Problem Chosen	2022	Team Control Number
C	MCM/ICM Summary Sheet	**2218530**

Coping with the market: Trading Strategy Based on LSTM Time Series Prediction and Linear Programming

Summary

We have witnessed the rapid growth of quantitative investment and there are many state-of-the-art techniques to help traders make the best strategy. Market traders buy and sell volatile assets frequently, with a goal to maximize their total return. There is usually a commission for each purchase and sale, while there is usually no cost to hold an asset. Two such assets are gold and bitcoin. In this paper, we developed a method to predict the future prices of gold and bitcoin so that a optimum trading strategy could be derived to maximize the final asset.

We've established 2 models: *Price Prediction Model Based on LSTM-CNN* and *Trading Model Inspired by Dynamic Programming*.

Model(1): For price prediction, considering that the streams of daily prices of gold and bitcoin are typical time series, we adopted a deep learning model featuring a joint "LSTM-CNN" architecture to achieve high forecast accuracy. In order to reduce the influence of noise, we preprocessed the training data by smoothing the price curves with a rectangular window to achieve a better result and help training loss converge faster at the same time. We also took a suitable dynamic training mechanism to obtain the up-to-date model to ensure a decent prediction accuracy. Meanwhile, the training cost was also controlled within a reasonable and endurable range. The prediction results are shown in section 3.4.

Model(2): As for the trading method, we were inspired by dynamic programming to develop an algorithm which determined the next trading day and maintained optimum overall asset on that day. To find trading days, we looked for maximum and minimum prices in predicted price curves, which was more insightful, more guaranteed and more time-saving than running models everyday. Then, we optimized future assets with linear programming, which gave the best trading strategy for the current day according to the optimal principle of dynamic programming. Our model reached a profit rate over 7000%, which is 139.4% annualized rate of return. Trading history and the final result of assets are shown in section 5.

In addition, sensitivity analysis showed the total asset decreased roughly linearly as the commission rate multiplies. For example, when the commission rate was set to twice of its original value, the total asset shrunk to about 97.48% of its original value.So our approach is robust to noises and commission rate changes. The sensitivity analysis is shown in figure 9. However, our rather greedy and risky approach requires high prediction accuracy rapid training of neural networks, so there's still much improvement to make in the future.

Keywords: Quantitative Transaction; Asset Configuration; Time Series Forecast; LSTM ; CNN; Dynamic Programming; Linear Programming

Contents

1 Introduction

1.1 Background

Should you invest in Bitcoins? Bill Gates said yes but only if you are Elon Musk. Goldman Sachs(GS) said in a report that this February bitcoin could be more than double within the next five years, while 74% of the professional investors surveyed by Bank of America saw bitcoin as a bubble last year. No matter what, historical price trend convinces us that investing in bitcoin and gold is promising.

Quantitative transaction is very popular since its birth because investment conducted by computer is objective and rational, and risks are effectively controlled. In recent years, it combines all sorts of time series prediction ranging from auto regression to deep learning, proposing multifarious investment strategies to help us maximize our assets in real-world markets.

1.2 Problem Restatement

This problem considers a trader who will start with $1000 on 9/11/2016. On each trading day i, the trader has a portfolio consisting of cash, gold, and bitcoin, noted as $[C_i, G_i, B_i]$ in U.S. dollars, troy ounces, and bitcoins, respectively. The initial state is [1000, 0, 0] on 9/11/2016. The commission rate for each transaction (purchase or sale) is $\alpha_{gold} = 1\%$ and $\alpha_{bitcoin} = 2\%$ of the amount traded. The problem is to help the trader develop a method which uses only the past stream of daily prices to determine each day if the trader should buy, hold, or sell his assets in the portfolio.

Specifically, we need to:

- Develop a that gives the **best** daily trading strategy during a five-year trading period till 9/10/2021.
- Try to prove our model provides the best strategy
- Analyse how sensitive our strategy is to transaction costs
- Communicate our strategy, model, and results to the trader

To simply the problem, we reasonably assume that the trader cannot lend cash or any other assets on each day, which means that the trader's cash on every should to be non-negative, and if the he/she attempts to make a sell, the amount of gold or bitcoin he/she sells should be no more than the amount he/she currently possesses.

1.3 Modeling framework

Our solution to the problem can be divided into two parts:

1. Price prediction with a deep learning model featuring a joint LSTM-CNN architecture
2. Trading strategies algorithm inspired by dynamic programming

Our workflow is illustrated in figure 1. Before we start trading, we wait for a particular length of time to observe gold and bitcoin prices since we need sufficient data to train the prediction

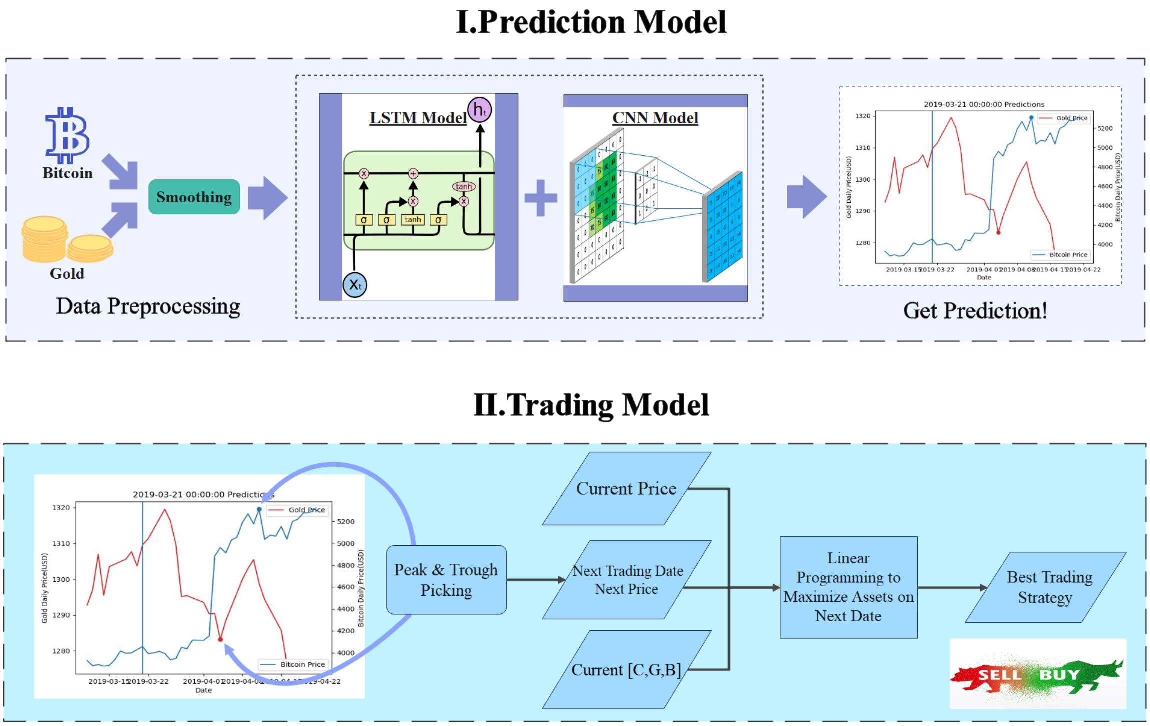


Figure 1: Modeling framework

model. When enough training data is collected, we train the deep learning prediction model with a joint *LSTM-CCN* architecture and use the model to predict gold and bitcoin prices in the future. Before we put training data into the model, we smooth the price curve since there is noise on it, especially for bitcoin. After we attain future price curves, we pick their peaks and troughs future days as potential future trading dates. Then we use linear programming maximize total assets on each candidate trading day and find out the day with the biggest asset along with the corresponding trading strategy for today. We complete today's trading based on the derived strategy and jump to the previously picked corresponding trading day. We repeat this procedure till the end of the trading period.

2 Assumptions & Notations

2.1 Assumptions

As mentioned in the problem restatement above, the assumptions can be formalized as:

1. The amount of cash on each day in the 5-year trading period is non-negative, i.e.
$$C_{d_i} \geq 0 \quad \forall d_i \in \mathbf{D} \tag{1}$$
2. The amount of gold and bitcoin sold each trading day is no more than the amount of gold and bitcoin in the current asset, i.e.
$$O^G_{d_i^*} \geq -G_{d_i^*} \quad \forall d_i^* \in \mathbf{D}^* \tag{2}$$
$$O^B_{d_i^*} \geq -B_{d_i^*} \quad \forall d_i^* \in \mathbf{D}^* \tag{3}$$
3. The gold and bitcoin prices are considered known before the trading decision is made on every day during the trading period, i.e. P_i^G and P_i^B can be treated as given conditions on day i, $\forall d_i \in \mathbf{D}$.

2.2 Notations

The main notations we defined during our modeling are shown in Table 1:

Table 1: Notations

Symbol	Description
d_i	The ith day
d_i^*	The ith trading day
$\mathbf{D}$	The set of all days in the trading periods
$\mathbf{D}^*$	The set of all trading days
A_i	Total assets on day i
C_i	Amount of cash on day i
G_i	Amount of gold on day i
B_i	Amount of bitcoin on day i
P_i^G	Gold price on day i
P_i^B	Bitcoin price on day i
O_i^G	Amount of gold purchased(+) or sold(-) on day i
O_i^B	Amount of bitcoin purchased(+) or sold(-) on day i
α_{gold}	Commission rate of gold transaction
$\alpha_{bitcoin}$	Commission rate of bitcoin transaction

3 Price Prediction Model

Prediction is the most essential part in the problem pipeline. Our price prediction model features a **joint LSTM + CNN architecture** which is expected to perform well on time series forecasts according to previous research. Here, we propose a method to predict future prices of gold and bitcoin using previous observations as training data. Our prediction model involves three parts: *data preprocessing*, *LSTM+CNN model* and *dynamic training mechanism*.

3.1 Data preprocessing

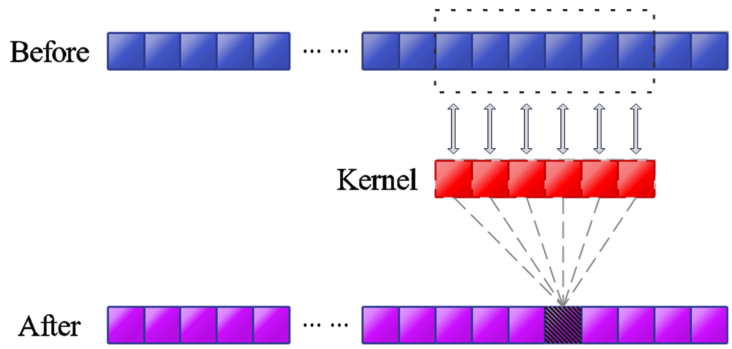
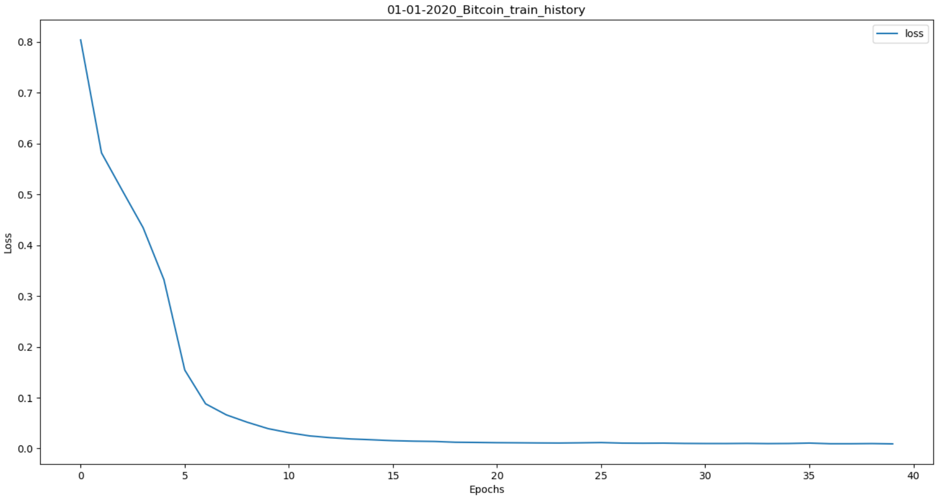
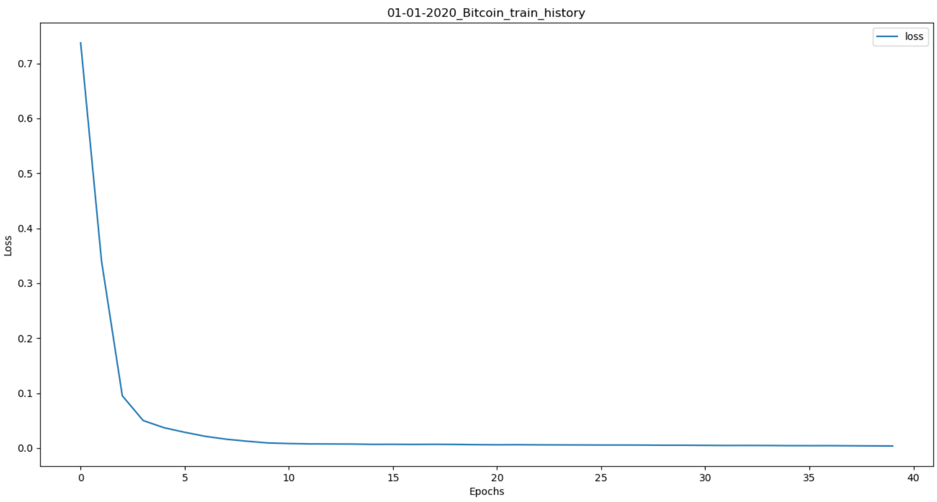


Figure 2: smooth process

If we look at the price curves, it's not difficult to spot noise. Moreover, the convergence speed of training loss when using raw price data (as shown in figure 3(a)) is slower than that if the data is smoothed with a rectangular window before it's put into the neural network (as shown in figure 3(b)). Therefore, we use a convolution window to preprocess price data (shown in figure 2) and the epoch number used in model training can be reduced.



(a) Before smooth



(b) After smooth

Figure 3: Contrast regarding smooth method

3.2 LSTM+CNN model

Previous research have proposed several tips to our approach. In 1997, Sepp Hochreiter *et al*. [1] invented the LSTM model to solve the problem of vanishing gradient and exploding gradient during long time series prediction. In 2019, Zhanhong He *et al*.[2], and in 2020, Ioannis E. Livieris *et al*.[3] illustrated that the utilization of LSTM layers along with additional convolutional layers could provide a significant boost in increasing the gold price forecasting performance. Besides, in 2021, GuoSihan [4] pointed out that the stream of bitcoin price is non-linear and in traditional economics, this currency is considered to be completely supplying inelastic, which means there is no apparent *linear* relationship between the production of bitcoin mining and its price. Take all the information into consideration, we've decided to build a "LSTM + CNN" model to predict gold and bitcoin prices, the structure of which is shown in figure 4.

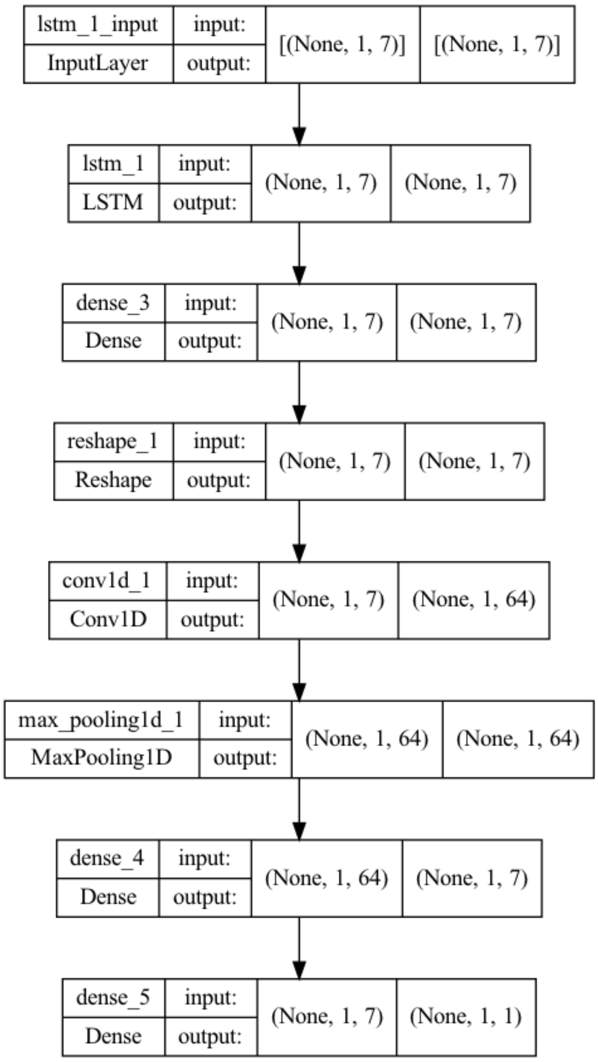


Figure 4: Network structure

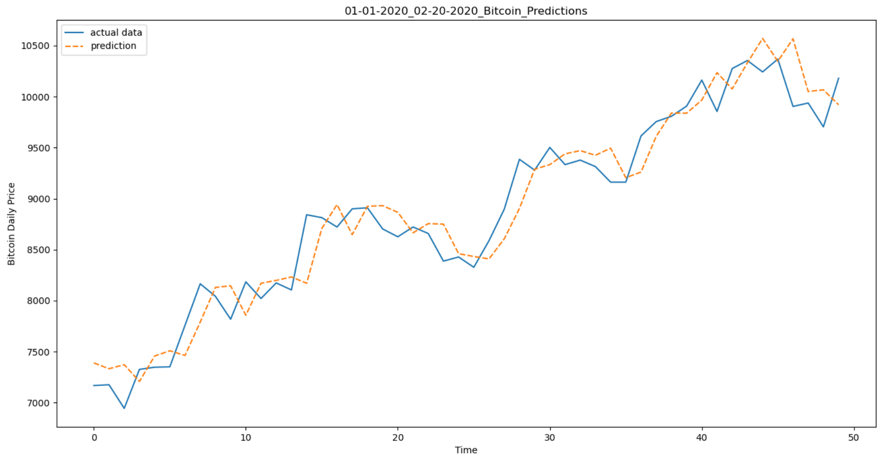
3.3 Dynamic training mechanism

In the paper [4], author Guo Sihan used some kind of dynamic method in model training to keep it up to date, and each time step is predicted by the model trained by price data right before that step. This method is also adopted in our approach.

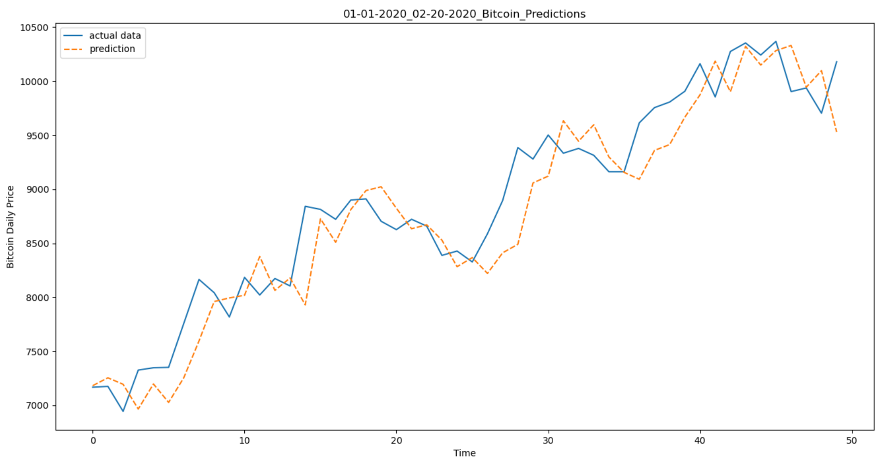
Due to the lack of training data at the beginning, we have to wait d_{wait} days (eg,120) to accumulate price data to train our prediction model for the first time. Later, all the data before d_i^* will be taken to train the model, which will be based on the *previous* one rather than training it starting from scratch. Meanwhile, the training epic will be reduced. Thus, not only the continuity of the model can be guaranteed, but also it will save lots of time. The d_i^* mentioned above refers to the ith trading day we choose, the first one is the 121th day, and the subsequent d_i^* will be selected according to the conclusion given by the trading model in section 4.3.

3.4 Prediction results

As we can see from figure 5, the prediction result matches well with the actual data, except for about a few days delay. Although the usage of smooth on raw data doesn't bring about much improvement in terms of prediction result, we have already shown the benefit of smoothing in the reduction of training time in subsection 3.1.



(a) without smooth



(b) with smooth

Figure 5: Prediction result regarding smooth method

4 Trading Algorithm

Every day in the trading period, our goal is to maximize the trader's final asset. In other words, we make the decision of whether to trade or not on a particular day to maximize the asset in the future, regardless of the profit or loss of the current trade with respect to previous trades. Our core idea is that the past cannot be changed (note that we've assumed that the price is given before trading on each day, the current price information on each day is in the "past" as well, since it won't change), therefore we have to make the trading decision that will maximize the overall assets on a particular point of time in the future. Inspired by *dynamic programming*, we present an algorithm to derive the best trading strategy for everyday.

4.1 Overview of the Trading Algorithm

The pipeline of our algorithm is illustrated in figure 6. We start by waiting for several days to accumulate training data, then we join the loop by training the prediction model and making future price predictions. We analysis the predicted price curve and determine the next trading date and the predicted price on that day. We optimize the asset on the next trading day with the aid of linear programming(since the target function of asset is linear) to get the optimum trading strategy for today. Now, we can purchase and sell asset according to the strategy derived and jump to the next trading date which was determined when analyzing the predicted price curve and start a new cycle of the loop. When we reach the end of the 5-year trading period, we jump out of the loop and finish the whole trading. The core parts of the algorithm are interpreted in the following sections in detail.

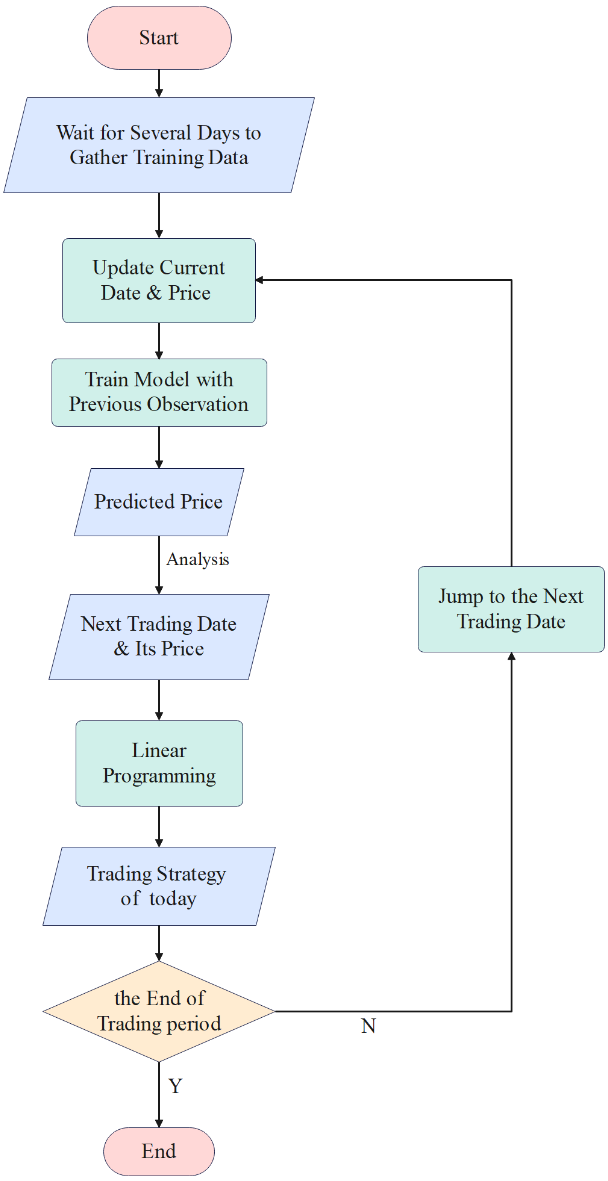


Figure 6: Trading algorithm pipeline

4.2 Analysis of prediction price

Before we demonstrate our analysis method, let us discuss the basic idea of trading. We can only gain profit when we purchase asset at a lower price and sell them at a higher price. The price difference is what we are relying on. Suppose the amount of a particular asset we possess is O, the lower price and higher price is P_low and P_high respectively and the commission rate is α,. The condition for gaining profit is

$$P_{high}O - (P_{low}O + P_{high}O) * \alpha > P_{low}O \tag{4}$$

thus

$$P_{high} > \frac{1+\alpha}{1-\alpha} P_{low} \tag{5}$$

The profit is $O * (P_{high}(1-\alpha) - P_{low}(1+\alpha))$.

In short, we are looking for larger price differences in the predicted data to get larger profit. Therefore, we look for peaks and troughs in the predicted price curve and address the corresponding date as potential next trading date, as shown in figure 7. The green dots showed the price on 2017-09-15, the left side of the green dots are previous observations, while the right side are predictions. The peak and trough are picked by locating the maximum and minimum price in the predicted curve within a given range of time. Our next step is to choose a specific date as next trading date and determine the trading strategy of today.

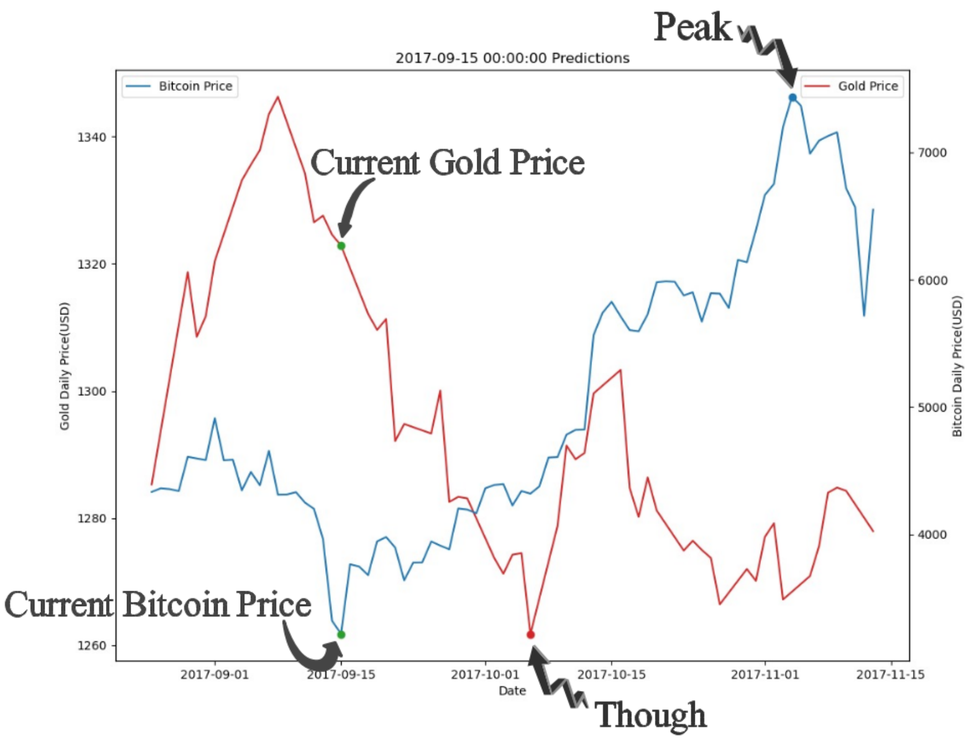


Figure 7: Example of peak and trough locating

4.3 Derive trading strategy

This section shows how we determine the next trading day among the candidates and trading strategy for the current day. Inspired by dynamic programming, we treat every trading day as a state and try to optimize the asset on the next state, i.e. the next trading day.

Suppose one of the potential next trading dates we found out is d_{i+1}^* with predicted gold price $P_{d_{i+1}^*}^G$ and bitcoin price $B_{d_{i+1}^*}^G$. The current assets is $[C_{d_i^*}, G_{d_i^*}, B_{d_i^*}]$ and the current day price is $P_{d_i^*}^G$ and $B_{d_i^*}^G$. Suppose we trade $O_{d_i^*}^G$ troy ounces of gold and $O_{d_i^*}^B$ bitcoins. The asset on d_{i+1}^* is

$$\begin{aligned} A_{d_{i+1}^*} = C_{d_i^*} - O_{d_i^*}^G P_{d_i^*}^G - O_{d_i^*}^B P_{d_i^*}^B - \alpha_{gold} O_{d_i^*}^G P_{d_i^*}^G - \alpha_{bitcoin} O_{d_i^*}^B P_{d_i^*}^B \\ + (G_{d_i^*} + O_{d_i^*}^G) P_{d_{i+1}^*}^G + (B_{d_i^*} + O_{d_i^*}^B) P_{d_{i+1}^*}^B \end{aligned} \tag{6}$$

To optimize the about function to get maximum asset on d_{i+1}^*, we can simply use linear programming since the function is linear and the boundaries are determined by our assumptions:

$$C_{d_i^*} - O_{d_i^*}^G P_{d_i^*}^G - O_{d_i^*}^B P_{d_i^*}^B - \alpha_{gold} |O_{d_i^*}^G| P_{d_i^*}^G - \alpha_{bitcoin} |O_{d_i^*}^B| P_{d_i^*}^B \geq 0 \tag{7}$$

$$O_{d_i^*}^G \geq -G_{d_i^*} \tag{8}$$

$$O_{d_i^*}^B \geq -B_{d_i^*} \tag{9}$$

Notice that there is an absolute value function in one of the boundaries, so we need to treat the linear programming problem with classified discussion. However, it is not a hard task for programming with Python. We treat every potential next trading date with the about method and choose the day that has the optimum asset as the next trading date and trade with the corresponding trading strategy. After that we can jump to next trading date and start a new loop.

5 Results

Using the given data and proposed model, we simulated the training process with Python (we trained the prediction model with TensorFlow). The trading history is shown in the figure 8. The $1000 asset eventually turned into $78658.58 after five years' trading!

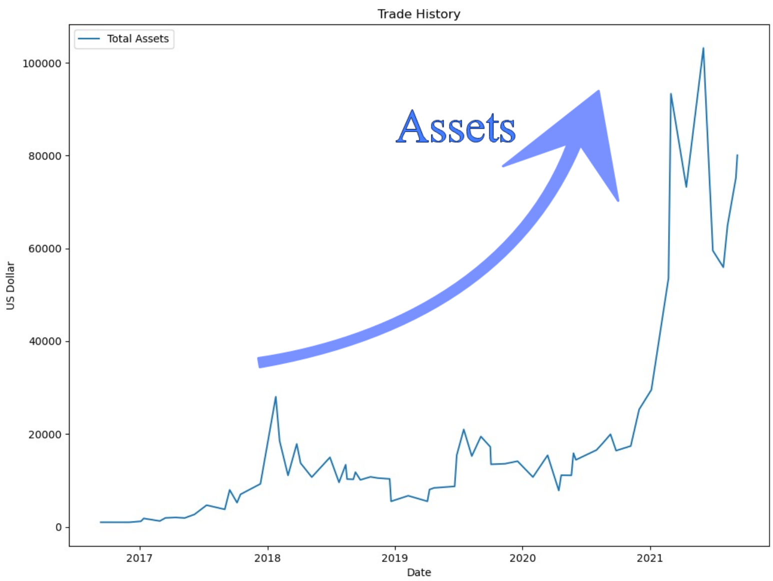


Figure 8: Trading history

6 Sensitivity analysis

In this section, we will discuss our model's sensitivity to commission rates. We change the commission rates and run simulations to get the following results.

As shown in figure 9, our final asset increases slightly with the reduction of commission rates, which means that the relative change is tiny. So our model is robust and insensitive to commission rates.

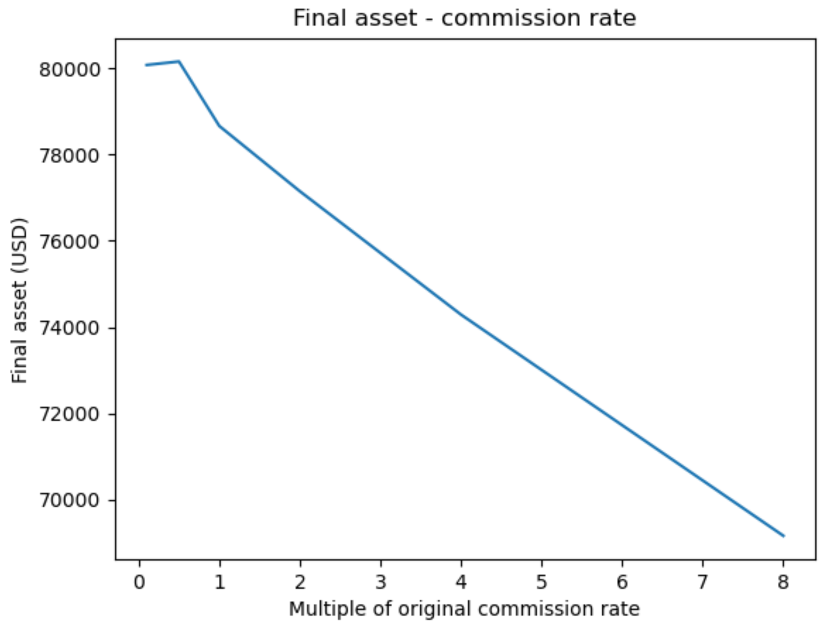


Figure 9: Sensitivity to commission rate

7 Model Evaluation

7.1 Strengths

1. According to our simulation, our model performs well on the given price data and is robust to different commission rates. The overall profit rate is 7765.86% with $\alpha_{gold} = 1\%$ and $\alpha_{bitcoin} = 2\%$.

2. Our model deals with noise by smoothing the training data. This measure allows our model to be **robust against noise**.

3. Our model features LSTM architecture in price prediction model. LSTM performs especially well on long time series forecast, which equip our model with a stronger prediction ability and greater accuracy.

7.2 Weaknesses

1. Our trading strategy aims to maintain a maximum asset on each coming trading day. It's actually a very greedy and risky approach. Since it desires maximum profit, it puts all asset at stake in order to gain profit. There will be huge loss when the trading strategy is erroneous.

2. Our method requires frequent training of neural networks in order to make accurate and up-to-date predictions. The training process consumes a lot of computing resources, and it takes a certain amount of time. Thus our method is not suitable for assets with rapid price fluctuations.

8 A memorandum to the trader

MEMORANDUM

To: trader

From: 2218530

Subject: Trading Strategy

Date: February 21, 2022

Dear Sir/Madam,

Thanks for reading this memorandum. We are a research team working on quantitative investing. We are honored to introduce our marketing strategy to you. This model will help you obtain maximum benefit with your portfolio consisting of cash, gold, and bitcoin [C, G, B] in U.S. dollars, troy ounces, and bitcoins, respectively, in a market where gold and bitcoin can be traded. We will be more than glad if you can make lots of money based on our approach.

Firstly, let us show you with the fantastic results of our simulation. To make it simple, we turn 1000 dollars into 78658.58 dollars when applying our approach on bitcoin and gold market from 09.2016 to 09.2021 (as shown in figure 10)! It means that you can expect a 139.4% annualized rate of return. In comparison to bank, the rate is 39.83 times of that for 5-year regular savings in the bank, which is just around 3.5%.

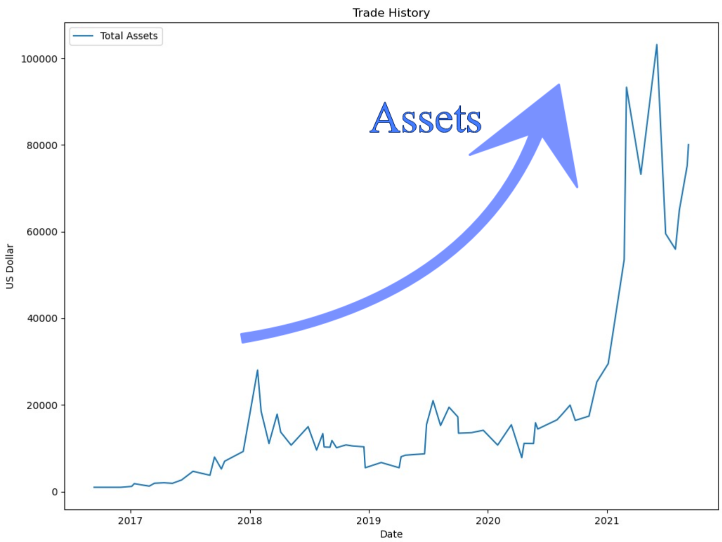


Figure 10: results

Secondly, we develop a method to predict the future prices of gold and bitcoin so that a optimum trading strategy can be derived to maximize the final assets. The frame of this approach is shown in figue 11

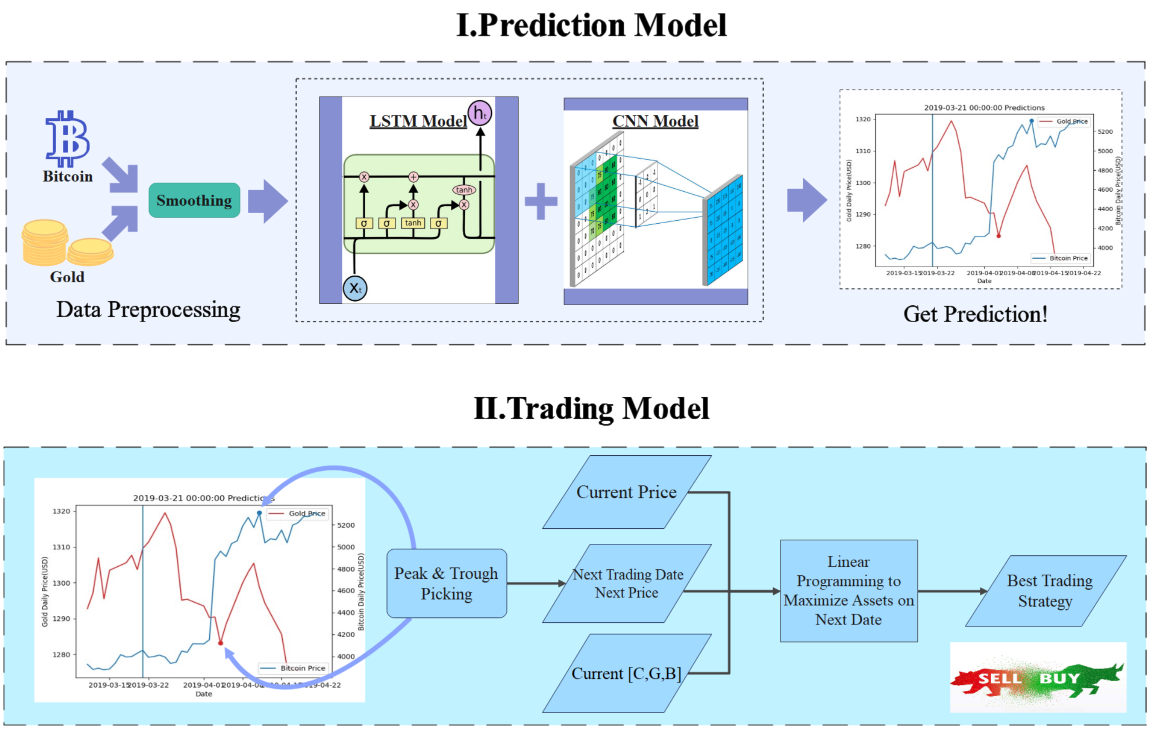


Figure 11: Modeling framework

Our approach contains two models:

Model(1): For price prediction, we adopt a deep learning model featuring a joint "LSTM-CNN" architecture to achieve high forecast accuracy. We also preprocess the training data by smoothing the price curves with a rectangular window to achieve a better result and help training loss converge faster at the same time. Then we take a suitable dynamic training mechanism to obtain the up-to-date model to ensure a decent prediction accuracy. Meanwhile, the training cost is also controlled within a reasonable and endurable range.

Model(2): As for the trading method, we are inspired by dynamic programming to develop an algorithm which determine the next trading day and maintain optimum overall asset on that day. To find trading days, we look for maximum and minimum prices in predicted price curves, which is more insightful, more guaranteed and more time-saving than running models everyday. Then, we optimize future assets with linear programming, which gives the best trading strategy for the current day according to the optimal principle of dynamic programming.

In addition, sensitivity analysis showed the total asset decreases roughly linearly as the commission rate multiplies. For example, when the commission rate is set to twice of its original value, the total asset shrinks to about 97.48% of its original value. So our approach is robust to noises and commission rate changes. However, our rather greedy and risky approach requires high prediction accuracy rapid training of neural networks, so there's still much improvement to make in the future.

To wrap up, we provide you with this up-to-date quantitative investment approach. I wish you making good use of it and good luck!

References

[1] Sepp Hochreiter, Jurgen Schmidhuber(1997). LONG SHORT-TERM MEMORY.

[2] Zhanhong He, Junhao Zhou, Hong-Ning Dai, Hao Wangy, (2019).Gold Price Forecast based on LSTM-CNN Model.

[3] Ioannis E. Livieris1, Emmanuel Pintelas1, Panagiotis Pintelas, (2020).A CNN–LSTM model for gold price time-series forecasting.

[4] Guo Sihan, (2021).Research on Bitcoin price prediction and trading strategy based on improved recurrent neural network.