Forecasting the price of Bitcoin using Liquid Time-stochasticity Networks

ABSTRACT

Despite the significant progress made in Deep Learning (DL) in areas such as Natural Language Processing (NLP) and Reinforcement Learning (RL), Time Series (TS) forecasting has yet to witness a comparable breakthrough due to inherent algorithmic limitations. Consequently, systems developed using these methods are constrained in forecasting performance.

Researchers have proposed various approaches to address this challenge, with neural Ordinary Differential Equations (ODEs) and Liquid Time-constant (LTC) networks among the most promising. While neural ODEs introduce the concept of continuous-time and depth models, their performance has been underwhelming compared to traditional neural networks like Long Short-Term Memory (LSTM). On the other hand, LTC networks have shown impressive results, but their outdated architecture of ODEs limits their usability and performance. This paper introduces the novel **Liquid Time-stochasticity** (LTS) algorithm that combines the adaptability of LTC networks with Stochastic Differential Equations (SDEs), that achieved state-of-the-art performance in TS forecasting, as demonstrated by the highly popular Bitcoin (BTC) price prediction problem. Applying LTS on a BTC price forecasting problem yielded a promising result, with a percentage error of **2.5% ± 0.1**.

KEYWORDS

Time Series (TS) forecasting, Liquid Time-constant (LTC) networks, Liquid Time-stochasticity (LTS) networks, Stochastic Differential Equations (SDEs)

1 Introduction

1.1 Time series forecasting

TS forecasting is a critical business concern, and an area ML could significantly influence. It forms the basis of crucial domains such as customer management, inventory control, marketing, and finance. Correspondingly, even slight enhancements in forecasting accuracy can have a substantial financial impact, potentially worth millions of dollars [1].

Despite the successes of ML and DL in NLP and computer vision, TS forecasting remains a challenge compared to classical statistical methodologies [2]. In fact, in the M4 competition, ensembles of traditional statistical techniques dominated the top-ranking methods. In contrast, traditional ML methods were not competitive [3]: the winner was a hybrid model of LSTM and classical statistics [4], who concluded that the only way to improve TS forecasting accuracy was through hybrid models. This paper seeks to challenge this conclusion.

1.2 Cryptocurrencies & bitcoin

Cryptocurrency, especially BTC, has been a widely discussed topic recently - it has even come to the point where crypto and BTC are used interchangeably. Cryptocurrencies are a fully decentralized digital currency form that operates using a peer-to-peer system without needing a third party, thus enabling safer online transactions [5]. BTC is the first and most popular digital currency, piquing many academic researchers' interest [6]. Recent advances in ML and statistics have shown acceptable results in analyzing and predicting cryptocurrencies. However, these algorithms are still static, and BTC's high volatility challenges investors [7].

1.3 Liquid Time-stochasticity networks

LTS networks are SDE-based Liquid neural networks proposed by [1], an enhancement to the Liquid Time-constant (LTC) network proposed by [8]. These algorithms exhibit greater flexibility with the ability to adapt to instantaneous and minute changes, that are mostly common in noisy and volatile TS data [1]. In order to efficiently predict and break through limitations of TS forecasting where the future is uncertain and data stream characteristics are bound to change, the LTS - proposed by the authors of this paper a few months ago - is an ideal way forward that provides promising characteristics.

2 Review of related work

2.1 Bitcoin forecasting

BTC forecasting has experienced exponential growth in the past decade, with many websites making historical data publicly available to satisfy stakeholders' need for forecasting models. As a result, research has focused on developing forecasting models to predict BTC prices accurately.

Paper [9] proposed using the LSTM to learn patterns in BTC's historical closing prices to predict future prices. While the approach was shown to be effective in producing accurate one-day predictions, its simplicity raises concerns about its long-term sustainability as a forecasting model. This is because the BTC market is an open and complex system influenced by many factors beyond the closing price. Therefore, while the study's results are promising, more sophisticated forecasting models are needed to capture the multifaceted nature of the BTC market.

Paper [10] suggested using the Prophet model by Facebook to distinguish it from ARIMA. They found that other features in the dataset (such as open price and volume) were not meaningful for predictions. After removing these features, the authors added a fiat currency price feature, which strongly correlates with BTC prices, resulting in more accurate forecasts.

2.1.1 Factors that affect the rate of Bitcoin

- A study conducted by [11] found a strong correlation between the number of deaths from COVID-19 and BTC price.
- Tweet sentiment - identified by [12].
- Tweet volume - identified by [13] & [14].
- Google Trends – identified by [13].

2.1.2 Overall reflection of bitcoin forecasting solutions

The above-proposed solutions have limitations in that they do not consider the impact of external factors on the price of BTC. This can be a concern as the cryptocurrency market is highly volatile, and even a single tweet can cause significant price fluctuations. While external factors such as country legislation and laws may be beyond researchers' control, it is crucial to consider those that can be accounted for, including sentiment analysis of tweets, tweet volume, and Google Trends data. However, it is important to note that these factors alone may not always provide accurate predictions as they are subject to change rapidly. Therefore, to build robust forecasting systems, it is essential to consider and account for all relevant external factors that could impact the price.

Paper [13] suggested incorporating Twitter and Google Trends data into forecasting models. They discovered that tweet volume and Google Trends were highly correlated with the BTC price, making them important factors to consider. However, they noted that tweet sentiment might not be reliable because tweets about cryptocurrencies tend to be objective. The authors employed a linear model for their predictions, but other studies have demonstrated that non-linear models can yield better performance [15].

2.2 Bitcoin Twitter analysis

As identified above, adding external factors can improve performance drastically, where some of the more impactful factors are Twitter sentiments and tweet volumes. However, creating such models is more complex than including them as another feature in the dataset. Several issues still need solving when using tweet data as a factor in crypto price predictions [16]. Some of them are detailed below:

- Bots often duplicate and automate tweets [17].
- They are often noisy (ex: hashtags, profile mentions, URLs).
- Sarcasm can affect sentiments [18].

Therefore, the data must first be preprocessed with such challenges in mind. A point worth noting is that [17] identified that more than Twitter data is needed for the prediction of BTC, but it can be helpful when used in conjunction with other data.

Paper [19] developed a model for BTC price prediction by incorporating Twitter sentiment analysis. They found a moderate correlation between Twitter sentiment and BTC price and integrated the sentiment scores of tweets with historical data to forecast future prices. However, as [13] pointed out, tweet volume is more influential than tweet sentiment in determining the price of BTC, which [19] did not account for.

Paper [20] explored statistical and DL models for the predictions utilizing sentiment in the BTC ecosystem. They identified that rather than financial features, the sentiment has a more significant impact on the overall price, which further justifies that the usage of the volume and open features do not have such significance. They justified that the tweet sentiment and BTC price produced the best performance. The issue here is: although the performance obtained is excellent, the implementation is not robust since, as mentioned by [17], the Twitter data alone is insufficient, and, as identified by [13], tweets can be objective.

2.2.1 Overall reflection of bitcoin Twitter analysis

The drawbacks of the above two solutions are that tweets are objective, can carry sarcasm, and can be duplicated or generated by bots. Therefore, utilizing the tweet sentiment solely is not robust. Hence, the optimal solution would be to utilize additional features as [13] proposed alongside a non-linear model, as identified by [15].

2.3 Time series forecasting

2.3.1 Neural ordinary differential equations

Neural ODEs are a novel DL model class with continuous-depth and constant memory costs. These models are known for their speed but may sacrifice numerical accuracy. One of their distinguishing features is their ability to adapt their evaluation methods based on input changes [21].

In the field of TS forecasting, the ability to adapt to incoming data streams may be a useful feature of neural ODEs. However, it is worth noting that these models may lack robustness and are not well-equipped to detect uncertainty in predictions, as noted by [22].

2.3.2 Liquid Time-constant networks

A remedy was put out by [8] to boost the neural ODEs' disappointing performance. The formulation shows 'liquid' qualities that lead to superior performance for TS predictions. The proposed architecture reveals stable and bounded behavior and is extremely expressive, enabling it to capture the dynamics in TS data better.

Despite the promising results of the LTC architecture, it is important to note that it uses ODEs, which are considered outdated by researchers [23]. Additionally, LTC models may not be suitable for rapidly changing data streams as they cannot quickly adapt to sudden changes in input.

3 Methodology of implementation

As identified, incorporating as many possible factors contributes to the solutions robustness; therefore, this paper combined the following factors with historical prices to produce a solution with the maximum possible accuracy and performance.

- Block reward size, Twitter volume & Google Trends
- Bitcoin tweets

3.1.1 Data collection

To ensure an on-demand and self-updating system, a fixed CSV dataset cannot be used; instead, an API must be queried to ensure that the available data is up-to-date.

- The historical prices were fetched from http://api.scraperlink.com/investpy/ which provides free access to the required historical prices.
- The block reward, Twitter volume and Google Trends were scraped from https://bitinfocharts.com/ using a simple Python script.
- Obtaining tweets provided multiple issues as Twitter's API has been recently extremely restricted. Therefore, the research had to limit to using a total of 500 tweets for a particular day rather than obtaining every single tweet.

It is essential to ensure that the data was obtained through APIs or updated dashboards so that the system remains updated.

3.1.2 Data preprocessing

Data fetched from the data fetchers must undergo preprocessing before being used by the model. Traditional techniques of normalization and cleaning was performed prior to the data combination. The historical prices, block reward, Twitter volume, and Google Trends are all numerical values; therefore, they had to undergo normalization to ensure values are in the same range and no bias is introduced.

The tweets underwent sentiment analysis via the VADER sentiment analyzer [24] as it is tailored for sentiment analysis of Twitter tweets. The compound score is weighted beforehand by utilizing a novel sentiment weighting formula described below.

$$influencer_{sum} = \alpha \log_{10}(followers_{count} + 1) + \beta \log_{10}(lists_{count} + 1)$$

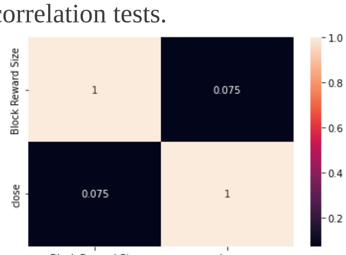
$$tweet_{sum} = \gamma \log_{10}(retweets_{count} + 1) + \delta \log_{10}(like_{count} + 1)$$

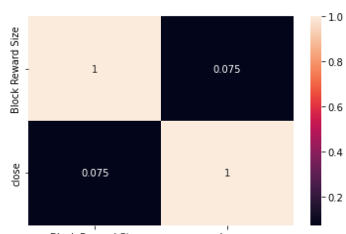
$$\boxed{weighted_{score} = \frac{tweet_{sum} + influencer_{sum}}{tweet_{sum} + influencer_{sum} + 1} * compound_{score}} \quad (1)$$

Where;

α	Weight of number of followers. **Set as 0.5**	γ	Weight of number of retweets. **Set as 0.1**
β	Weight of number of lists. **Set as 0.3**	Δ	Weight of number of likes. **Set as 0.1**

Upon these preprocessing steps, feature selection is performed via the Pearson correlation test, to filter out only features that have an impact on the price to be forecasted. The images below illustrate the results of the correlation tests.





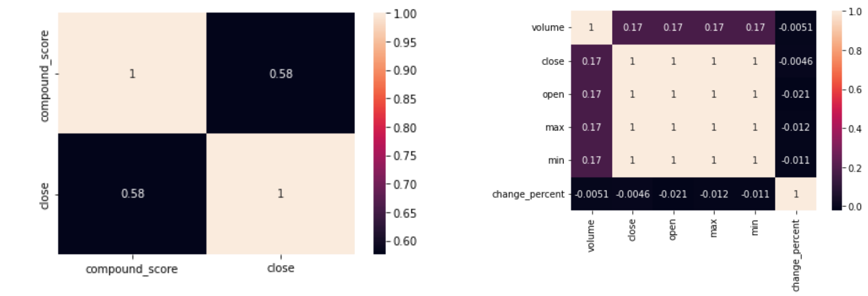


Figure 1: Correlation test results of exogenous features

The above results demonstrates that the exogenous features can be considered due to a relatively lower correlation value; however, the close, volume, open, max, min and change_percent features must be excluded.

3.1.3 The LTS algorithm

Two types of models were implemented: univariate and multivariate. The multivariate single-horizon forecasting model combined the above-mentioned data sources, while the univariate multi-horizon forecasting model solely relied on historical data

The LTS algorithm proposed by [1] was constructed as a Keras layer; therefore, it can be integrated into a Keras model to be used in implementation. Several LTS-based models were built and trained using the Adam optimizer and added to an ensemble list. The upper-bound, lower-bound, and point forecasts are generated from this list by extracting the highest, lowest, and median forecasts, respectively – for each of the two types of models described in the previous paragraph.

4 Testing and evaluation

4.1.1 Testing

Two ensemble models were implemented: univariate and multivariate. Both models were tested similarly by setting a ‘pseudo-future’ from a specific timeframe. The below graphs illustrate the closing price change with the actual and predicted prices.

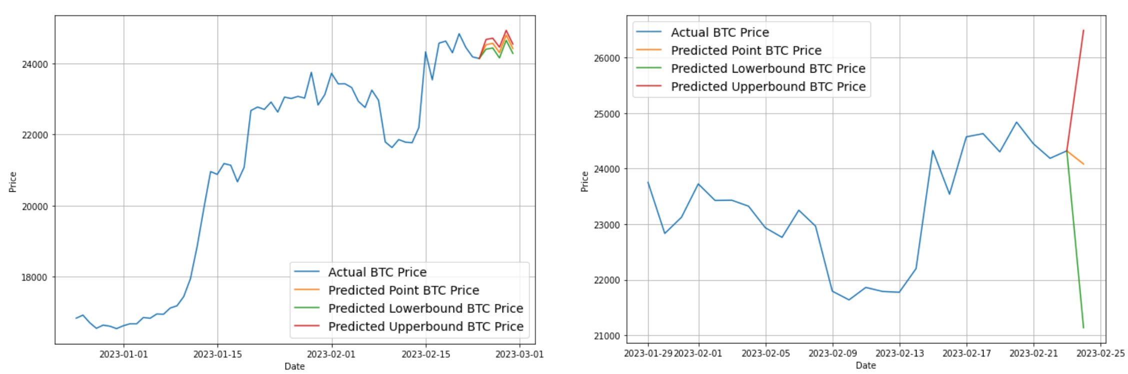


Figure 2: Model testing results: Univariate & Multivariate

Thirty different models were trained and combined to form the ensemble, resulting in 3 predicted prices. The median prediction is used as the predicted point BTC price, while the highest and lowest predictions are used [1]to create the upper and lower bounds, respectively.

[1] The LTS algorithm repository can be found at https://github.com/Ammar-Raneez/Liquid-Time-stochasticity

4.1.2 Evaluation

Standard prediction and test set evaluation techniques were applied to both univariate and multivariate ensembles to evaluate the models' performance. The results were compared with the naïve forecast, which was chosen as a benchmark due to its reputation of being notoriously difficult to beat in open market systems [25]. In addition, other architectures were also evaluated during the univariate phase to ensure that the ensemble architecture was the best performing.

Table 1: Univariate model evaluation

	MAE	MSE	RMSE	MAPE	MASE
Traditional architectures					
Basic dense	1227	2882849	1697	3.06%	1.07
2x dense	1146	2628006	1621	2.98%	1.05
Stacked dense	1147	2604766	1613	2.88%	1.04
Conv1D	1153	2653370	1628	2.90%	1.02
LSTM	1216	2834756	1683	3.07%	1.06
N-BEATS	1142	2614896	1617	2.86%	1.03
Benchmark & LTS Ensemble					
Naïve forecast	951	2021966	1421	2.56%	1.00
Ensemble	950	2013928	1419	2.56%	0.99

As the performance of traditional architectures was not satisfactory, evaluating them for the multivariate model was deemed unnecessary.

Table 1: Multivariate model evaluation

	MAE	MSE	RMSE	MAPE	MASE
Naïve forecast	858	1631648	1277	2.41%	1.00
Ensemble	932	1826788	1351	2.67%	1.00

4.1.3 Overall reflection of the evaluation

Based on the above evaluation, the multivariate model performs better than the univariate model, attributed to its ability to incorporate multiple exogenous features that provide more information and context for accurate predictions. This makes the model more robust and less reliant on a single input variable, resulting in improved performance. It can also be observed that the naïve forecast performs better in the multivariate model, whereas the opposite is true for the univariate model. This difference in performance could be attributed to the increased complexity of the ensemble architecture used in the multivariate model, as it can add more noise, especially since a more diverse dataset is used, which must be addressed as enhancements in the future.

The application code can be found at https://github.com/Ammar-Raneez/BitForecast

5 Conclusion and future work

In this paper, the authors proposed using the novel LTS algorithm to forecast the price of bitcoin with the intention of obtaining a greater performance improvement, due to the excellent characteristic of being able to adapt to noisy and volatile environments. The results obtained supported this claim as the performance obtained was greater when compared to other neural networks that are supposedly the most state-of-the-art in time series forecasting.

Additionally, to further create the most robust solution possible, the authors included as many exogenous features possible to ensure that the system does not act immature.

As future work, researchers must look into enhancing the Twitter sentiment weighing formula to incorporate more factors. However, it is essential that factors that intuitively do not have an impact must not be included, as this would reduce the accuracy and effectiveness of the formula. Furthermore, other techniques should be researched on so that more than 500 tweets can be obtained for a particular day.

ACKNOWLEDGMENTS

To the developers behind https://bitinforcharts.com and the *snscrape* Python library for enabling a more seamless data scraping experience.

REFERENCES

[1] A. Raneez and T. Wirasingha, "A Review On Breaking the Limits of Time Series Forecasting Algorithms," in 2023 IEEE 13th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA: IEEE, Mar. 2023, pp. 0482–0488. DOI: 10.1109/CCWC57344.2023.10099071.

[2] S. Makridakis, E. Spiliotis, and V. Assimakopoulos, "The M4 Competition: Results, findings, conclusion and way forward," International Journal of Forecasting, vol. 34, no. 4, pp. 802–808, Oct. 2018, DOI: 10.1016/j.ijforecast.2018.06.001.

[3] S. Makridakis, E. Spiliotis, and V. Assimakopoulos, "Statistical and Machine Learning forecasting methods: Concerns and ways forward," PLOS ONE, vol. 13, no. 3, p. e0194889, Mar. 2018, DOI: 10.1371/journal.pone.0194889.

[4] S. Smyl, "A hybrid method of exponential smoothing and recurrent neural networks for time series forecasting," International Journal of Forecasting, vol. 36, no. 1, pp. 75–85, Jan. 2020, DOI: 10.1016/j.ijforecast.2019.03.017.

[5] S. Nakamoto, (2020). Bitcoin: A peer-to-peer electronic cash system. Available from https://bitcoin.org/bitcoin.pdf.

[6] M. Rahouti, K. Xiong, and N. Ghani, "Bitcoin Concepts, Threats, and Machine-Learning Security Solutions," IEEE Access, vol. 6, pp. 67189–67205, 2018, DOI: 10.1109/ACCESS.2018.2874539.

[7] I. sibel Kervanci and F. Akay, "Review on Bitcoin Price Prediction Using Machine Learning and Statistical Methods," Sakarya University Journal of Computer and Information Sciences, Dec. 2020, DOI: 10.35377/saucis.03.03.774276.

[8] R. Hasani, M. Lechner, A. Amini, D. Rus, and R. Grosu, "Liquid Time-constant Networks." arXiv, Dec. 14, 2020. DOI: 10.48550/arXiv.2006.04439.

[9] J. P. Fleischer, G. von Laszewski, C. Theran, and Y. J. Parra Bautista, "Time Series Analysis of Cryptocurrency Prices Using Long Short-Term Memory," Algorithms, vol. 15, no. 7, p. 230, Jul. 2022, DOI: 10.3390/a15070230.

[10] I. Yenidogan, A. Cayir, O. Kozan, T. Dag, and C. Arslan, "Bitcoin Forecasting Using ARIMA and PROPHET," in 2018 3rd International Conference on Computer Science and Engineering (UBMK), Sarajevo: IEEE, Sep. 2018, pp. 621–624. DOI: 10.1109/UBMK.2018.8566476.

[11] S. A. Sarkodie, M. Y. Ahmed, and P. A. Owusu, "COVID-19 pandemic improves market signals of cryptocurrencies–evidence from Bitcoin, Bitcoin Cash, Ethereum, and Litecoin," Finance Research Letters, vol. 44, p. 102049, Jan. 2022, DOI: 10.1016/j.frl.2021.102049.

[12] Y. B. Kim et al., "Predicting Fluctuations in Cryptocurrency Transactions Based on User Comments and Replies," PLoS ONE, vol. 11, no. 8, p. e0161197, Aug. 2016, DOI: 10.1371/journal.pone.0161197.

[13] J. Abraham, D. Higdon, J. Nelson, and J. Ibarra, "Cryptocurrency Price Prediction Using Tweet Volumes and Sentiment Analysis," SMU Data Science Review, vol. 1, no. 3, Aug. 2018, [Online]. Available: https://scholar.smu.edu/datasciencereview/vol1/iss3/1.

[14] D. Shen, A. Urquhart, and P. Wang, “Does twitter predict Bitcoin?,” Economics Letters, vol. 174, pp. 118–122, Jan. 2019, DOI: 10.1016/j.econlet.2018.11.007.

[15] M. Maiti, Y. Vyklyuk, and D. Vuković, “Cryptocurrencies chaotic co‐movement forecasting with neural networks,” Internet Technology Letters, vol. 3, no. 3, May 2020, DOI: 10.1002/itl2.157.

[16] J. V. Critien, A. Gatt, and J. Ellul, “Bitcoin price change and trend prediction through twitter sentiment and data volume,” Financ Innov, vol. 8, no. 1, p. 45, Dec. 2022, DOI: 10.1186/s40854-022-00352-7.

[17] F. Valencia, A. Gómez-Espinosa, and B. Valdés-Aguirre, “Price Movement Prediction of Cryptocurrencies Using Sentiment Analysis and Machine Learning,” Entropy, vol. 21, no. 6, p. 589, Jun. 2019, DOI: 10.3390/e21060589.

[18] S. Rosenthal, A. Ritter, P. Nakov, and V. Stoyanov, “SemEval-2014 Task 9: Sentiment Analysis in Twitter,” in Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), Dublin, Ireland: Association for Computational Linguistics, 2014, pp. 73–80. DOI: 10.3115/v1/S14-2009.

[19] D. R. Pant, P. Neupane, A. Poudel, A. K. Pokhrel, and B. K. Lama, “Recurrent Neural Network Based Bitcoin Price Prediction by Twitter Sentiment Analysis,” in 2018 IEEE 3rd International Conference on Computing, Communication and Security (ICCCS), Kathmandu: IEEE, Oct. 2018, pp. 128–132. DOI: 10.1109/CCCS.2018.8586824.

[20] G. Serafini et al., “Sentiment-Driven Price Prediction of the Bitcoin based on Statistical and Deep Learning Approaches,” in 2020 International Joint Conference on Neural Networks (IJCNN), Glasgow, United Kingdom: IEEE, Jul. 2020, pp. 1–8. DOI: 10.1109/IJCNN48605.2020.9206704.

[21] R. T. Q. Chen, Y. Rubanova, J. Bettencourt, and D. Duvenaud, “Neural Ordinary Differential Equations.” arXiv, Dec. 13, 2019. DOI: 10.48550/arXiv.1806.07366.

[22] S. Anumasa and P. K. Srijith, “Latent Time Neural Ordinary Differential Equations,” AAAI, vol. 36, no. 6, pp. 6010–6018, Jun. 2022, DOI: 10.1609/aaai.v36i6.20547.

[23] Duvenaud, D (2021). Directions in ML: Latent Stochastic Differential Equations: An Unexplored Model Class. YouTube. Available: https://www.youtube.com/watch?v=6iEjF08xgBg.

[24] C. Hutto and E. Gilbert, “VADER: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text,” Proceedings of the International AAAI Conference on Web and Social Media, vol. 8, no. 1, pp. 216–225, May 2014, Accessed: Oct. 19, 2022. [Online]. Available: https://ojs.aaai.org/index.php/ICWSM/article/view/14550

[25] Gilliland, M. (2014). A naive forecast is not necessarily bad. The Business Forecasting Deal. Available: https://blogs.sas.com/content/forecasting/2014/04/30/a-naive-forecast-is-not-necessarily-bad/.