Final Report

Cryptocurrency Prices Forecasting using Hidden Markov Models (HMMs)

CSDS 491 Spring 2021

Randolph Zhao, Marcus Lin, Shuhao Zhou

1. Motivation

We want to apply our knowledge we learned in the lectures to the real world. We have learned the Markov process and Markov chain which raises our interests in exploring probabilistic models using a series of previous data to predict the future trends. Hidden Markov Models (HMMs) are the answer, which is developed based on the Markov process but has a much wider real-world application potential due to its idea of separating observable and hidden states. We found HMMs are often applied to stock prediction problems. Since cryptocurrency is a hot topic these days and predicting cryptocurrency is more challenging than predicting stocks, we want to explore the HMMs on cryptocurrency prices forecasting problems. Furthermore, we learned that Long-Short Term Memory (LSTM) is also frequently used in stock prediction and thus we also want to compare the performance of LSTM with HMM on cryptocurrency price forecasting problems.

2. Introduction

In this paper, we introduce the cryptocurrency prices forecasting using Hidden Markov Models (HMMs). We develop a HMM adapted for cryptocurrency and test its performance based on Bitcoin prices from Yahoo Finance. We found that there is a one-day delay in the prediction of HMM, which we believe HMM is overfitting and learn a 'copy trick' due to the volatile prices of cryptocurrency. To solve the drift, we compared the performance of our HMM with different numbers of hidden states and applied it on other stock-FaceBook stock prices- where the prices of stock change less violently. Since Long-Short Term Memory (LSTM) is the most frequently used model in stock prediction, we also compare the performance of LSTM and HMM. We found if LSTM learned for a long time observation, LSTM also predicts prices with the one-day delay.

3. Background

Stock prediction is always a hot topic since the stock market is highly profitable but with high risks. There are different types of machine learning algorithms applied on stock prediction, but stock prediction still has various limitations due to instability, seasonality, unpredictability. Recently, cryptocurrency, the digital currency that is secured by cryptography, becomes a hot topic. Unlike the stock market, cryptocurrency has no limit up or down and runs 24/7, which makes cryptocurrency more profitable with much higher risks compared to stock. Besides, stock prices are largely based on company performance and finance, while cryptocurrency prices are based on nothing but the confidence of investors. Thus, cryptocurrency is a more challenging prediction task.

There are two major machine learning algorithms used frequently in stock prediction: Hidden Markov Models (HMMs) and Long-Short Term Memory (LSTM).

3.1 Hidden Markov Models

The Hidden Markov Model (HMM) is the probabilistic graphical model, where it has two kinds of states: observable states and hidden states (Figure 1). Hidden states are the states that contain 'hidden' information that decide the conditions of the observable states. They can be described by a Markov process [1]. Observable states are the states that can be directly observed by others and that we are trying to predict. These two kinds of states make HMM have a wide range of applications. HMM will learn the probability distribution of each hidden state from a series of observable states and use the learned hidden states to predict the future observable states. Thus, HMM is often used to predict or analyze facts that are based on time, such as speech recognition, natural language processing, and stock prediction.

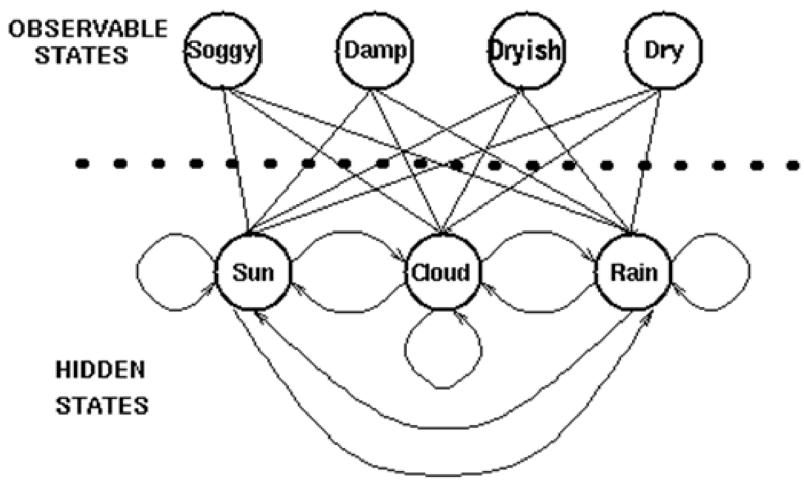

Figure 1: two kinds of states in Hidden Markov Models [1]

Due to the two kinds of states of a HMM, it is very intuitive to train a HMM on stock prediction. For investors, observable states are close, open, high, low prices for each day. Hidden states are the states that drive the changes of prices but are invisible to investors. Instead of using the close prices directly, we uses the fraction of changes as the $frac_{change} = \frac{(close-open)}{open}$, $frac_{high} = \frac{(high-open)}{open}$, $frac_{low} = \frac{(open-low)}{open}$ based on the open prices [2]. Similarly, the predicted observable states will be calculated as $close = open \times (1 + frac_{change})$, $high = open \times (1 + frac_{high})$, $low = open \times (1 - frac_{low})$. Since the distribution of prices can be described as a Gaussian distribution, we will use a continuous Gaussian Hidden Markov Model.

3.2 Long-Short Term Memory (for further exploration)

Since the long short term memory(LSTM) is widely used on stock prediction, we decide to explore how it works and compare it with the performance of HMM.

First, LSTM is an updated version for Recurrent Neural Networks(RNN), which allows previous outputs to be used as inputs while having hidden states. However, RNN suffers from short-term memory. If a sequence is long enough, RNN will have a hard time carrying information from earlier time steps, and may leave out important information from the past. Also, RNN suffers from the vanishing gradient problem, which occurs when the gradient, which used to update a neural network's weight, shrinks during back propagation.[3] If a gradient value becomes extremely small, it doesn't contribute too much learning. LSTM is created as the solution for short-term memory. It has internal mechanisms called gates that can regulate the flow of information. These gates can learn which data in a sequence is important to keep or throw away, and it can pass relevant information down the long chain of sequences to make predictions.

As shown on Figure 2, there are 3 gates in one LSTM cell state: forget gate, input gate, and output gate. The cell state information is passed on the upper chain, and the hidden state, which contains information on previous inputs, is passed on the lower chain.

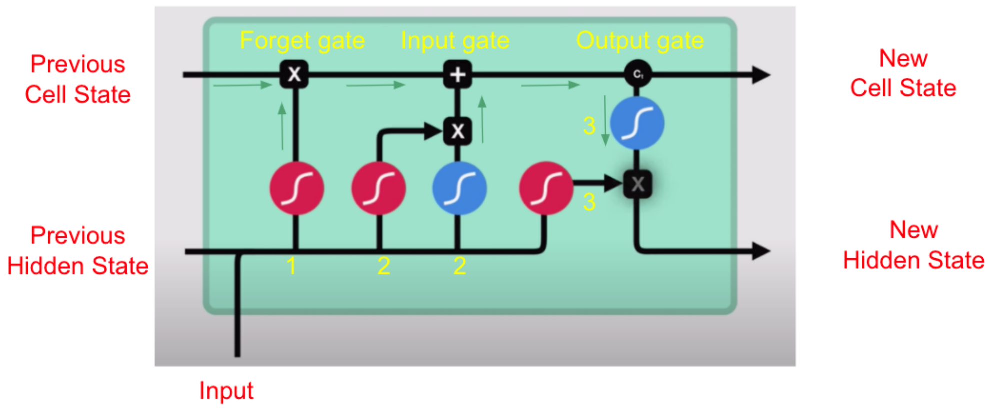

Figure 2: Mechanism in one LSTM cell state [3]

Firstly, at step 1 on Figure 2, it comes to the forget gate which decides what information is relevant to keep from prior steps. The input and previous hidden state information are passed into the sigmoid function, which squishes values between 0 and 1, and then it times the previous cell state, so part of the information will be forgotten.

Then, information goes to the input gate which can decide what information is relevant to add from the current state. The previous hidden state and input are passed into the sigmoid function again to get how much information would be preserved, and the tanh function which squishes data between -1 and 1, then we add the product of these two values to determine how much information is added.

After that, we already get the new cell state information by forgetting some previous state information and adding new state information. The output gate decides what the next hidden state should be, and the next hidden state is just the new cell state times the sigmoid value. [3]

4. Experiments

4.1 Hidden Markov Models

We first implement our HMM on Bitcoin prices downloaded from Yahoo Finance and our HMM is designed to have four hidden states. The training data is ‘BTC-USD’ from 1/1/2015 to 3/1/2021 and we predict the prices of the next two month: 3/1/2021 to 5/1/2021. We first only use close price as our observations. As Figure 3 shows, the predicted close price seems to successfully capture each change of the close price and have a good prediction accuracy at first glance.

Then, we use close price, high price, and low price as our observations at the same time. As Figure 4 shows, the predicted close prices has the highest accuracy and the predicted low price is the poorest one, although our HMM still predicts the trend of changes in low prices successfully.

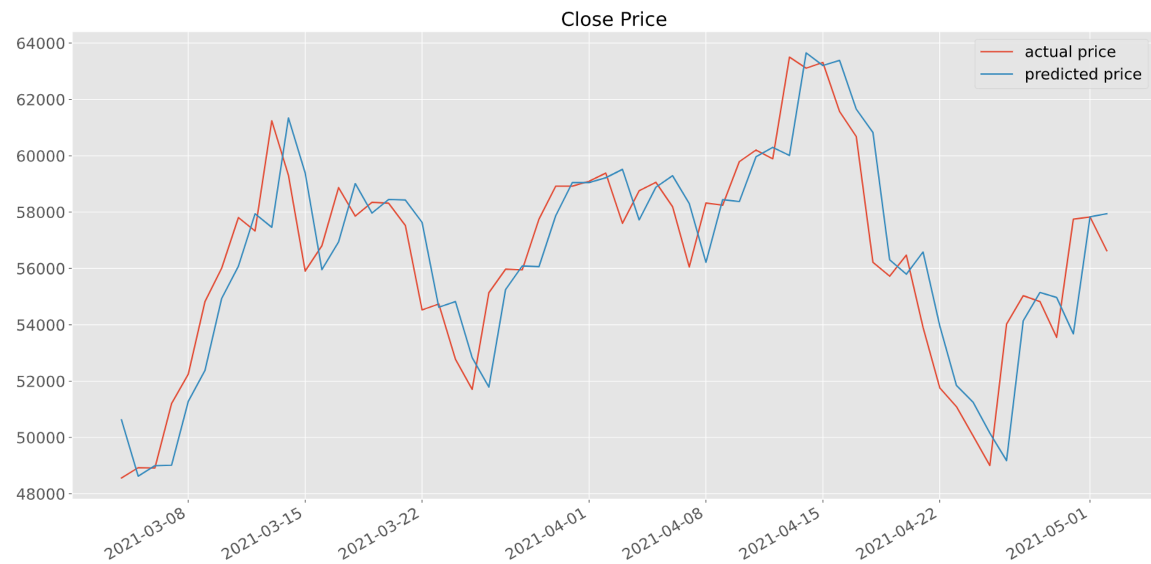

Figure 3: Close price prediction of Bitcoin

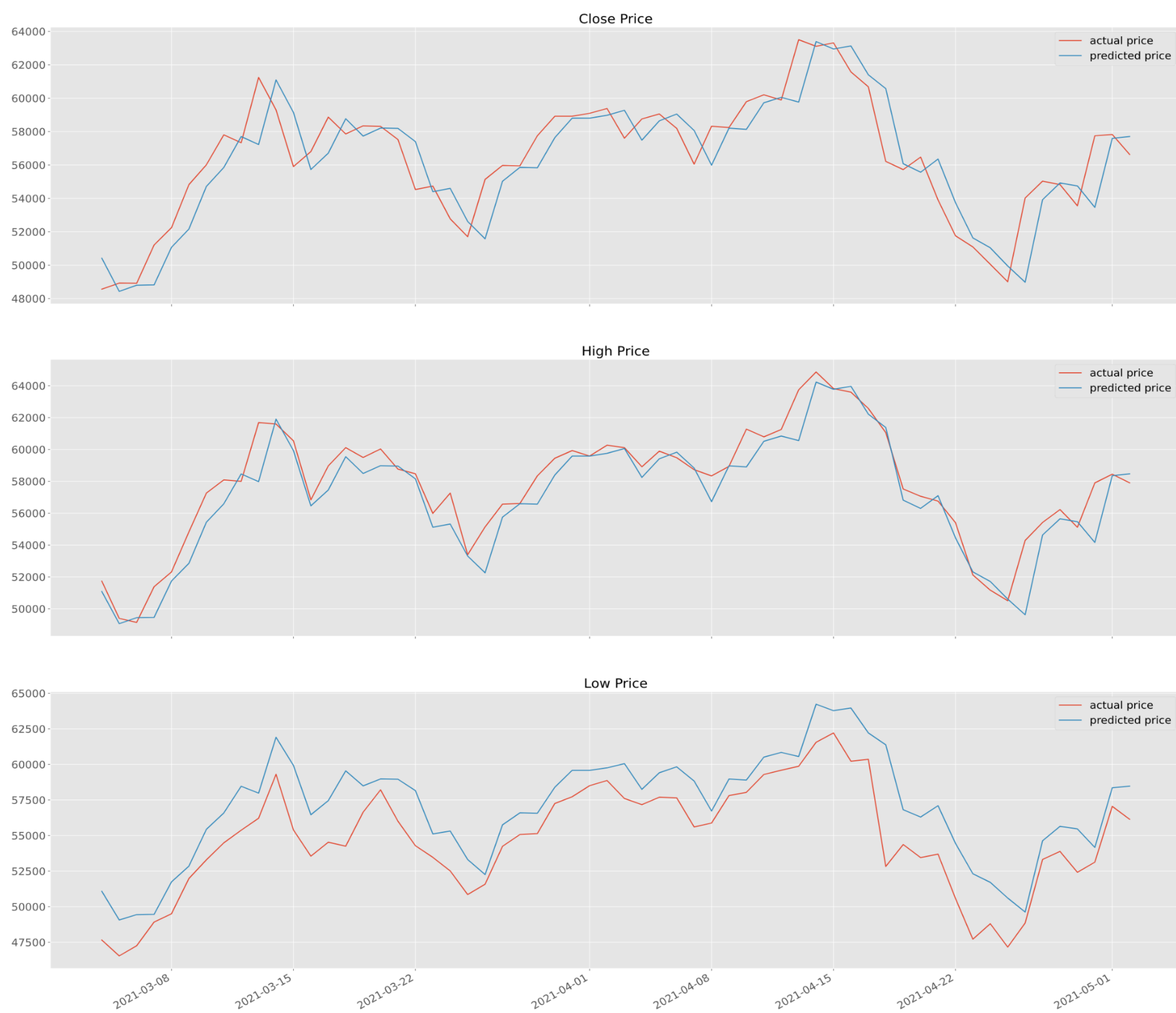

Figure 4: Close price, high price, low price prediction of Bitcoin

5. Result Analysis

When taking a closer look at the results, we can observe that there is a slight right shift from the actual price to the predicted price in the HMM prediction. Although the accuracy of the prediction shows HMM has great performance, HMM only predicts the next value with the previous one. The reshown of previous actual value can cause great potential financial damage to users who want to use this HMM to predict the bitcoin stock price. The up and down and down in stock price is shown only after the actual up and down is taken place.

According to [4], this kind of shift in the prediction is called ‘copy trick’, this means that the HMM did not actually capture the underlying processes that determine the stock prices and only simply learned to predict a value close to the previous value.

To explain this shift in the prediction, our team makes two assumptions:

1. The ‘copy trick’ is caused by complex number of hidden states
2. The ‘copy trick’ is caused by violent fluctuation in market stock price

5.1 Number of Hidden States

The complex hidden states may overfit the HMM during the training. Large numbers of hidden states may cause the model to be trained more closely to the train dataset. The correlation between actual data and prediction may tend to be high when the number of hidden states increases.

To explore the influence of different numbers of hidden states to the prediction, our team varied the number of hidden states from 1 to 10. Figure 5 shows the excerpts when the number of hidden states equal to 2, 4, 6 and 8.

In figure 5, by changing the number of hidden states, the correlation between actual data and prediction doesn’t change. Which means, the number of hidden states is not the main factor causing the shift in the prediction.

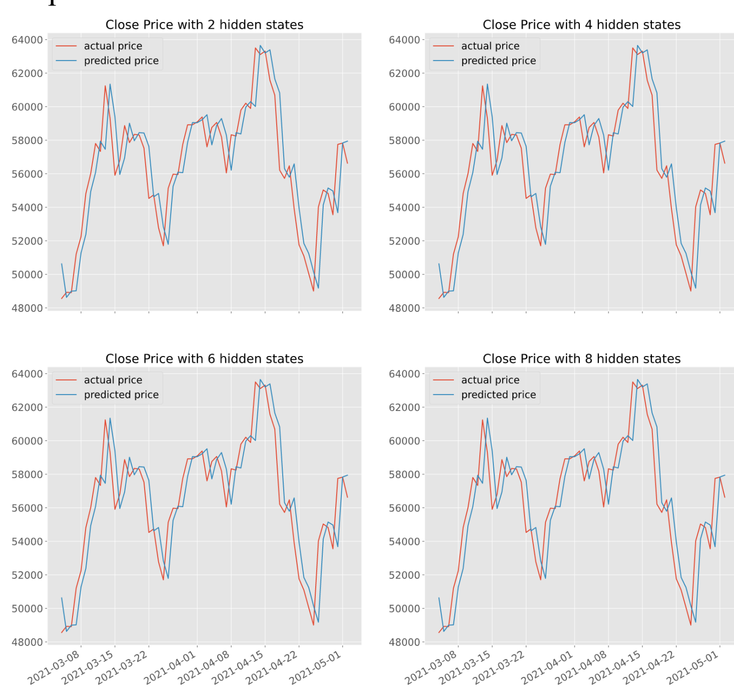

Figure 5: Close Prediction of Bitcoin with different number of hidden states

5.2 Violent Fluctuation

The price of bitcoin fluctuates so violently that HMM cannot capture the general important information under the process. Too much attention is paid on the trends shown in the recent observations and the previous value is taken to predict the next day's value.

To explore the influence of fluctuation in the market to the prediction, a more easing company, Facebook, is introduced to the model. According to Figure 6, by replacing the bitcoin market with Facebook's stock price, the 'copy trick' is not so obvious anymore. The rise and fall in the stock market can be predicted on the same day.

Thus, HMM can follow the general trend of the actual data when the market fluctuates at a relatively slow rate. Which means, the non-volatile market is more suitable for HMM prediction.

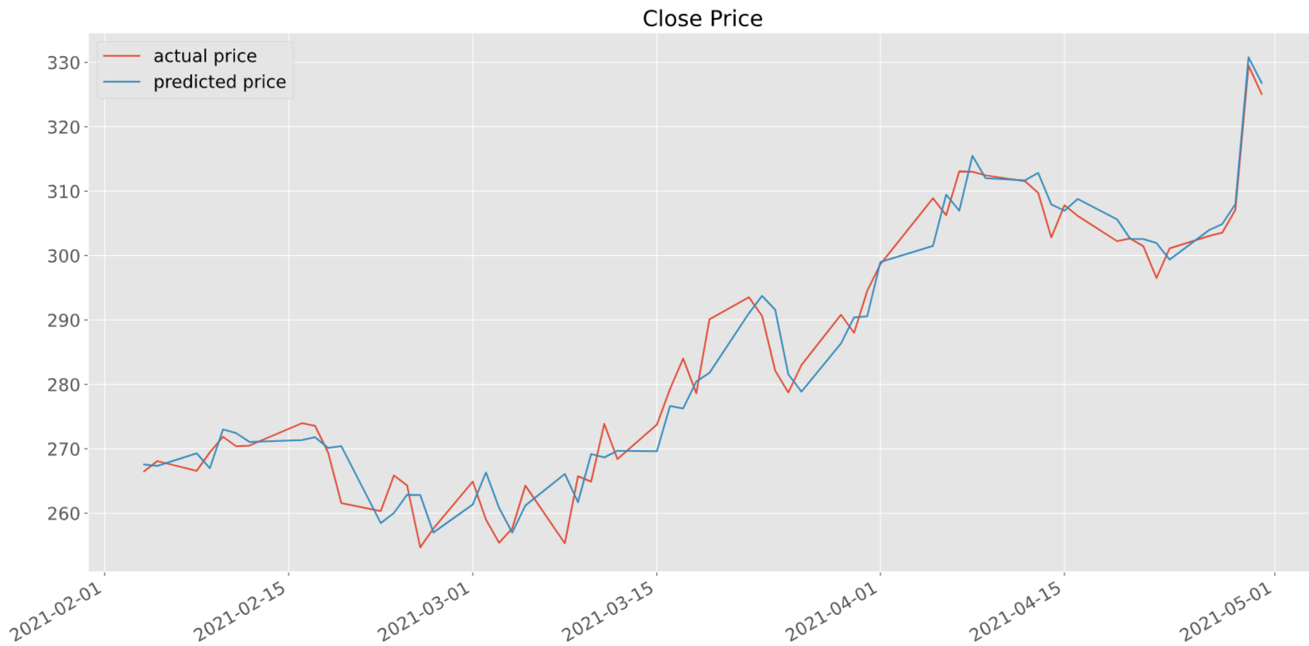

Figure 6: Close price prediction of Facebook stock prices

6. Further Exploration - LSTM

To prepare the dataset, we import the close price data from 2015/1/1 to 2021/3/1 for BTC-USD price on Yahoo finance, and normalize the data by MinMaxScaler. Then, we use the Sequential model and add three LSTM layers onto it, and dense it to 1 unit. We compile the model by 'Adam' optimizer and use the mean square error for the loss. To train the model, we use batches of size 32.

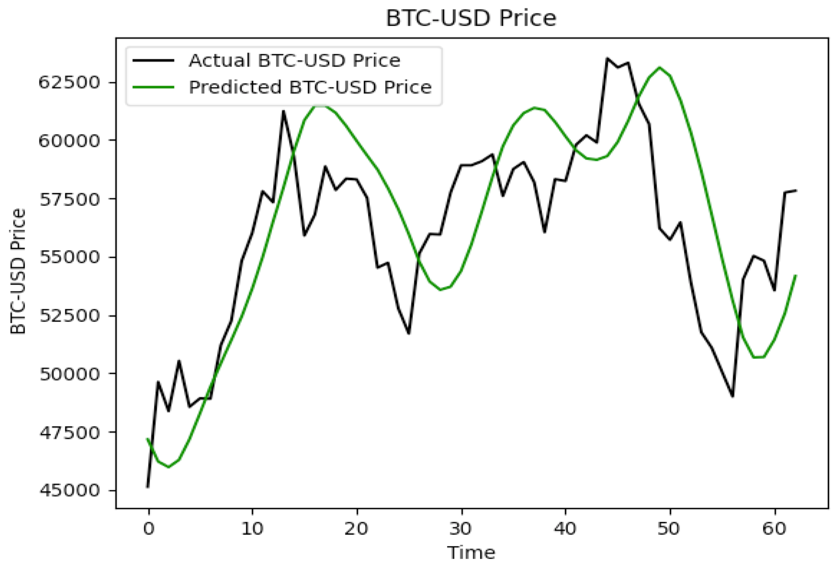

Figure 7: Close price prediction of BTC for 2 months by LSTM

Then, we predict on the test dataset which is from 2021/3/1 to 2021/5/1. On figure 7, we can see the general predicted trend is close to the actual trend, but the same as HMM, there is a delay as time goes on. Therefore, we decided only to predict the next day's close price after we trained the model, and this is close to reality since people often predict the next day’s stock price to decide whether to trade the next day or not. We iterated this process for the past seven week, and the result is shown on Figure 8. Now, if we don’t look at the price but the trend, we can see the prediction does not have a delay, besides the second day and the fifth day, it is actually making good predictions. Compared to hmm’s result, we can see LSTM has better performance.

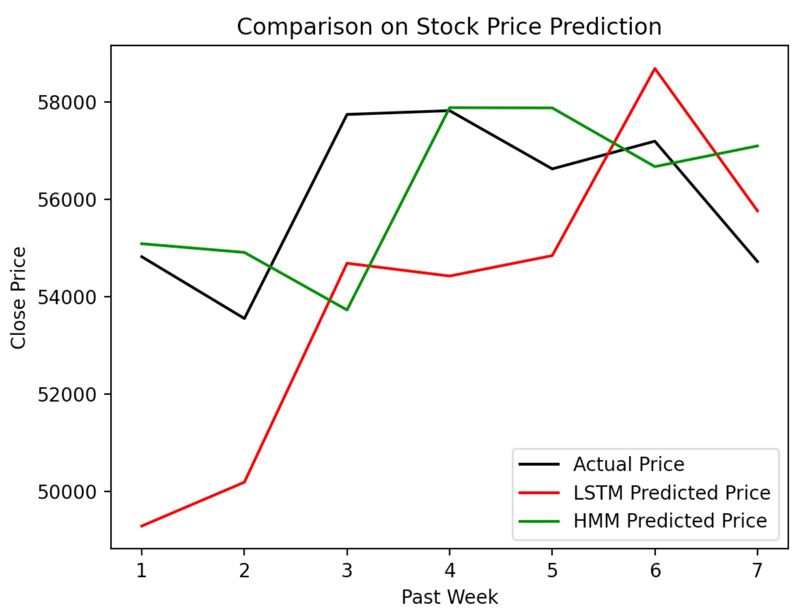

Figure 8: Close price prediction for the past week on BTC

7. Conclusion

Compared to the LSTM, HMM is a probabilistic model which

1) Has a shorter runtime. HMM is easy to train and test.
2) Easy to implement. HMM has less parameters to adjust.
3) Easy to adapt to prices that change sharply. From Figure 6, HMM correctly predicted the sharp increase in the same day accurately and precisely.

However, when applied on the cryptocurrency price forecasting problem, which fluctuates violently, HMM is easy to ovefit and apply the ‘copy trick’. If the ‘copy trick’ is taken into account in assessing the performance of HMM for such time sensitive data prediction, HMM might not capture the trend underlying the market.

Thus, if HMM is applied to a correct market, a market with smooth fluctuation rate with a long period of time, it can provide more accurate and precise prediction compared to LSTM especially on the sharp change in price. If the market has more violent fluctuation within a short period of time, LSTM is more recommended since it can capture the trends better with no ‘copy trick’ appearing.

For future exploration, a more indepth research can be done in finding the cause of ‘copy trick’ in HMM because this misleading accuracy is not only shown in [4] but also many other HMM stock prediction projects. One approach could be to feed the HMM with a violent fluctuation market within a long period of time. HMM might learn the underlying trend of the market if the time period is long enough.

To achieve the best performance from LSTM, a wide range of hyper parameters can be fed to the model and compare the MAPE (Mean Absolute Percentage Error) and RMSE (Root Mean Square Error) to get the best output.

Reference

1. Shalini Venkataraman and Vidhya Gunaseelan, "HIDDEN MARKOV MODELS IN COMPUTATIONAL BIOLOGY", Retrieved from https://www.evl.uic.edu/shalini/coursework/hmm.ppt.
2. A. Gupta and B. Dhingra, "Stock market prediction using Hidden Markov Models," *2012 Students Conference on Engineering and Systems*, 2012, pp. 1-4.
3. Michael Phi. "Illustrated Guide to LSTM's and GRU's: A step by step explanation", Retrieved from https://towardsdatascience.com/illustrated-guide-to-lstms-and-gru-s-a-step-by-step-explanation-44e9eb85bf21
4. Bram de Wit, "Stock Prediction Using A Hidden Markov Model Versus A Long Short-Term Memory", Retrieved from http://fse.studenttheses.ub.rug.nl/20683/1/AI_BA_2019_Bram.pdf