

Incorporating Sentiment and Temporal Information for Bitcoin Price Prediction

Fengyuan Shen*

School of Physics, Xi'an Jiaotong University, Xi'an, China

*Corresponding author: johnluacs@stu.xjtu.edu.com

Abstract. Recent years have witnessed the rapid development of bitcoin as the first digital currency. Considering the advantages of bitcoin for both individuals and society, the price prediction of bitcoin is a hot topic. However, there remain two main challenges to be addressed in this task. Firstly, because bitcoin is vulnerable to the attitudes of the investors, incorporating the sentiment semantics from the social media into the prediction is challenging. Secondly, it is intractable to predict extreme volatility of the bitcoin price. To tackle the above challenges, this paper proposes to incorporate sentiment and temporal information simultaneously. For the first challenge, this paper employs external unsupervised corpus to conduct the domain-specific post-pretraining on the off-the-shelf language model. And the sentiment analysis on the tweets is done to obtain the scores. For the second one, Long Short Term Memory (LSTM) network is leveraged to joint model the temporal price data and the sentiment scores, thus deriving the final predictions. Experiments on the real data show that compared with the single-layer LSTM model, the model in this paper works better, which provides help for investors to specify trading strategies and also provides implications for government agencies that are developing digital currencies.

Keywords: Bitcoin, Finance, Sentiment analysis, Transformer, LSTM.

1. Introduction

Bitcoin is the first digital currency to operate independently of any central bank or authority[1, 2]. Due to the digital and anonymous nature, bitcoin has aroused wide interests in the world. Meanwhile, some previous works[3, 4] have shifted their attention to the prediction of bitcoin price, owing to the following advantages. On one hand, as many countries are currently developing digital currencies, the study of bitcoin, the pioneer of digital cryptocurrencies, can help bring some reference to the development of digital currencies in each country. Also, it has some practical significance for the development and promotion of national digital currencies. On another hand, studying the price mechanism of bitcoin can create a reasonable and objective understanding for investors and better help them specify their investment strategies.

However, there still exist two challenges in the task of bitcoin price prediction. Firstly, since bitcoin price is vulnerable to investors' sentiment[5-7], it is challenging to mine precise sentiment semantics from social media, such as Twitter. Shah et al.[4] applied the latent source model from Bayesian regression to bitcoin forecasting. McNally et al.[8] predicted bitcoin price through Bayesian optimized recurrent neural network and Long Short Term Memory (LSTM) network. Both of them lack the modeling of sentiment information, which makes the prediction less convincing. Stenqvist et al.[9] predicted the price by analyzing 2.72 million bitcoin related comments with a prediction accuracy of up to 83%. But it does not apply the large pretrained language model (PLM)[10, 11] to conduct the sentiment analysis and lacks the modeling of domain-specific knowledge[12]. Secondly, it is intractable to predict extreme volatility of the bitcoin price. Due to its scarcity and unique issuance mechanism, it is more difficult to predict bitcoin price compared to the stock market. For example, Muzammal et al.[13] found that the price of bitcoin is highly volatile, so it is different from the price forecast in traditional financial markets. Katsiampa et al.[14], Ozturk et al.[15], and Duan et al.[16] used traditional time series forecasting techniques to anticipate bitcoin price but the problem of extreme volatility of bitcoin price has not been well solved.

In view of the above challenges, this paper proposes to incorporate sentiment and temporal information simultaneously. For the first challenge, the off-the-shelf PLM is introduced to conduct

the sentiment analysis. Specifically, this paper employs external unsupervised corpus to conduct the post-pretraining, thus enhancing the domain-specific capability of the PLM. For the second challenge, this paper employs LSTM to model the temporal information. Combining both the calculation of the log return price and sentiment scores, final prediction results can be derived. Experiments show that after introducing sentiment scores, the prediction accuracy of the model is superior to that of using only a single metric, bitcoin price.

2. Related Works

To predict the bitcoin price, one needs to know what factors affect the price of bitcoin and also what prediction models are currently available to make more accurate predictions. Therefore, this section will be divided into two parts to present the current researches.

2.1 Factors Affecting Bitcoin Price

For predicting the price of bitcoin, one must take into account bitcoin's influential determinants because its series shows complex and highly volatile characteristics. According to the research of Kristoufek et al.[6], the amount of searches on the Wikipedia website and Google search trends are connected with bitcoin exchange rates. According to the research of Polasik et al.[7], the frequency of transactions, media coverage, and overall popularity of bitcoin all influence exchange rates. Network traffic, a specific indicator of bitcoin trading complexity, has been discovered by Guesmi et al.[17] to be highly connected with the returns and volatility of the bitcoin market. Time-series analysis was used by Georgoula[5] to investigate the relationships between bitcoin prices and various economic, technical, and sentimental metrics from Twitter subscription feeds. According to the findings, Twitter popularity ratio is correlated favorably with the price of bitcoin.

2.2 Prediction Method of Bitcoin Price

Traditional time series forecasting techniques like Autoregressive (AR), Moving average (MA), and Autoregressive Integrated Moving (ARIMA) were employed by Katsiampa et al.[14], Ozturk et al.[15], and Duan et al.[16] to anticipate bitcoin prices and volatility, respectively. In contrast to Autoregressive Moving Average (ARMA), machine learning techniques enable us to capture the non-linear characteristics of extremely volatile cryptocurrency pricing. According to the analysis of Wu et al.[18], while neural networks successfully approximate the log return distribution of bitcoin, more advanced learning techniques like Recurrent Neural Network (RNN) and LSTM can result in predictions with a higher degree of accuracy. Ramadhani et al.[19] compared ARIMA model with LSTM model to estimate the future price of bitcoin and found that the LSTM model had a significantly lower average absolute error, thus indicating that the LSTM is more accurate when used for bitcoin price prediction.

3. Methodology

To address the task of bitcoin price prediction, this paper proposes to leverage the Transformer-based LM to obtain the sentiment analysis scores from tweets, and utilize LSTM to joint model both sentiment scores and temporal information. The detailed methods are discussed as follows.

3.1 Domain-specific Sentiment Analysis Based on Transformer

Since the price of bitcoin is vulnerable to the influence of social media, the emotional attitudes towards bitcoin from people may be fundamental to the prediction accuracy. This paper utilizes the Transformer-based model with the unsupervised domain pretraining to model the sentiment semantics obtained from tweets.

3.1.1 Transformer Architecture

Transformer was proposed by Google[20] in 2017 and has aroused wide interests[12, 21, 22] in some previous work. Its entire network structure is composed of attention mechanism. More specifically, Transformer is composed of Self-Attention and Feed Forward Network. Transformer based neural networks can be constructed by stacking Transformer structures.

Transformer has two distinct mechanisms in its most basic form: an Encoder that receives text input and a Decoder that produces predictions for different tasks. Fig. 1[20] shows its complete structure:

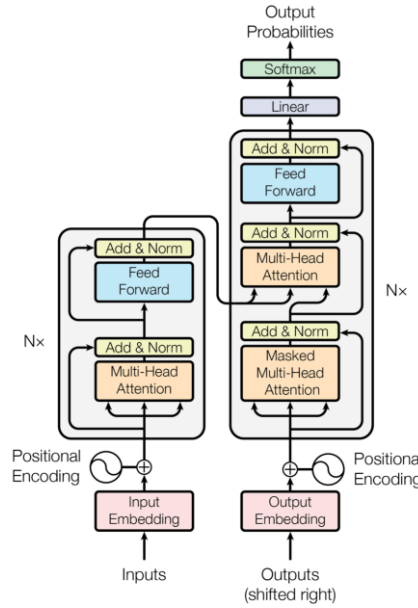


Fig. 1 Transformer model structure

In the Encoder structure of the Transformer, the input data will first get a feature vector X , namely $MultiHead(Q, K, V)$, through the Multi-Head Attention module:

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d^k}}\right)V, \quad (1)$$

$$head_i = Attention(QW_i^Q, KW_i^K, VW_i^V), \quad (2)$$

$$MultiHead(Q, K, V) = Concat(head_1, \dots, head_h)W^O, \quad (3)$$

Where d^k is the dimension of queries and keys, and the projections are parameter matrices $W_i^Q \in \mathbb{R}^{d_{model} \times d_k}$, $W_i^K \in \mathbb{R}^{d_{model} \times d_k}$, $W_i^V \in \mathbb{R}^{d_{model} \times d_k}$ and $W^O \in \mathbb{R}^{hd_v \times d_{model}}$.

The input feature X will then be transmitted to the Encoder's subsequent module: Feed Forward Network (FFN). This is a fully connected layer divided into two layers. The ReLU activation function is the first layer, and the linear transformation function is the second layer. It is modeled as follows:

$$FFN(X) = max(0, XW_1 + b_1)W_2 + b_2, \quad (4)$$

The Self-attention and the Feed Forward Network make up the Transformer's Encoder. Self-Attention, Encoder-Decoder Attention, and Feed Forward Network make up its decoder structure. In addition, Transformer obtains the ability of capture the order of the sequence by introducing Position Encoding. The Position Encoding is to add the position information of words in the word vector, so that the Transformer can recognize words in different positions.

3.1.2 Domain-specific Pretraining

Considering that some Transformer-based models (e.g., BERT, RoBERTa) are pretrained on large-scale corpus in the general domain (e.g., wikipedia), they may struggle at some domain-specific tasks. In the bitcoin-related scenes, the comments from tweets have close relations with the domain of financial technology. Therefore, there naturally exists a gap between the PLM and the downstream sentiment analysis task on bitcoin tweets. To this end, this paper attempts to inject the domain knowledge into the PLM to enhance the quality of text representation.

At the very beginning, this paper crawls abundant corpus about financial technology over the internet and reorganizes them with several sentences per line. Since the obtained corpus has no supervision signal, the pretraining stage is conducted in a self-supervised manner. Following the common practice in previous works, this paper leverages Mask Language Modeling (MLM) as the training objective. To further attend to the terminologies in the specific domain, this paper introduces the span mask strategy. That is to say, this paper randomly masks a text span (consisting of some tokens) each time, and the span length follows the geometric distribution. In this way, PLM can focus more on the span-level semantics, rather than the token-level and word-level information in the previous works. After domain-specific pretraining, the PLM is equipped with more domain knowledge and is more suitable for the down-stream task.

3.1.3 Sentiment score

In this paper, RoBERTa-base model is used to analyze bitcoin related tweets[11]. The model will return three labels: Positive, Negative and Neutral, as well as their respective scores.

According to the output of RoBERTa model, the following formula is defined to calculate the sentiment score of each tweet:

$$SSET = 1 \times Positive + (-1) \times Negative + 0 \times Neutral, \quad (5)$$

Where $SSET$ represents the sentiment score of each tweet.

In this way, after doing sentiment analysis on all tweets, each tweet has its own sentiment score. However, considering the large number of tweets to be processed, sentiment analysis on all tweets may take a lot of time in the real applications, so the number of tweets must be streamlined.

According to the like function, people's attitudes towards each tweet can be tracked. Generally, the higher the number of favorites a tweet receives, the more people support the tweet's view. In the actual calculation process, tweets are grouped every 6 hours, and then the top 30 tweets in each group with the highest number of favorites are intercepted, so the sentiment of people on Twitter towards bitcoin in 6 hours can be calculated by the following formula:

$$TSS = \frac{\sum_1^{30} SSET \times \text{Number of Favorites}}{30}, \quad (6)$$

Where TSS denotes total sentiment score of top 30 tweets every 6 hours. The denominator is chosen to be 30 instead of the total number of favorites in 6 hours because it clearly shows the difference in people's sentiment towards bitcoin in different time periods.

3.2 Prediction of Log Return Price of Bitcoin Based on LSTM

Section 3.1 describes how to perform domain-specific sentiment analysis on bitcoin related tweets, and this section will focus on why and how to use LSTM for bitcoin price prediction.

3.2.1 RNN and LSTM

RNN takes advantage of its short-term memory by constructing connections between neurons in the same hidden layer to obtain the before-and-after correlation information of the data. The input information of the hidden layer comprises both the output information from the hidden layer at the prior moment and the input information from the current moment, forming a temporal dependency.

With this connection structure of implicit nodes at different moments, the RNN is capable of realizing the recollection of information from past moments and using it to compute present output.

However, when the RNN is applied to long time span sequences, the gradient disappearance and the gradient explosion leads to a small memory because all layers share the same weight parameters and the derivative multipliers of the activation function accumulate, so the RNN is often ineffective in dealing with long time series problems.

To solve the problem of gradient disappearance when faced with long sequence data, Hochreiter and Schmidhuber[23] proposed the LSTM network in 1997, which effectively alleviates the problem generated by traditional RNN models through a specially designed gate. LSTM introduces a gated cell system based on RNN, including input gate, forgetting gate and output gate. These three types of gates control the current input data, the update of historical data to the memory cell state and the output, respectively. The information is selectively controlled by different gates through which the network learning can appropriately forget the historical information and update cell state based on new information. Fig. 2 illustrates the fundamental structure of LSTM:

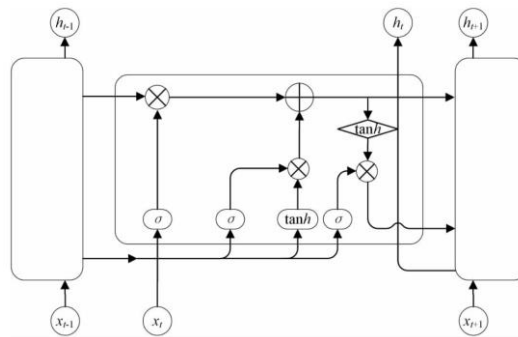


Fig. 2 LSTM model structure

The calculation procedure of each LSTM unit is broken down into the following 3 steps:

(1) The candidate memory cell value $\tilde{c}_t$, the input gate value i_t and the forgetting gate value f_t at moment t are calculated respectively, as follows:

$$\tilde{c}_t = \tanh(\omega_c[h_{t-1}, x_t] + b_c), \quad (7)$$

$$i_t = \sigma(\omega_i[h_{t-1}, x_t] + b_i), \quad (8)$$

$$f_t = \sigma(\omega_f[h_{t-1}, x_t] + b_f), \quad (9)$$

Where ω_c , ω_i and ω_f are the corresponding weight matrices; b_c , b_i and b_f are the corresponding biases; h_{t-1} is the output of the LSTM cell prior to the moment t ; x_t is the value of the memory cell at moment t ; and σ is the sigmoid function.

(2) Multiply old state with the forgetting gate and discard some information, add input gate with the value of candidate memory cell, and obtaining the memory cell's current value c_t :

$$c_t = f_t c_{t-1} + i_t \tilde{c}_t, \quad (10)$$

(3) Finally, the value of the output gate o_t is confirmed by the output gate and the output h_t is determined by:

$$o_t = \sigma(\omega_o[h_{t-1}, x_t] + b_o), \quad (11)$$

$$h_t = o_t \tanh(c_t), \quad (12)$$

Where ω_o , b_o are its corresponding weight matrix and bias.

By establishing the above control gate and memory cell structure, longer sequence of data processing can be achieved, and problems such as gradient explosion are solved.

3.2.2 Prediction of bitcoin's log return price

The previous section shows the advantages of LSTM over RNN and gives a brief explanation of the principles of LSTM. This section will do some processing on bitcoin price to enable the LSTM model to make better prediction results based on the input values.

Compared to the traditional stock market, bitcoin price rises and falls more dramatically because of the lack of companies or government agencies to guarantee its value. Its price is more influenced by investors' sentiment. There is not much practical significance by simply using an LSTM model to predict the price of bitcoin. For example, suppose bitcoin is now worth \$20000, and the LSTM model predicts that the price will be \$20001 tomorrow, with MAPE (Mean Absolute Percentage Error) of 1% in the model, then according to the trading strategy, investors should buy bitcoin. However, the actual price of bitcoin the next day is \$19999, and although that actual value is within the error range, which means the prediction is accurate, the investor loses money because of the predicted value given by the model. So, it makes more sense to say that when using LSTM to predict the price of bitcoin, one should focus on whether its price is going up or down.

Therefore, how to predict the rise and fall of the bitcoin price with an LSTM model? It is necessary to do some processing on the bitcoin price. Instead of using the bitcoin price directly, this paper calculates the log return price of bitcoin. First, take the logarithm of the price, and then subtract the logarithm price of the previous time from the logarithm price of the current time to get current time's log return price:

$$LRP_n = \log(P_n) - \log(P_{n-1}), \quad (13)$$

Where LRP_n means log return price of bitcoin at moment n , P_n means the price of bitcoin at moment n , P_{n-1} means the price of bitcoin at moment $n - 1$.

For example, suppose the last 3 days of bitcoin prices are \$20000, \$20010 and \$19990, then bitcoin's log return price of today is $\log(19990) - \log(20010)$, and bitcoin's log return price of yesterday is $\log(20010) - \log(20000)$. By doing so, people can directly tell whether the price is going up or going down according to the positive and negative of the log return price.

After calculating the log return price of Bitcoin, the prediction results are obtained by combining the sentiment scores calculated in section 3.1.3 and feeding the two sets of data into the LSTM model.

4. Experiment Results

This section shows the application of the proposed methods in reality, including both the data acquisition and result prediction. Also, the results will be further analyzed.

4.1 Data Preparation and Cleaning

Bitcoin price data is collected from February 5, 2021 to July 12, 2022, divided by groups of 6 hours each. In the implementation, this paper uses the close price at the last moment to represent each 6-hour period. The log return price for each 6-hour period is then calculated by the method in Section 3.2.2.

Considering that Twitter is one of the most visited websites on the global Internet, it provides a platform for current global real-time events and hot topics to be discussed. The large number of Twitter users and the large number of tweets provide excellent data for analysis. In this paper, we select tweets containing the hashtags #bitcoin and #btc on the platform and analyze their sentiment to determine users' attitudes towards bitcoin.

At the beginning, four million tweets were collected from February 5, 2021 to July 12, 2022. Since the data is too large and requires a high level of computing power, this paper streamlines the tweet data in the following steps:

- (1) Restrict the platforms of tweets to: Twitter Web App, Twitter Android and Twitter iPhone.
- (2) Only English tweets are considered, thus excluding tweets in other languages.
- (3) Remove meaningless characters, spaces, links, etc. from tweets.
- (4) The tweets are grouped every 6 hours, and then the top 30 tweets with the highest number of favorites in each group are selected.

After the above restrictions, the number of tweets is reduced to 1.8 million.

4.2 Model Prediction

In the implementation, this paper utilizes the RoBERTa-base model to conduct the domain-specific pretraining and make the sentiment analysis on the collected tweet data. The total sentiment score of bitcoin related tweets is calculated according to equation 5 and 6.

The log return price of bitcoin and the sentiment scores of bitcoin related tweets are composed into a two-dimensional array into an LSTM model with the epoch set to 100, batch size set to 1. This paper employs the Mean Absolute Error (MAE) as the loss function, and leverages Adam as the optimizer. The model predicts the last 180 time points in the dataset (each time point is 6 hours apart) by using 80% of the data for training and 20% of the data for testing. The prediction is repeated 5 times. The RMSE (Root Mean Square Error) is chosen to measure the prediction error.

The model is in a multi-step manner and the step size can be transformed between 1 and 180. In practice, it is found that the best prediction results are obtained when the prediction step size is 70 (70 time steps are predicted at one time). The following table shows the RMSE of each step and the average RMSE of 5 times.

Table 1. RMSE of each step and average RMSE of 5 times

	1 Times	2 Times	3 Times	4 Times	5 Times	Average
t+70 RMSE	0.016531	0.016413	0.016447	0.016904	0.016584	0.016556
t+67 RMSE	0.016791	0.016690	0.016618	0.016671	0.016603	0.016671
t+71 RMSE	0.016725	0.016917	0.016525	0.016663	0.016727	0.016702
...	...	...	...	...	...	...
t+152 RMSE	0.046262	0.046263	0.046341	0.046352	0.046239	0.046290
t+154 RMSE	0.046265	0.046375	0.046286	0.046321	0.046328	0.046312
t+153 RMSE	0.046381	0.046511	0.046456	0.046285	0.046410	0.046403

4.3 Analysis of the Results

Fig. 3 displays the prediction results of the model.

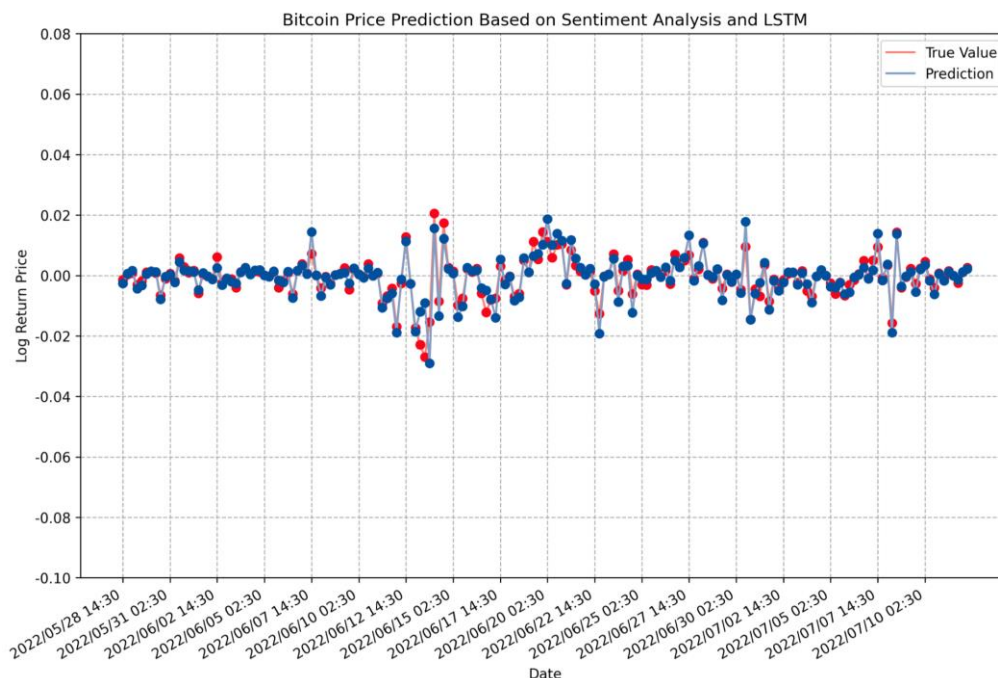


Fig. 3 Bitcoin price prediction based on sentiment analysis and LSTM

From the visualization results in Fig. 3, the RMSE of the model is 0.016556. Considering that the LSTM predicts the future log return price of bitcoin based on historical data, there is a certain lag effect in the model's prediction results, and the RMSE is slightly larger because the price of bitcoin is in a surge and plunge phase during the time period tested.

This paper also considers the case where only log return price is used for prediction. The results are shown in Fig. 4.

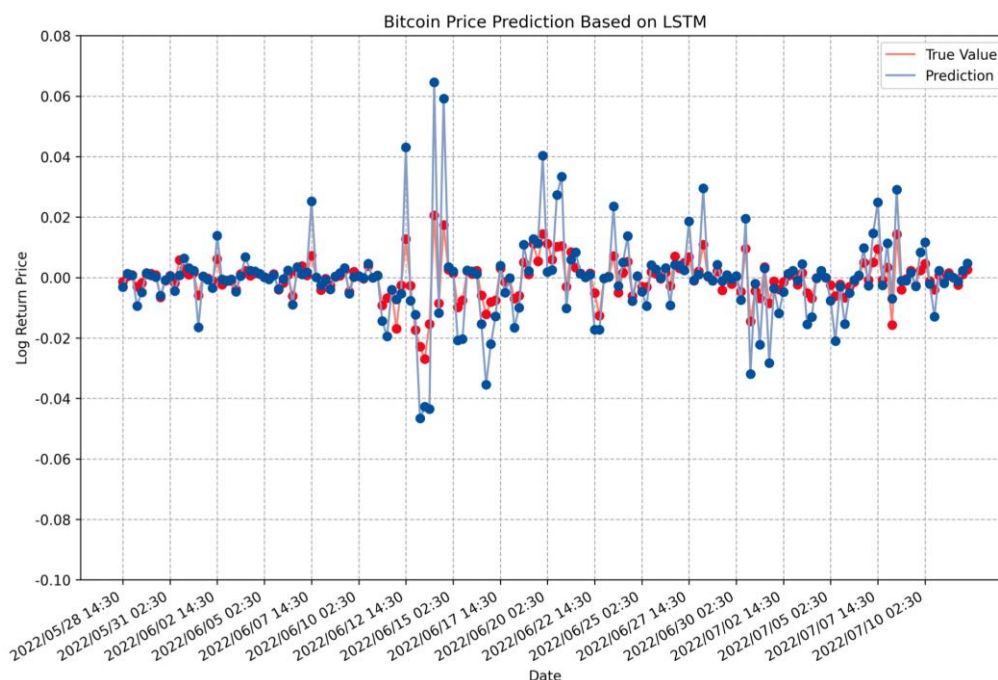


Fig. 4 Bitcoin price prediction based on LSTM

As can be seen in Fig. 4, the RMSE of this model is 0.044, which is slightly larger compared to the previous model. Fig. 4 also shows a large deviation of the prediction results from the true value. Clearly, the use of sentiment analysis can increase the prediction accuracy of the model, which

confirms that combining domain-specific sentiment analysis and temporal price of bitcoin to predict the rise and fall of bitcoin is the right approach.

5. Conclusion

This paper proposes to incorporate the sentiment and temporal information to make the price prediction of bitcoin. Firstly, to bridge the gap between the general domain of PLM and specific domain of bitcoin-related tweets, this paper proposes to conduct the self-supervised post-pretraining on the crawled domain-specific corpus. Specially, to capture the span-level semantics, this paper employs the span mask strategy during the pretraining stage. On the basis of the post-pretrained model, sentiment analysis on the tweets is conducted to obtain the scores. Secondly, this paper joint models the sentiment and temporal price data in the LSTM model. Combining both the calculation of the log return price and sentiment scores, this paper derives the final prediction results. Experiments show the superiority of the model after incorporating sentiment scores and temporal information.

In the future, it is worth considering optimizing the method of fusing the sentiment and temporal information. Since many tasks benefit from the pretrained transformer structures, it will be beneficial and meaningful to design a temporal transformer model to make better predictions.

References

- [1] Velde, F., Bitcoin: A primer. 2013.
- [2] Böhme, R., et al., Bitcoin: Economics, technology, and governance. *Journal of economic Perspectives*, 2015. 29(2): p. 213-38.
- [3] Balciar, M., et al., Can volume predict Bitcoin returns and volatility? A quantiles-based approach. *Economic Modelling*, 2017. 64: p. 74-81.
- [4] Shah, D. and K. Zhang. Bayesian regression and Bitcoin. in 2014 52nd annual Allerton conference on communication, control, and computing (Allerton). 2014. IEEE.
- [5] Georgoula, I., et al., Using time-series and sentiment analysis to detect the determinants of bitcoin prices. Available at SSRN 2607167, 2015.
- [6] Kristoufek, L., BitCoin meets Google Trends and Wikipedia: Quantifying the relationship between phenomena of the Internet era. *Scientific reports*, 2013. 3(1): p. 1-7.
- [7] Polasik, M., et al., Price fluctuations and the use of bitcoin: An empirical inquiry. *International Journal of Electronic Commerce*, 2015. 20(1): p. 9-49.
- [8] McNally, S., J. Roche, and S. Caton. Predicting the price of bitcoin using machine learning. in 2018 26th euromicro international conference on parallel, distributed and network-based processing (PDP). 2018. IEEE.
- [9] Stenqvist, E. and J. Lönnö, Predicting Bitcoin price fluctuation with Twitter sentiment analysis. 2017.
- [10] Barbieri, F., et al., Tweeteval: Unified benchmark and comparative evaluation for tweet classification. *arXiv preprint arXiv:2010.12421*, 2020.
- [11] Loureiro, D., et al., Timelms: Diachronic language models from twitter. *arXiv preprint arXiv:2202.03829*, 2022.
- [12] Xu, F., et al., MoCA: Incorporating Multi-stage Domain Pretraining and Cross-guided Multimodal Attention for Textbook Question Answering. *arXiv preprint arXiv:2112.02839*, 2021.
- [13] Muzammal, M., Q. Qu, and B. Nasrulin, Renovating blockchain with distributed databases: An open source system. *Future generation computer systems*, 2019. 90: p. 105-117.
- [14] Katsiampa, P., Volatility estimation for Bitcoin: A comparison of GARCH models. *Economics letters*, 2017. 158: p. 3-6.
- [15] Ozturk, S.S., Dynamic connectedness between bitcoin, gold, and crude oil volatilities and returns. *Journal of Risk and Financial Management*, 2020. 13(11): p. 275.
- [16] Duan, J., et al., A content-analysis based literature review in blockchain adoption within food supply chain. *International journal of environmental research and public health*, 2020. 17(5): p. 1784.

- [17] Guesmi, K., et al., Portfolio diversification with virtual currency: Evidence from bitcoin. *International Review of Financial Analysis*, 2019. 63: p. 431-437.
- [18] Wu, K., et al., A Coefficient of Variation Method to Measure the Extents of Decentralization for Bitcoin and Ethereum Networks. *Int. J. Netw. Secur.*, 2020. 22(2): p. 191-200.
- [19] Ramadhani, A.M., et al., Bitcoin Price Forecasting Using Neural Decomposition and Deep Learning. *Journal of the Korea Industrial Information Systems Research*, 2018. 23(4): p. 81-92.
- [20] Vaswani, A., et al., Attention is all you need. *Advances in neural information processing systems*, 2017. 30.
- [21] Xu, F., et al. Logiformer: A Two-Branch Graph Transformer Network for Interpretable Logical Reasoning. in *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2022.
- [22] Yun, S., et al., Graph transformer networks. *Advances in neural information processing systems*, 2019. 32.
- [23] Hochreiter, S. and J. Schmidhuber, Long short-term memory. *Neural computation*, 1997. 9(8): p. 1735-1780.