LSTM Time Series Price Prediction - Deterministic Strategy Gradient Dynamic Programming

Yuxiao Duan, Yuzhuo Dang, and Honghui Chen(✉)

Key Laboratory of Information Systems Engineering, National University of Defense Technology, Changsha, Hunan, China
248856416@qq.com

Abstract. With its unique challenge and excitement, the stock market has attracted many scholars and investors to study it, especially in the new field of investment transaction and computer technology, and has made a series of achievements. Using the time-series price data of gold and bitcoin, our team builds LSTM time-series price prediction-deterministic strategy gradient dynamic programming model to maximize profit, forecasting the price of gold and bitcoin, and formulates the best dynamic investment strategy. We also evaluate the effectiveness and risk of this model. Considering the impact of transaction costs, we change the cost ratio and analyse the sensitivity of the model. The results show that the model has good robustness and can provide some reference value for investors to make strategies.

Keywords: LSTM · Deterministic Strategy Gradient · Dynamic Programming · Risk Assessment · Sensitivity

1 Introduction

The securities market has always attracted many investors with its high risks and high returns. It is a very complicated nonlinear system, and it is very difficult to predict the stock market. In recent years, with the continuous development of science and technology, the new technology combining investment with computers brings investors more accurate price forecasts more rationally. Up to now, the stock price forecast mainly includes technical analysis method, statistical method, chaotic dynamics, grey forecast, intelligent forecast and combination forecast method. Especially, with the rapid development of artificial intelligence, the intelligent prediction has provided a lot of new ideas for stock market prediction. For example, Pang Sulin and others have established the corresponding BP neural network algorithm prediction model using the actual data of the Shenzhen stock market in our country [1]. The prediction results are good, but their adaptability to unexpected events is poor.

In the early stage, our team collected the time series data of gold and bitcoin prices in the international market [5]. In this paper, aiming at maximizing investors' income, we

N. Radojević et al. (Eds.): ICAID 2022, AHIS 7, pp. 279–289, 2023.
https://doi.org/10.2991/978-94-6463-010-7_29

constructed the LSTM time series price forecast-deterministic strategy gradient dynamic programming model to get the best trading strategy. The time-series data of gold and bitcoin are substituted into the model for experiments to verify the effectiveness and robustness of the model, and the risk assessment and sensitivity analysis of the model is carried out. This model can effectively improve investors' income, and deal with the price fluctuations caused by unexpected situations, which are relatively stable, and has certain constructive significance for securities trading.

2 LSTM Price Forecasting and Strategic Dynamic Planning

2.1 LSTM Price Forecasting Model

2.1.1 Model Building

For the processing of time series data, the most commonly used tool is Recurrent Neural Networks. But for stock prices, it needs to be memorized for a long time, which requires the current results of the RNN [2] to be associated with the results of the first n implicit layers, which will lead to an exponential increase in computation volume and a decrease in model training efficiency. Therefore, we introduce an RNN-based LSTM model which can carry out long-term memory. It adds several types of valves to the RNN structure: forgetting valves, input valves, and output valves. The memory function of the LSTM model is implemented by these valve nodes (Fig. 1).

For the state of the hidden layer c_t, the information of the input of the current layer x_t and the state of the previous implicit layer c_{t-1} is integrated, and the gradient is effectively backpropagated through the short-circuit connection from c_{t-1} to c_t. When f_t closed, the gradient c_t can be passed directly to c_{t-1}, independent of other parameters, which is the key to LSTM's effective mitigation of gradient vanishing.

The opening or closing of a valve is often associated with the sigmoid function.

$$\sigma = \frac{1}{1 + e^{-x}} \tag{1}$$

The forgetting gate f_t is used to control how the state of the previous implicit layer c_{t-1} is preserved:

$$f_t = \sigma\left(W_f x_t + U_f h_{t-1} + b_f\right) \tag{2}$$

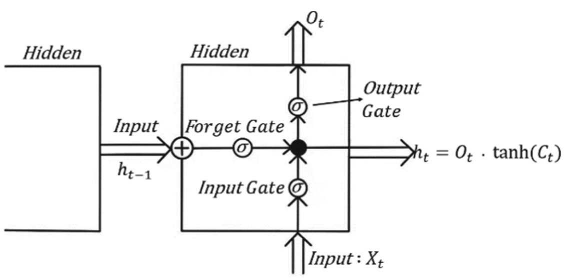


Fig. 1. LSTM unit

Input gates are used to control how new input information is saved to the internal state of the currently implicit layer:

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \tag{3}$$

The output gate controls how the internal state of the currently hidden layer corresponds to the state output to the outside:

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \tag{4}$$

The output and internal state of the t-time LSTM unit are calculated as follows:

$$\hat{c}_t = \tanh(W_c x_t + U_c h + b_c) \tag{5}$$

$$c_t = f_t \circ c_{t-1} + i_t \circ c_{t-1} \tag{6}$$

$$h_t = o_t \circ \tanh(c_t) \tag{7}$$

wherein its initial state,

$$c_0 = 0, h_0 = 0 \tag{8}$$

2.1.2 Model Analysis

Under the framework of the LSTM model structure, we need to first determine the parameters.

Activation function, which defaults to tanh;

Determine the discard rate of each layer of network nodes 0.2, to prevent overfitting;

Determine the calculation method of error, which is the mean squared difference;

The number of layers of the LSTM module does not exceed 3 layers, preventing too many layers and making it difficult to converge (Fig. 2).

In this way, we can put the time series of stock prices into the model for training, get the results and enter them into the normal layer [4]. After training the model, the previous price data is imported into the model as input data to obtain the predicted values. Compared with the actual price of the day, the model can be judged.

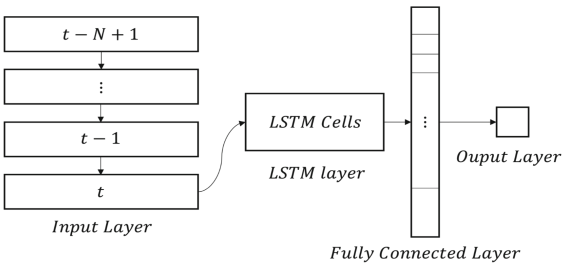


Fig. 2. Neural network layer diagram

2.1.3 Model Training

The LSTM time series price forecasting model is used to forecast the price of gold and bitcoin. We need to use the known data for model training to determine the number of hidden layers and nodes. It is stipulated that the number of nodes in each layer is 128, and the model prediction accuracy of 1, 2, and 3 hidden layers are calculated respectively. It is found that when the number of hidden layers is 2, the accuracy is the highest and the effect is the best. Based on determining that the number of hidden layers is 2, changing the number of nodes to 64,128,256 for optimization shows that the number of nodes 256 is better. Therefore, a long-term and short-term memory network model with a hidden layer (256,256) is established. In the process of model training, we randomly split the price data into training set and verification set by 4:1, and use 50% cross-validation to avoid over-fitting of the model.

2.1.4 Analysis of Results

After training the model, the past price data of gold is imported into the model as input data, and the predicted values of all trading days can be obtained. The prediction results are as follows:

Figure 3 shows the comparison between the gold price forecast and the actual price. On the whole, the price of gold also shows an upward trend year by year, occasionally falling back. The rising trend of gold is gentle, and the overall prediction accuracy is high, which can reflect the fluctuation trend and fit the reality.

2.2 Deterministic Strategy Gradient Algorithm Optimization Model

2.2.1 Model Building

Since the price fluctuates over a continuous period, it can be brought into the framework of deterministic strategy gradient algorithm in reinforcement learning [3].

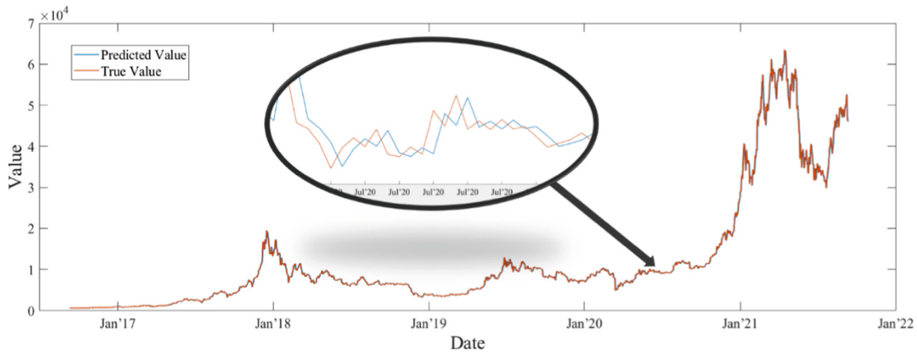


Fig. 3. A comparison chart of gold price forecast and the actual price

Algorithm 1 Deterministic strategy gradient

Input: State s
Number of iterations N
Trading day t
Prediction model Repay r
Output: Tactics $\pi\theta(a|s)$
Random initialization parameters θ;
repeat
  Initialization trajectory τ;
  Initialize the environment and set the initial state $s0$;
  repeat
    In state S_t, calculate strategy $a_t = \pi\theta(a|s_t)$;
    Execute action a_t, get return $r_{t+1}]$ and $s_{t+1}]$;
    Add a_t, S_t and r_{t+1} to track τ;
  until S_t is terminated, $s_t = s_N$
  Get track τ: $\tau = s_0, a_0, r_1, \ldots, s_{N-1}, a_{N-1}, r_N, s_N$;
  for $t = 0$ to N **do**
    Calculate the total return: $G(\tau_1; N)$
    Update policy parameters θ;
    end for
until N

The stock price is updated every day, the environmental state is determined as the trading day, the market price fluctuates, and the movement is bought or sold. The goal of intensive study is to buy and sell before a certain date so that the final cost will be as small as possible and the income will be as large as possible, and finally, the best investment proportion of different stocks in each trading day will be obtained.

2.2.2 Algorithmic Flow

Based on the deterministic strategy gradient algorithm, the highest value dynamic programming model is established. There are three states of the investor's assets: cash, stock G, stock B, of which stock G, stock B are investment objects, so the state defines the amount of storage: define the trader's account cash, stock G, stock B are three-way arrays. Trade cost: The cost of trade is a% of the amount of trade, of which 1% and 2% of stock G and B are assumed, respectively, and there is no cost to hold stocks.

Trade Property Holding Status:

1) Day n Price Stock G, G_n^* Stock B, B_n^*;
2) Day n+1 price Stock G, G_{n+1}^* Stock B, B_{n+1}^*;
3) Day n Holdings: $A_n = [C_n, G_n, B_n]$;
4) The amount of stock G or B is bought per trade: $u_n = [C_n, 0, 0]$.

Establish the state transfer equation:

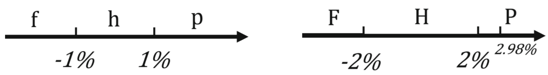
$$A_{n+1} = A_n + \begin{cases} \left[0, \frac{C_n}{aG_n^*}, \frac{C_n}{bG_n^*}\right] & \frac{G^*_{n+1}{}' - G_n^*}{G^*_{n+1}{}'} \geq 1\% \& \frac{B^*_{n+1}{}' - B_n^*}{B^*_{n+1}{}'} \geq 2\% \\ \left[0, \frac{C_n}{aG_n^*}, 0\right] & \frac{G^*_{n+1}{}' - G_n^*}{G^*_{n+1}{}'} \geq 1\% \& \frac{B^*_{n+1}{}' - B_n^*}{B^*_{n+1}{}'} < 2\% \\ \left[0, 0, \frac{C_n}{bG_n^*}\right] & \frac{G^*_{n+1}{}' - G_n^*}{G^*_{n+1}{}'} < \% \& \frac{B^*_{n+1}{}' - B_n^*}{B^*_{n+1}{}'} \geq 2\% \\ [0, 0, 0] & \frac{G^*_{n+1}{}' - G_n^*}{G^*_{n+1}{}'} < 1\% \& \frac{B^*_{n+1}{}' - B_n^*}{B^*_{n+1}{}'} < 2\% \\ \vdots & \vdots \end{cases}$$

Target Return Indicator: $f_n = C_n + G_n^* * G_n + B_n^* * B_n$.

Solve with the goal of the maximum target return indicator, bring in the forecast price of stocks G and B the next day, as well as the actual price, to get the highest account holding value at the end of the deadline.

2.2.3 Model Training

Then, based on the deterministic strategy gradient algorithm, we build the highest value, dynamic programming model. Plan daily transactions based on the results of the forecast, determine the daily [cash, gold, bitcoin] portfolio, and thus calculate the return after several trades. The growth rate interval is as follows:



2.2.4 Analysis of Results

After 15 trades, we finally got our forecast value, which increased the value by 4141.125%, far exceeding the value increase without a trading strategy (Table 1 and Fig. 4).

Table 1. Purchase Strategy

Group Number	Predict Change	Purchase Strategy		
		C	G	B
S1	fF	hold	sell	Sell
S2	fH	hold	sell	hold
S3	fP	buyB	sell	hold
S4	hF	hold	hold	sell

(*continued*)

Table 1. (*continued*)

Group Number	Predict Change		Purchase Strategy		
			C	G	B
S5	hH		hold	hold	hold
S6	hP	P > 2.98%	buyB	sell	hold
S7		P <= 2.98%	buyB	hold	hold
S8	Pf		buyG	hold	sell
S9	pH		buyG	hold	hold
S10	pP	0 <= (p-P) <= 2.98%	buyG	hold	hold
S11		0 <= (p-P) <= 2.98%	buyB	hold	hold
S12		(p-P) > 2.98%	buyG	hold	sell
S13		(p-P) > 2.98%	buyB	sell	hold

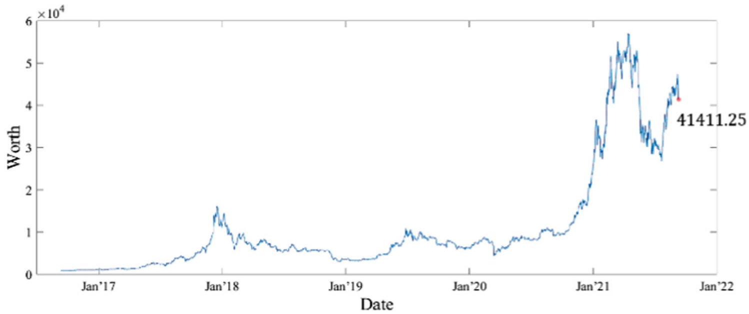


Fig. 4. Profit value of our strategic

3 Risk Assessment

It should not be overlooked that there are risks in the forecasts and strategic planning we make, and we cannot determine how the risks will affect market prices. For example, the new crown pneumonia epidemic in early 2020 has caused a sustained blow to the US stock market. Of course, there is no shortage of venture capitalists in the financial sector who take on high risks to obtain high returns, and low risk means that stocks have a strong ability to reduce non-systemic risks, and the returns must decline.

So its survivability in the face of sudden risks and dangers, as well as its return at higher risks, are significant factors in evaluating the quality of the model, which is risk assessment.

VaR indicates that investors may lose a certain level of confidence. It is an asymmetrical measure of risk that in a sense represents the level at which a portfolio would not lose more than at a certain probability. If X is a random variable and $\alpha \in [0, 1]$ is a

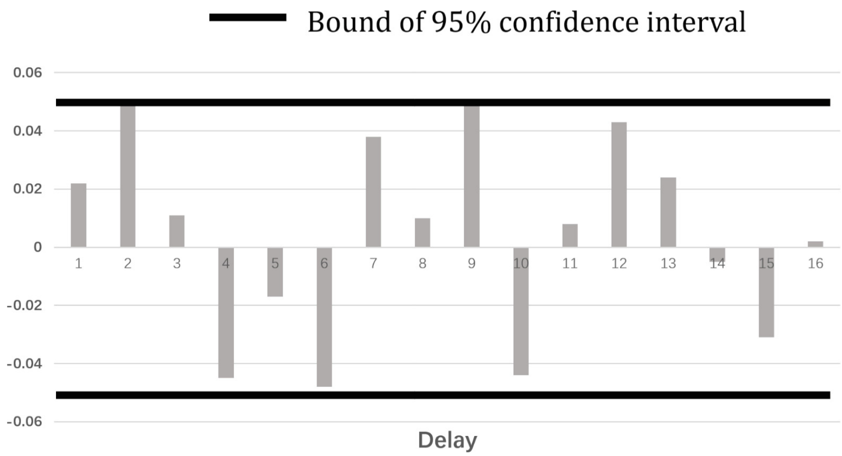


Fig. 5. Random risk

confidence level, then the VaR of the random risk X can also be defined as the quantile of X.

$$VaR_{\alpha}(X, T) = -\inf\{\tau | P(X \leq \tau) \geq 1 - \alpha\} \tag{9}$$

where τ is the predicted target value.

We specify the confidence level as $\alpha = 1\%$ or $\alpha = 5\%$, accordingly calculate the VaR value at a certain point in the forecast data. First, it is assumed that the data are normalized and conform to the standard normal distribution on a specific interval. In particular, in the risk assessment of the model, we should choose the predicted price of a certain period with large fluctuations, without a clear linear relationship, to prove that in the face of risk, our model is more robust and will not be easily broken (Fig. 5).

4 Sensitivity Analysis

4.1 Model Updates

In the process of formulating a quantitative investment strategy, the impact of transaction costs on the strategy cannot be ignored. The cost of each transaction is a% of the trade amount, on this basis, consider the sensitivity of changing the transaction cost, explore the impact of transaction cost on the price prediction result and trading strategy, and conduct sensitivity analysis [6].

Table 2. Combination of cost

Programme	1	2	3	4
$a_G\%$	0.5	1	1.5	2
$a_B\%$	1	2	3	4

Changes in transaction costs affect the final rate of return and investment strategy, so the model changes as follows:

$$f_{n+1} = \begin{cases} f_n + \left(\frac{G^*_{n+1}-G^*_n}{G^*_{n+1}} - a_G\%\right) * G^*_n * G_n + \left(\frac{B^*_{n+1}-B^*_n}{B^*_{n+1}} - a_B\%\right) * B^*_n * B_n & pP \\ f_n + \left(\frac{G^*_{n+1}-G^*_n}{G^*_{n+1}} - a_G\%\right) * G^*_n * G_n + \left(\frac{B^*_{n+1}-B^*_n}{B^*_{n+1}}\right) * B^*_{n+1} * B_n & pF \\ f_n + \left(\frac{G^*_{n+1}-G^*_n}{G^*_{n+1}}\right) * G^*_{n+1} * G_n + \left(\frac{B^*_{n+1}-B^*_n}{B^*_{n+1}} - a_B\%\right) * B^*_n * B_n & fP \\ f_n + \left(\frac{G^*_{n+1}-G^*_n}{G^*_{n+1}}\right) * G^*_{n+1} * G_n + \left(\frac{B^*_{n+1}-B^*_n}{B^*_{n+1}}\right) * B^*_{n+1} * B_n & fF \\ \vdots & \vdots \end{cases}$$

4.2 Experimental Design

Among them, the transaction costs of gold and Bitcoin are respectively, and in the process of sensitivity analysis, the transaction cost values are taken separately as shown in Table 2.

A total of 4 sets of samples were taken for sensitivity analysis, and the results of the model operation were as follows.

4.3 Analysis of Results

When changing the cost of each trade (purchase or sale), i.e. the value of the trade amount a%, different results are shown in Fig. 6.

It is found that in the overall trend, with the change in trade cost, the final account value gradually decreases with the increase of trade cost, and the growth curve of the overall account value has no obvious change with the commodity prices of gold and bitcoin, and the overall commodity trade is conducted with the commodity prices of gold and bitcoin under the trading strategy (Table 3).

The resulting analysis of its trade costs shows that the value of a% is increased, and the number of transactions is decreasing, which can be intuitively obtained from the above numbers and visual curves. Combined with investment transaction decisions, when the transaction cost continues to increase, the number of transactions shows a downward trend under the laws of nature.

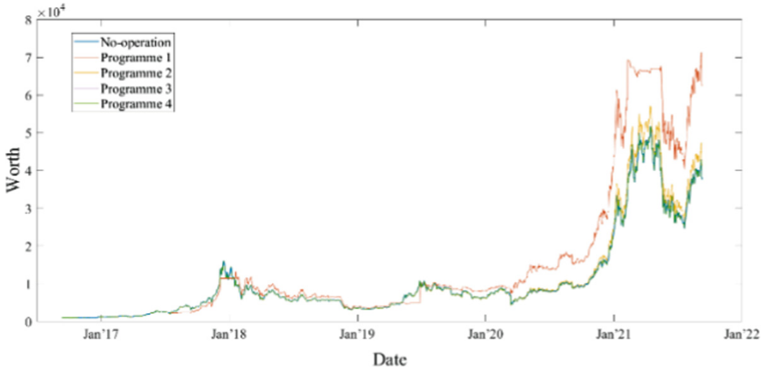


Fig. 6. Visualization chart of the account value curve

Table 3. Number of Transactions

Programme	Number of Transactions		
	Gold	Bitcoin	Total
1	26	10	35
2	14	2	15
3	9	1	9
4	5	1	5

5 Conclusions

The reason why we can look far is that we stand on the shoulders of giants. We have fully studied the research results of previous scholars, integrated and improved them, and constructed the LSTM-deterministic strategy gradient highest value dynamic programming joint model, which not only reasonably predicts the trend of stock prices, but also provides investors with good investment advice so that they can obtain greater benefits. We used the actual situation of gold and bitcoin to prove that the model has high adaptability and robustness, verified the impact of changes in transaction costs on transactions through sensitivity tests, and then dynamically adjusted our strategy.

Acknowledgements. The experiment was strongly supported by the Key Laboratory of Information Systems Engineering of the National University of Defense Technology, as well as the strong support of Professor Chen.

References

1. Chen J (2019) Research on financial series prediction based on the hybrid neural network. Hunan University
2. Meng X (2013) Research on futures price trend prediction and quantitative trading strategy based on machine learning. Wuhan University of Science and Technology
3. Sulin P (2006) Empirical comparison between BP algorithm and symmetric ARCH model in forecasting stock market volatility. Control Theor Appl 04:658–662
4. Qian L, Cheng J (2021) Comparative study of different stock risk measurement methods. J Huangshan Univ 23(1):48–52
5. The Problem C of MCM/ICM (2022). http://www.mathmodels.org/Problems/2022/MCM-C/index.html. Accessed 18 Feb 2022
6. Wang M (2021) Research on futures price trend prediction and quantitative trading strategy based on machine learning. Nanjing University of Finance and Economics