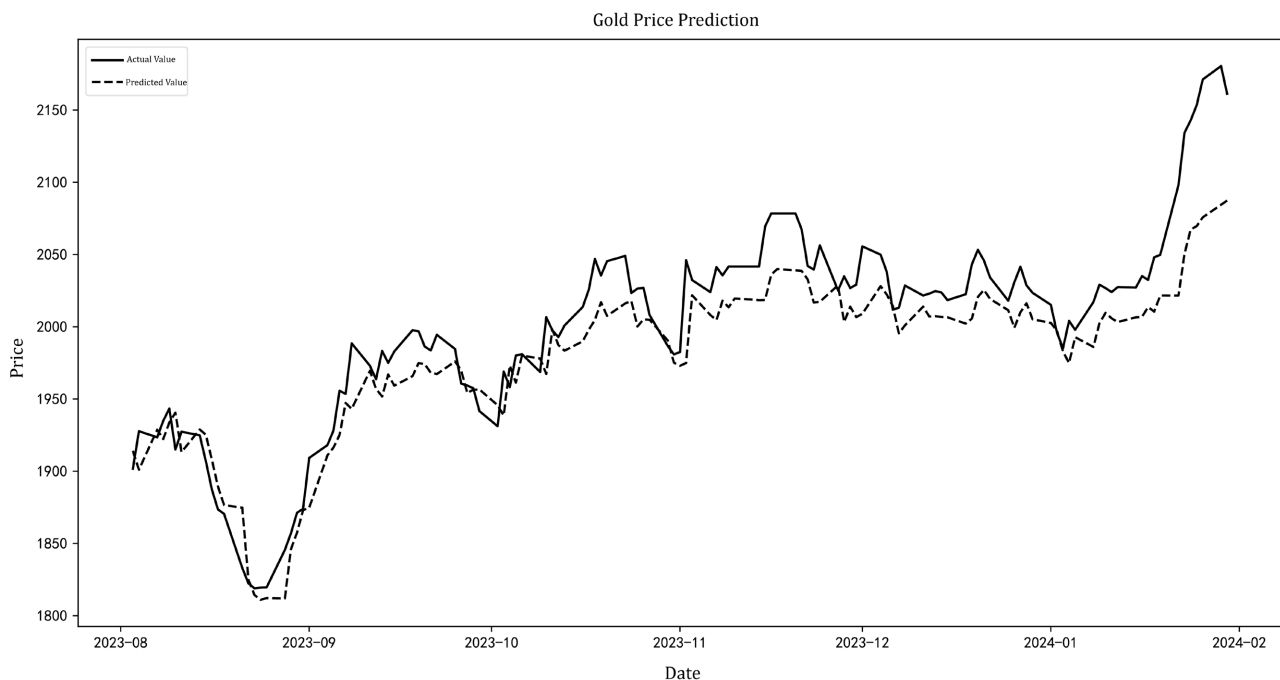


Figure 5. Comparison of Convolutional Self-Attention (CSA) network predictions.

4.5. Prediction Based on LSTM-Attention Combined Model

The LSTM-Attention combined model mainly integrates the attention mechanism on the basis of the LSTM model, which can dynamically select the most rel-

evant parts in the input sequence. The Attention layer allows the model to consider information from both the front and back when processing the input sequence, and this structure usually has a significant impact on the performance of the model output. The data processing part of the LSTM-Attention combined model is similar to that of the LSTM model. After dividing the data into training and test sets, the training set data is scaled to the range [0, 1] for normalization.

At the same time, a function is defined to create input and output sequences, converting time series data into a format suitable for machine learning models. The time step is set to 30, and the training and test set data are reshaped into the LSTM input format. A deep learning model with LSTM (Long Short-Term Memory Network) and attention mechanism is built using TensorFlow and Keras. First, define the input layer to specify the input shape, then set an LSTM layer with 50 hidden units, and set 0.2 in the Dropout layer, indicating that 20% of the LSTM output is randomly discarded during training; after defining the relevant structure of LSTM, add a multi-head attention layer, using the output generated by LSTM as input, and calculate the new output through the attention mechanism to improve the model's ability to capture key information. At the same time, scale the attention weights to avoid gradient explosion. The multi-head attention is set with 4 heads and a key dimension of 50. Then, pass the LSTM output as queries, values, and keys to this layer to calculate the attention output; after adding the multi-head attention layer, add a flatten layer to flatten the attention output so that it can be input into the fully connected layer; finally, add a fully connected layer for output, use the adam optimizer, specify the loss function as MSE.

Consistent with the previous two models, this model also uses RandomSearch for hyperparameter optimization, with the goal of minimizing validation loss. The optimal hyperparameter configuration is: LSTM layer hidden units 50, dropout rate 0.2; multi-head attention layer set to 4 heads, key dimension 50; learning rate 0.0035; MSE as the loss function, Adam as the optimizer. The hyperparameter search process is consistent with the LSTM and CSA models to ensure a fair comparison among the three.

After defining the model, the model is trained for 100 epochs with 32 samples per batch; at the same time, the training loss and validation loss at the end of each iteration are output. After training, prediction is performed, and it is found that the overall fitting accuracy is high, as shown in **Figure 6**.

5. Analysis and Comparison of Model Results

To objectively evaluate the prediction effect of the models, we calculated and compared three evaluation indicators—Mean Absolute Error (MAE), Mean Squared Error (MSE), and R^2 —based on the price prediction results of the models on the test set and the actual data. The calculation results are shown in **Table 1**:

As can be seen from the table, the gold price prediction model built based on LSTM-Attention is far superior to the traditional LSTM and Convolutional Self-Attention (CSA) Network in terms of the three indicators MAE, MSE, and

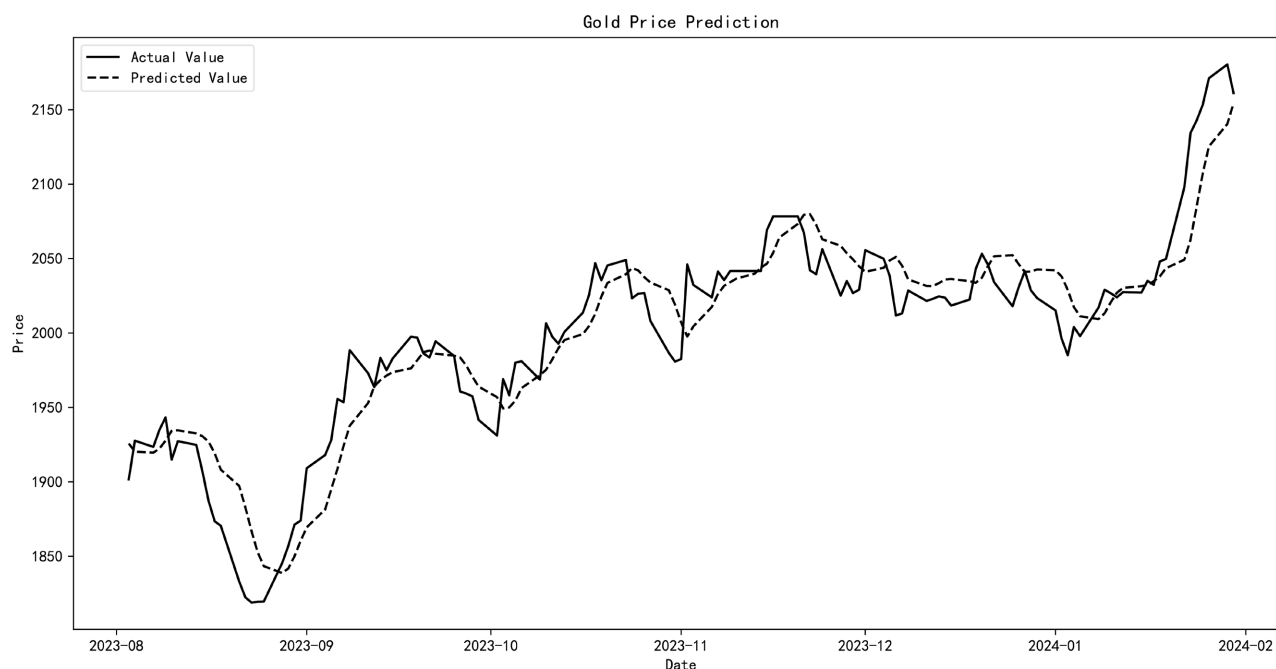


Figure 6. Comparison of LSTM-Attention combined model predictions.

Table 1. Model evaluation and comparison.

Model	MAE	MSE	R ²
LSTM	23.09	830.52	0.84
Attention	18.66	604.26	0.88
LSTM-Attention	14.46	373.09	0.92

(R^2). In addition, the accuracy of the Convolutional Self-Attention (CSA) Network in predicting price ups and downs on the test set is slightly higher than that of the LSTM model, indicating that both models can show good performance when the data volume is small and no gradient explosion occurs. At the same time, the accuracy of the LSTM-Attention combined model in predicting gold price ups and downs is 0.92, which is significantly higher than 0.88 of the Convolutional Self-Attention (CSA) Network. This indicates that the memory ability of LSTM in processing sequence data helps to improve the feature extraction effect of the attention mechanism.

Combined with the advantages of LSTM, the LSTM-Attention combined model has significantly improved the accuracy and stability of the attention mechanism in gold price prediction. In addition, this research result suggests that in feature extraction and sequence modeling tasks, combining the advantages of different models may achieve better performance and results, which is worthy of in-depth research in more fields.

6. Conclusions and Prospects

As a traditional investment asset, gold is not only regarded as a safe-haven tool

but also plays a role in preserving value in the face of economic uncertainty, inflation, and currency depreciation. Therefore, in-depth understanding and prediction of gold price trends help investors make more informed decisions in complex market environments. The combination of the LSTM-Attention model leverages the unique advantages of both, becoming an effective solution. LSTM (Long Short-Term Memory Network) has strong memory capabilities and can capture long-term dependencies in time series data. At the same time, the attention mechanism can effectively capture the dynamic relationships between different time points, thereby improving prediction accuracy. The combined model helps investors make more informed decisions in complex market environments and optimize their investment portfolios.

The research results of this paper have practical application value for investors. In terms of short-term trading decisions, the model can generate daily/weekly price trend predictions. When the predicted value is significantly higher than the current price and the (R^2) fitting degree remains above 0.9, it can be used as a buy signal; on the contrary, it can be used as a sell reference to help investors seize short-term market opportunities. In terms of long-term risk management, based on the model's prediction of price fluctuation amplitude (quantified by MAE and MSE), investors can adjust the proportion of gold in their asset portfolios—when the predicted fluctuation intensifies (MSE rises), appropriately reduce gold allocation to control risks; when the predicted trend is stable, increase allocation to give play to its safe-haven attribute.

However, gold prices are affected by a variety of factors. This paper only predicts based on the gold price time series alone and does not consider the impact of other factors. To improve prediction accuracy, future research can consider incorporating these factors into the model, adopting multivariate time series analysis or other machine learning methods, and comprehensively considering data from various aspects such as macroeconomic indicators, market sentiment, and supply-demand relationships. Specifically, three key external variables can be included: 1) US Dollar Index (USDIX), as gold is priced in US dollars, and its trend is typically negatively correlated with the strength of the US dollar; 2) Federal Funds Rate, as an increase in interest rates will increase the opportunity cost of holding gold and significantly affect gold demand; 3) Consumer Price Index (CPI), as when inflation expectations rise, the value-preserving demand for gold will increase. This will help to more comprehensively understand the fluctuations of gold prices and provide more reliable support for investment decisions.

Conflicts of Interest

The authors declare no conflicts of interest.

References

- [1] Si, J. (2016) Short-Term Prediction of Gold Prices Based on ARIMA Model. *Times Finance*, No. 17, 221-222.

- [2] Xu, J. (2017) Empirical Analysis of Gold Futures Prices Based on ARMA Model—523 Groups of Data from the New York Exchange from 2006 to 2016. *Industrial Economic Forum*, **4**, 16-22.
- [3] Xu, J. and Kong, M. (2022) Research on New York COMEX Gold Futures Prices Based on ARIMA Model. *China Business & Trade*, No. 18, 123-125.
- [4] Wang, P., Mei, Z. and Gao, W. (2022) Gold Price Prediction Based on SARIMA Model. *Electronic Technology & Software Engineering*, No. 23, 233-237.
- [5] Pan, X. (2020) Empirical Analysis of Gold Prices Based on ARIMA-GARCH Model. *Business News*, No. 20, 155-156.
- [6] Luo, H. (2014) Intelligent Hybrid Prediction Research of Financial Time Series Based on Gray-ARIMA. *The Theory and Practice of Finance and Economics*, **35**, 27-34.
- [7] Liu, L., Lou, L., Liu, X., *et al.* (2021) Comparative Study on COMEX Gold Futures Price Prediction Based on LSTM. *Journal of Changchun University of Science and Technology (Natural Science Edition)*, **44**, 129-135.
- [8] Kim, H.Y. and Won, C.H. (2018) Forecasting the Volatility of Stock Price Index: A Hybrid Model Integrating LSTM with Multiple GARCH-Type Models. *Expert Systems with Applications*, **103**, 25-37. <https://doi.org/10.1016/j.eswa.2018.03.002>
- [9] Ouyang, H., Huang, K. and Yan, H. (2020) Financial Time Series Prediction Based on LSTM Neural Network. *Chinese Journal of Management Science*, **28**, 27-35.
- [10] Yang, C. and Chen, G. (2023) International Gold Futures Price Prediction Based on EMD-LSTM. *Journal of South-Central Minzu University (Natural Science Edition)*, **42**, 857-864.
- [11] Jing, Z. and Shi, G. (2017) Gold Price Prediction Based on Wavelet Analysis Combined with LS-SVM-ARIMA Model. *Gold*, **38**, 5-8+14.
- [12] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L. and Polosukhin, I. (2017) Attention Is All You Need. CoRR.
- [13] Jin, X., Wang, Y.X. and Yan, X. (2020) Inter-Series Attention Model for COVID-19 Forecasting. arXiv:2010.13006.
- [14] Li, S., Jin, X., Xuan, Y., *et al.* (2019) Enhancing the Locality and Breaking the Memory Bottleneck of Transformer on Time Series Forecasting. *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, Vancouver, 8-14 December 2019, 5243-5253.
- [15] Wu, N., Green, B., Ben, X., *et al.* (2020) Deep Transformer Models for Time Series Forecasting: The Influenza Prevalence Case. arXiv:2001.08317.
- [16] Cai, L., Janowicz, K., Mai, G., Yan, B. and Zhu, R. (2020) Traffic Transformer: Capturing the Continuity and Periodicity of Time Series for Traffic Forecasting. *Transactions in GIS*, **24**, 736-755. <https://doi.org/10.1111/tgis.12644>
- [17] Li, C., Zhang, X., Qaosar, M., Ahmed, S., Alam, K.M.R. and Morimoto, Y. (2019) Multi-Factor Based Stock Price Prediction Using Hybrid Neural Networks with Attention Mechanism. 2019 *IEEE International Conference on Dependable, Autonomic and Secure Computing, International Conference on Pervasive Intelligence and Computing, International Conference Cloud and Big Data Computing, International Conference on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech)*, Fukuoka, 5-8 August 2019, 961-966. <https://doi.org/10.1109/dasc/picom/cbdcom/cybersci.2019.00176>
- [18] Lim, B., Arik, S.O., Loeff, N., *et al.* (2019) Temporal Fusion Transformers for Interpretable Multihorizon Time Series Forecasting. arXiv:1912.09363.
- [19] Xu, J. (2021) Shanghai Gold Futures Price Prediction Based on Attention Mechanism

Combined with CNN-LSTM. Chengdu University of Technology.

- [20] Wang, H., Shi, J. and Zhang, Z. (2018) Semantic Relation Extraction Based on LSTM with Attention Mechanism. *Application Research of Computers*, **35**, 1417-1420+1440.