Investment Strategy Based on LSTM Network and PSO Model

Kunqi Han[1], Wei Zhang[2](✉), and Yuzi Zhang[3]

[1] College of Energy and Electrical Engineering, Hohai University, 8th Focheng West Road, Nanjing, People's Republic of China
hhuhkq@hhu.edu.cn
[2] Business School of Hohai University, 8th Focheng West Road, Nanjing, People's Republic of China
1907020220@hhu.edu.cn
[3] School of Public, Administration of Hohai University, 8th Focheng West Road, Nanjing, People's Republic of China

Abstract. With the rapid development of economy, many people are keen to buy and sell unstable financial products to maximize their interests. This paper proposes an investment strategy based on Sharp ratio and neural network particle swarm optimization, which is able to predict the best time to buy, hold and sell various financial products through artificial intelligence based on the price flow data of the products over the past period of time. Taking cash, gold and bitcoin as examples, this paper conducts empirical research on the algorithm and obtains the final profit of the optimal investment scheme. Then, through the sensitivity analysis, we found that as the transaction fee increases, the number of transactions of gold and Bitcoin decreases significantly, and the value decreases. On the contrary, there is the same theory, which proves that our model is very good. However, the model proposed in this paper still has some shortcomings. In summary, although the model proposed in this paper has some shortcomings, its accuracy and stability are enough to solve this problem, so it can be explained again the accuracy of the model proposed in this paper.

Keywords: Investment Strategy · LSTM Network · Sharpe Rate · Particle Swarm Optimization

1 Introduction

In recent years, people's savings have been increasing, and more and more people are choosing to invest in one or more financial products with high or low risk and return. In order to maximize returns under a certain risk, it is necessary to build an efficient investment portfolio.

Linear forecasting models include Autoregressive Moving Average (ARIMA), Generalized Autoregressive Conditional Heteroskedasticity Model (GARCH), Exponential Smoothing Model, etc. The application of the above time series models has made great progress in financial market forecasting research [2].

N. Radojević et al. (Eds.): ICAID 2022, AHIS 7, pp. 330–339, 2023.
https://doi.org/10.2991/978-94-6463-010-7_34

However, given the uncertainty and high noise characteristics of financial time series, accurate forecasting is still very difficult, and the relationship between independent variables and dependent variables usually changes dynamically over time, making it difficult for traditional time series models to effectively conduct financial markets prediction. In addition, it is only suitable for stationary series modelling, which greatly limits the application and expansion of time series models [4].

We found that in addition to cash, bitcoin and gold are two representative investment products. Bitcoin represents a high-reward, high-risk product, while gold is relatively low-reward and risky [3].

For the above background, we will start at $1,000 and use the five-year trading period from November 9, 2016 to October 9, 2021 as the data source. Build a prediction model based on LSTM network, Sharpe ratio and particle swarm optimization to find the best investment strategy for two investment products, Bitcoin and Gold [1].

2 Method

2.1 Research Method

2.1.1 LSTM Neural Network

The LSTM neural network structure used in this paper is a variant structure of the species RNN proposed by Sepp Hochreiter and Jurgen Schmidhuber in 1997. As a variant of the RNN neural network, the biggest difference from the LSTM NN network is that it draws on the selective input and selective forgetting mechanism of the human brain, introducing three "gate" structures of forgetting gate, input gate and output gate, and a memory unit to selectively receive information from the afferent neural network.

In them, the "gate" structure belonging to the logical unit is only responsible for completing the setting of weights at the edge of the neural network connected to the memory unit, without impact on other neuron nodes. The LSTM neural network structure is shown in Fig. 1.

2.2 Data Sources

We will use a five-year trading period from November 9, 2016 to October 9, 2021 of gold and bitcoin. On each trading day, the trader will have a portfolio consisting of cash,

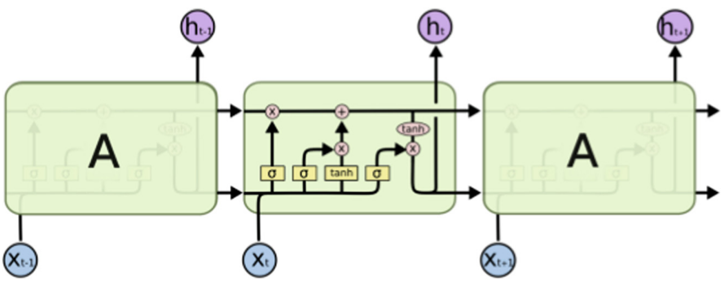

Fig. 1. Structural diagram of the LSTM network.

gold and bitcoin [C, G, B], and the units are Dollars, Troy Ounce and Bitcoin. The initial state is [1000, 0, 0]. While the commission cost per transaction (buy and sell) is α% of the transaction amount. Assume Alpha Gold = 1% and Alpha Bitcoin = 2%. There is no cost to hold the asset.

Furthermore, we assume: a. The data alignment process starts on September 11, 2016 and ends on September 10, 2021 (non-trading days help). Fill Missing Values Using Linear Interpolation Algorithm. b. Assume that cash held will increase at a fixed annual rate of 2%.

3 Proposed Approach

3.1 Symbol Description

See Table 1.

3.2 Prediction Model Based on LSTM Network

3.2.1 Basic Architecture

In the network structure of LSTM, the forgetting door is responsible for receiving the previous moment information to the current point, which determines the previous memory unit state can be retained to the current moment, the purpose of the logical unit is to forget the useless information, forgetting door according to the output of the previous moment and the input of the information retention degree:

$$C_{t-1}, C_t, h_{t-1}, x_t, f_t$$
$$f_t = \mathrm{sigmoid}\left(W_f\left[h_{t-1}, x_t\right] + b_f\right) \tag{1}$$

After the action of the sigmoid activation function, the ft. The values range between 0 and 1, the closer f_t is to 1, the more information it remains. After retaining the previous

Table 1. Symbol description.

Symbol	Symbolic meaning
i_t	input gate
f_t	Forgotten Gate
O_t	output gate
E_1	Cash, Gold, Bitcoin as % of Total Assets
$[c_k, g_k, b_k]$	the weight of inertia
$[\Delta G_k, \Delta B_k]$	The amount of change in the proportion of gold and the proportion of bitcoin
$[0.00547\%, \overline{G_k}, \overline{B_k}]$	Cash, gold, bitcoin yield on day k
o_t	level of information output

information, the neural network also needs input x_t from the current moment t. New information is generated in t to update the memory unit status c_t.

The effect of the input gate is performed according to h_t-*1* and x_t. to determine which information can be added to the previous moment of memory unit state c_{t-1} and updated to the new memory unit state c_t. First, the input door is based on the output h_{t-1} of the previous moment t-*1*. And the input x for the current moment t, figure out it to be:

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

In formula, $\mathrm{W_i}$ multiply the hidden layer output h_{t-1} at the previous moment $t-1$ and x_t at the current moment t. The parameter matrix of b_g, represents a bias item.

Then z_t is calculated as:

$$\mathrm{z_t = tanh\left(W_g\left[h_{t-1}, x_t\right] + b_g\right)} \tag{3}$$

Updated current moment memory unit status c_t for:

$$\mathrm{c_t = f_t \cdot c_{t-1} + i_t \cdot Z_t} \tag{4}$$

The function of the output gate is to determine how much information at the current moment can be output, according to the output h of the previous moment $t-1$, and the input x_t for the current moment t.

Computing the information output degree o_t as:

$$\mathrm{o_t = sigmoid\left(W_o\left[h_{t-1}, x_t\right] + b_0\right)} \tag{5}$$

The final output is the output h_t at the current moment t:

$$\mathrm{h_t = o_t \cdot tanh(c_t)} \tag{6}$$

At this point, the forward propagation of the circulation body in the LSTM network ends, followed by the reverse propagation of error, and then corrects the weight coefficients in the process of error back transmission. The core idea is still to find the partial derivative of the parameters in each gate unit, and more weights according to the direction of gradient convergence.

3.2.2 Strategies for Optimization Algorithms Normalization

In the process of processing data, due to the large difference between the size range and dimension of the data itself, the non-standard data will lead to longer training time and worse training effect of the model. Therefore, we need to the data is normalized.

In this paper, we choose the more common maximum and minimum standardization. In the data pre-processing stage, we map the value of the data to [0, 1], and after the prediction result is obtained, the obtained prediction value is reversed. Normalized.

Max Min Normalized Mapping Function:

$$\mathrm{y_i = \frac{x_i - \min\limits_{1\le j\le n}\left\{x_j\right\}}{\max\limits_{1\le j\le n}\left\{x_j\right\} - \min\limits_{1\le j\le n}\left\{x_j\right\}}} \tag{7}$$

Since the prices of gold and bitcoin are updated every day, we can add the latest prices of gold and bitcoin into our training set every day, and under this, our training set has been expanded, thus Ability to better predict future gold and bitcoin prices.

3.3 Policy Model Based on Sharpe Rate and Particle Swarm Optimization

3.3.1 Sharpe Ratio

In order to facilitate the calculation of the planning model in the future, we assume that the proportion of cash, gold, and Bitcoin held in the total assets on the kth day is recorded as $\left[c_k, g_k, b_k\right]$, So we have the relation

$$c_k + g_k + b_k = 1 \tag{8}$$

We make the decision variable a two-tuple $[\Delta G_k, \Delta B_k]$ are the changes in the proportion of gold and the proportion of bitcoin, respectively.

Suppose, the increase of cash, gold, and bitcoin on day k are respectively $\left[0.00547\%, \overline{G_k}, \overline{B_k}\right]$. According to the prediction result, on the kth day, we will have the data of the kth day and the prediction result of the kth day. In order to better quantify the relationship between daily investment risk and return, we introduce the Sharpe ratio. Its mathematical expression is:

$$SR_p = \frac{E\left(r_p\right) - r_f}{\sigma_p} \tag{9}$$

$$E\left(r_p\right) = c_k \cdot E(r_c) + g_k \cdot E\left(r_g\right) + b_k \cdot E(r_b) \tag{10}$$

$$\delta_p = \sqrt{g_k \delta_g^2 + b_k \delta_b^2 + 2 g_k b_k \mathrm{cov}(G, B)} \tag{11}$$

In the formula, $E\left(r_p\right), \sigma_p, r_f$ are the expected return, the standard deviation of the return, and the return of the risk-free asset during the observation period, respectively.

According to the planning model, on day, we set the objective function to $f(k)$. It is set to the value of the Sharpe ratio of the third day when the position is adjusted today and the next three days remain unchanged. When this objective function reaches the maximum, it is the state when the risk and return are balanced.

Taking the end of day k as an example, the proportion of cash, gold, and bitcoin in total assets is $\left[c_k, g_k, b_k\right]$.

At the end of each day, that is, the end of the k-th day, the ups and downs need to be settled first.

If the increase of cash, gold and bitcoin on that day equals $\left[0.00547\%, \overline{G_k}, \overline{B_k}\right]$.

Then, the proportion will become:

$$\left[(1 + 0.00547\%)c_k, \left(1 + \overline{G_k}\right)g_k, \left(1 + \overline{B_k}\right)b_k\right]$$

It can be seen that the sum of the proportions here may no longer add up to 100%, which is due to changes in the total holding amount. After we have completed all changes, we will re-normalize in order to perform the kth + 1 day for repeated actions.

Then, trade to adjust the position and settle the transaction fee.

When only the purchase of gold is required, record the change in the proportion of gold as deltaG_k $>$ 0. At this time, the change in the proportion of cash is 1.0101* deltaG_k (this is because the buying rate is 1%, 101 blocks Can't buy 100 pieces of gold). When only the transaction of selling gold is required, record the change of the gold amount as deltaG_k < 0, at this time, the change of the cash proportion is 0.99* deltaG_k. When only the transaction of buying Bitcoin is required, record the change in the proportion of Bitcoin amount as deltaB_k > 0, at this time, the change in the proportion of cash is 1.0204*deltaB_k. When only the transaction of selling Bitcoin is required, note that the change in the proportion of Bitcoin amount is deltaG_k < 0, at this time, the change in the proportion of cash is 0.98*deltaG_k.

After the transaction fee settlement is completed, the final day's cash, gold, and bitcoin will account for the proportion of total assets as:

$$\begin{bmatrix} (1 + 0.00547\%)c_k + (1 \pm 0.01)\Delta G_k + (1 \pm 0.02)\Delta B_k, \\ \left(1 + \overline{G_k}\right)g_k - \Delta G_k, \left(1 + \overline{B_k}\right)b_k - \Delta B_k \end{bmatrix}$$

Then, we normalize it, and multiply the sum of the three proportions by the total amount of yesterday, to get today's total amount and rate of return, as well as today's proportion.

According to our assumption, for the function f_k, The amounts will remain the same for the next three days, therefore, in the following process ΔG_{k+1} and ΔB_{k+1}. Directly equal to 0, until the third day, and then calculate the Sharpe rate of the third day. Constraints: After all operations are performed on the day, the proportion of cash, gold, and Bitcoin in total assets should be greater than or equal to 0, that is,

3.3.2 Particle Swarm Optimization

To further obtain the optimal value of each variable, the particle group optimization (PSO) is introduced to solve it.

First, this paper initializes the particle population so that the population size is k, then the particle population can be expressed as:

$$G = \left\{g_1, g_2, g_3 \ldots g_k\right\} \tag{12}$$

Each particle in the particle group has the dimensions in n directions, along with the corresponding coordinates and the instantaneous velocity in each dimension. When the current number of iterations is t, the position and speed of each particle are recorded as:

$$\begin{aligned} x_i^t &= \left(x_{i\_1}^t, x_{i\_2}^t, x_{i\_3}^t \cdots x_{i\_n}^t\right) \\ v_i^t &= \left(v_{i\_1}^t, v_{i\_2}^t, v_{i\_3}^t \cdots v_{i\_n}^t\right) \end{aligned} \tag{13}$$

The second value of the 1 i k target corresponds to a certain dimension of the particle. Notably, the position and velocity of the particles in different dimensions need to be limited, i.e:

$$\begin{aligned} x_{i\_j}^t &\in \left[\text{mindis}(j), \text{maxdis}(j)\right] \\ v_{i\_j}^t &\in \left[\text{minvel}(j), \text{maxvel}(j)\right] \end{aligned}$$

Among them, $1 \leq i \leq k$, $1 \leq j \leq n$. In order to calculate the fitness value of the particle, this paper needs to introduce an adaptation function with the particle position as the independent variable, recorded as:

$$\mathrm{fit\_value} = \mathrm{fit}\left(x_i^t\right) \tag{14}$$

Where the fit_value is the fit value. When particles iterate, this paper needs to update the optimal position of individual particles and the optimal position of the particle group, recorded as:

$$x_{i\_best} = \mathrm{fit}^{-1}\left(\max\ \mathrm{fit}\left(x_i^r\right)\right) \tag{15}$$

$$x_{best} = \mathrm{fit}^{-1}\left(\max\ \mathrm{fit}\left(x_{i\_best}\right)\right) \tag{16}$$

Where 0 r t, that is, the best position of the particle from the number of iterations as the optimal position of the single particle, and the best one from the optimal position of each particle as the optimal position of the whole population.

During each round of iteration, the position and speed of the particles need to be updated. The update of the particle velocity is based on three factors: the current velocity of the particle, the individual optimal position of the particle and the group optimal position of the particle group. The reasons are listed as follows:

1) The particle has inertia, so the current velocity of the particle must affect its next velocity;
2) The particle has the self-cognitive ability, specifically manifested as: the particle tends to advance toward the direction of its own optimal position;
3) Particles have social cognitive ability, specifically manifested as: particles tend to move toward the optimal position of the group.

Based on the above discussion, the particle velocity and position:

$$v_i^{t+1} = \alpha v_i^t + \beta r_1\left(x_i^t - x_{i_{best}}\right) + \gamma r_2\left(x_i^t - x_{best}\right) \tag{17}$$

$$x_i^{t+1} = x_i^t + v_i^{t+1} \tag{18}$$

Where, α represents the weight of inertia, β the self-learning factor, and γ the group information transfer factor r1 and, r2 represents the random number within the [0, 1] range, used to increase the randomness of the search. It should be noted that the position and velocity of the particle are the vector, and the formula design follows the principle of vector operation.

4 Results

As for the prediction model based on LSTM network, we bring in the data, add the latest gold and bitcoin prices to the training set every day for continuous training, record the predicted values, and the fitting results are shown in Figs. 2 and 3.

Further, According to PSO, when the Sharpe rate on the third day is used as the objective function to take the maximum value, the result is that the starting $1,000 can end up with $128,675.

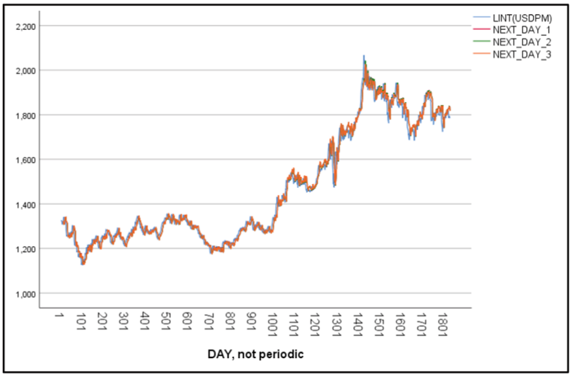

Fig. 2. Gold price prediction renderings.

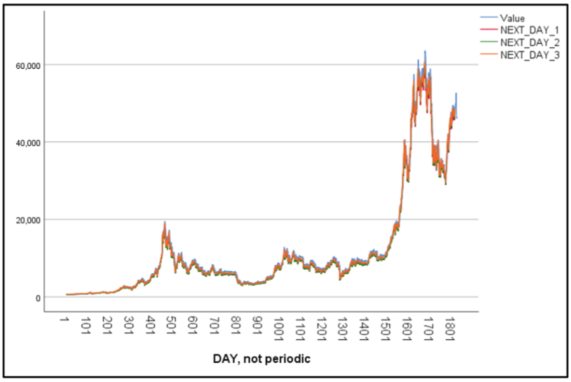

Fig. 3. Bitcoin price prediction renderings.

5 Discussions

5.1 Sensitivity Analysis

As shown in Table 2, a_gold and a_bit represent the transaction fee ratio of gold and Bitcoin, respectively, and returns represent the final benefit. We can see that as the transaction fee increases, the number of transactions between gold and Bitcoin decreases significantly, and the final value increases. Decrease, as the fee becomes lower, the number of transactions of gold and bitcoin increases significantly, and the final value rises.

5.2 Sensitivity of the Strategy to Transaction Costs

In order to prove that the solution we choose is the local optimal solution, we can consider giving the result a certain disturbance, and explain that the result after disturbance is no longer the maximum value of the objective function. Specifically, for our above

Table 2. The transaction fee ratio of gold and Bitcoin.

Item	a_gold	a_bit	returns
Data	0.01	0.02	128675
	0.013	0.02	78214.4
	0.007	0.02	172153.1
	0.01	0.017	458137.1
	0.01	0.023	21249.6
	0.007	0.017	1368392
	0.013	0.023	16653.4

Table 3. Proof of the best solution.

sharpen	0.01	0.02	0.03
OUR_RESULT	4.8467	4.8467	4.8467
MORE_GOLD	4.4464	4.2836	4.3151
MORE_BITCOIN	4.3358	3.9132	3.7451
ALL_MORE	3.969	4.1634	3.85
LESS_GOLD	4.5724	4.3219	4.1226
LESS_BITCOIN	4.3386	4.5859	4.6124
ALL_LESS	4.6137	4.5984	4.5626

gold and bitcoin. Add the fluctuations in the range of 1%–3% to the buying and selling plan, and re-plan the buying and selling plan. The results of multiple tests are shown in Table 3.

6 Conclusions

Combining the application of PSO and Sharpe ratio, this paper constructs a deep LSTM neural network and applies it to the forecast analysis of the gold and bitcoin markets after September 10, 2021. We found the optimal solution by simulating the optimization process through particle swarm optimization. We take the Sharpe rate on the third day as the objective function, take the maximum value, and finally find that the original $1,000 can eventually become $128,675. Through a sensitivity analysis, we found that as transaction fees increased, gold and Bitcoin significantly decreased in transaction volume and decreased in value. Instead, there is the same theory, which proves that our model is very good. However, the model proposed in this paper still has some shortcomings.

Due to space limitations, this study only tested two mainstream financial products, gold and bitcoin. It is the next research direction to further increase the number of samples, expand the test scope, and establish an investment forecasting model under the influence of multiple factors on the basis of this model.

References

1. Huang Y, He X, Lu L, Lin Z (2022) Application of improved particle swarm optimization in inventory forecasting. Light Ind Mach 02:103–108
2. Hui X, Liu H, Hu W, He D (2003) RMB exchange rate forecast based on time series GARCH model. Financ Res 05:99–105
3. Wu W, Qiu Y, Zhang L (2015) The effectiveness of Chinese household investment portfolio: a study based on sharpe ratio. World Econ 01:154–172
4. Yang Q, Wang C (2019) Research on global stock index prediction based on deep learning LSTM neural network. Stat Res 03:65–77