www.editada.org

International Journal of Combinatorial Optimization Problems and Informatics, 16(1), Jan-April 2025, 164-176. ISSN: 2007-1558.
https://doi.org/10.61467/2007.1558.2025.v16i1.552

Optimizing Bitcoin Price Prediction: Multivariate LSTM Triumphs

Brian Scanlon, Keith Quille, Rajesh Jaiswal

[1] School of Enterprise Computing and Digital Transformation, TU Dublin, Ireland
Emails: brianscanlon.academic@gmail.com, keith.quille@tudublin.ie, rajesh.jaiswal@tudublin.ie

Abstract. This study aims to assess the validity and precision of employing a multivariate LSTM model compared to traditional models and stock analysis techniques for predicting the price of the cryptocurrency BTC. The research incorporates a feature elimination technique to optimize price predictions across various time intervals by removing non-essential and redundant features, including economic factors. In the case of BTC, with a finite total supply of 21 million coins, an increase in popularity generally leads to a surge in price. To gauge BTC's popularity, tweet frequency and Google search trends were considered as input factors. Additionally, traditional indicators like USD, Gold and the Volatility Index (VIX) were used to measure the stock market atmosphere. The LSTM model's performance was benchmarked against other models such as RNNs, ANN, SVR and ARIMA. The LSTM model exhibiting superior learning in multivariate data, achieving an RMSE score of 268.83.

Keywords: Bitcoin, LSTM, Deep-Learning, Feature Selection, Stock Price Prediction.

Article Info
Received November 13, 2024
Accepted March 12, 2025

1 Introduction

In recent years, cryptocurrency has grown in popularity as new companies or 'coins' are constantly appearing because of the emergence of blockchain technology. BTC is digital, so these coins are not physical, they are online. There has been a significant increase in traffic towards these coins in the financial world recently, especially Bitcoin (BTC), the most popular coin with the largest market cap in the cryptocurrency market at the time of writing (Investopedia, 2021). Many factors can influence prices, and this type of stock is highly volatile compared to more traditional 'NYSE' companies (Klein et al., 2018).

Stock price prediction involves forecasting a stock's future value based on its historical price performance and/or the fundamental aspects of the company. These fundamentals can consist of financial items such as debt, revenue, operating income, etc. (Alberg & Lipton, 2017). This technique is used for investors to invest in these companies based on this analysis to take profit from their stock trading; trading can be split into two main methods:

1. Fundamental Analysis: Makes the trading decision based on the company's future trend based on the fundamentals. This method might utilize public information such as market news, quarterly results or the general industry trend.
2. Technical Analysis: Takes the human element out of trading and uses mathematical models to make the decisions based on the analysis of a company's price chart trends. This might include information such as closing price, opening price, highest and lowest price throughout the trading period (can be 5 seconds, 15 minutes, 1-day etc. intervals).

Often, when modelling time series regression problems such as stock price prediction, Machine & Deep Learning modelling techniques are utilized to predict future performance based on past values. Traditionally, methods such as ARIMA and Linear Regression have been used in this area. However, this paper will investigate if more newly developed Deep Learning techniques, such as LSTM, perform better with equivalent inputs. The variables used will also be investigated during feature analysis as there are a plethora of features explored, with some being more traditional methods for stock prediction, i.e. technical analysis-related features. Other features are more fundamentally analysis-related, which is not a typical stock market time series prediction technique. By exploring these methods, this study seeks insights into the effectiveness of features on the models and identifies the best permutation among them. Consequently, the research objectives are as follows:

- Research Objective 1: Investigate feature selection/importance techniques to determine the optimal number of features for Bitcoin price prediction
- Research Objective 2: Predict the closing price of BTC using multivariate features as inputs to the LSTM model, gauging the model's performance versus other traditional models.

This paper is structured in the following format: Section 2 comprises the literature review processing previous work completed on similar topics. Section 3 focuses on the methodology, detailing the processes used in data selection and acquisition. This section also explains the criteria for algorithm selection, as well as the procedures involved in determining topologies, hyperparameters, and variables during model development. Section 4 displays the results obtained from the various implemented models, and finally, Section 5 concludes the paper by drawing insights and conclusions from the results.

2 Literature Review

One challenge in predicting stock prices lies in the non-stationary behaviour of their properties (Kazem et al., 2013). Unbiased estimation of time series data, such as stock prices, cannot always be reliably forecasted by well-known linear techniques. The stock market accumulates large and highly non-linear data, necessitating more advanced time series prediction algorithms.

Typically, tweet sentiments on a given day possess predictive power for the returns of BTC and other cryptocurrencies; this phenomenon is well demonstrated through the NLP models in (Kraaijeveld & De Smedt, 2020). Tweet volumes, sentiment analysis, and Google Trends data related to Bitcoin can be successfully used to construct a BTC price prediction model (Abraham et al., 2018). For this the data was gathered from the Google trends website (Google, n.d.) and Bitinfo for X volume (Bitinfo, 2009), as the API has exceptions to the amount of data gathered. The authors compared the correlations using Pearson's coefficient and found that sentiment analysis of tweets was not a reliable indicator for the change in BTC price. However, unlike the sentimental analysis, Google Trends and tweet volume were highly correlated with price; they also found these correlations to hold for negative and positive trends. Similarly, it was found that the number of tweets of BTC is a significant driver of next-day trading volume and realised volatility for BTC (Shen et al., 2019).

Data can be purchased for minute (price and volume) trading of the S&P 500 ETF (SPY); this can then be divided into train, development and test (Libman et al., 2019). An LSTM was used on its own as well as hybrid models (combining LSTM with algorithms such as support vector regression, or auto regression, where the algorithm prediction would feed into the feature set of the input to the LSTM model). The results showed that the use of the LSTM model contributed to a superior prediction in volume change, especially when coupled with other algorithm's predictions in the feature input layer.

The normal technical indicators like close price in their study along with some corporate accounting statistics like profit margin and debt to equity ratio were used in this research (Zou & Qu, 2020). They used the top 10 stocks in the S&P 500 in their study, getting their values from 2004 to 2013, forward filling any missing statistics between two realising dates. They applied the min-max scale to normalise the data and used a ratio of 70-15-15 for train, development and testing. They used normal LSTM model, a stacked LSTM, and the ARIMA model. These models fed into an investing strategy which turned out to be quite profitable, interestingly the stacked LSTM model did not perform better than the normal LSTM model, due to the potential of overfitting.

Ta-Lib was used to generate 175 extra price indicators from their data obtained from different companies in the Brazilian stock exchange (Nelson et al. 2017). In total, 180 features would be used as the input to the LSTM here, they also used a 'sliding window' approach. They generated a new neural network at the end of each trading day, updating the weights with the new training/validation data. It used the previous 10 months of data for training and the performance was validated by using the past week's data. They also used other industry baseline stock prediction methods in the Ta-Lib library to validate the model's performance. Despite the size of the input dimension, feature selection methods were not needed here. The authors found that the model performed well in terms of accuracy, and while it may not have captured the highest profits every time there was a 'signal', it did maintain a positive portfolio compared to the other baseline methods.

LSTM, RNN and CNN-sliding window model can be utilised for stock price prediction (Selvin et al., 2017). The data was minute-wise stock prices for 1,721 NYSE companies from July 2014 to July 2015, consisting of information like day or time stamp, transaction ID, stock price and volume. A sliding window approach was used here for a short-term future prediction. The window size was 100 minutes with an overlap of 90 minutes information, and prediction was made for 10 minutes in the future. The error for different window sizes was gotten to calculate the best window length. 1,000 epochs were used to train the models by varying layer size for tuning, if the MSE was less than the preceding epoch then the weight matrices were updated. It was found that RNN

& LSTM models were more susceptible to not accurately catching the dynamic changes, as the system patterns will not always be the same. However, they performed significantly better than the ARIMA model.

A model on six stocks in the Shanghai composite index which consisted of two parts. An emotional analysis model (based on Naïve Bayesian) which analyzed sentiment of posts in a popular stock forum and a LSTM time series learning model, which used the stock price data (Zhuge et al. 2017). The authors found that using emotional data and the technical indicators, they were able to create a model which was better at predicting the price than one just using the technical indicators. The LSTM based model performed better than its RNN and NLP counterparts, as stock data exists long-term dependence, therefore the LSTM model increased accuracy.

A hybrid of a convolutional layer and a recurrent layer (LSTM) was used to get the semantics of news articles along with price indicators as inputs to their model (Vargas et al., 2017). CNN was used to get the semantics of news titles the day before and the technical indicators were also fed into their own LSTM, these worked side by side as two inputs into this hybrid model. The last stage is a traditional fully connected layer with softmax as an activation function that outputs a binary label, 1, 0 for an increase in stock price, 0, 1 for a decrease. The hybrid RCBB architecture used here is better performing than just CNN, with both sources of information (news and technical indicators) being relevant and it is found that information in news articles has a short temporal effect in the financial market.

From the above it can be concluded that researchers have had varying degrees of success in terms of performance overall and versus other, and sometimes more complex models. LSTM models possess the ability to predict changes in time series models based on historical data, given the right inputs they can be very successful. While many researchers have used semantics of tweets as inputs to varying degrees of success, this research will employ the use of fundamental analysis such as X mentions and Google searches, while also using traditional safe havens such as Gold price, and then stocks such as the Volatility index (VIX) that show the trends in the market. This research will use some of the model development methodologies in the aforementioned literature review, combined with more traditional technical indicator inputs for Bitcoin, and coupling this with the fundamental analysis inputs. This will potentially lead to a better-performing model overall to predict the price of BTC.

3 Methodology

3.1 Data Selection

A combination of technical indicators and fundamental analysis was utilized for this research. The technical inputs consisted of the traditional open/close/volume/high/low indicators for the BTC stock used in this dataset. In stock analysis, traders tend to use patterns seen from technical indicators over some time to predict what the next movement of the stock price will be. This time series analysis is normally performed on just the stock that is being analyzed.

Bitcoin is a digital currency, and often referred to as digital gold and a hedge against the USD. Gold is a commodity stock itself, which the market often looks to in times of stock market volatility. The Volatility Index (VIX) is a gauge of stock market volatility. This index is tracked as a stock price and is generally used as an indicator of the temperature of the market as seen by financial professionals. If the price of gold or VIX increases, there tends to be an increase in overall market volatility to coincide with it. For the purposes of this project Gold, USD and VIX stock close prices were the other technical indicators included as these are good indicators of volatility in the overall stock market.

The fundamental analysis collected is the trends on Google or X for mentions of the phrase "Bitcoin", as this would give a good indication of market news. Market news is qualitative data and is hard to transform into data usable for a model as it tends to be non-structured in nature and can generally be hard to quantify and converted into a format suitable for a Deep Learning model. To combat this, Google & X data would be transformed into numeric data by encoding Google searches or tweets containing “Bitcoin” into a numeric value for a given day’s tweet volume containing these phrases. These market trends or popularity measures were now represented as quantitative data to allow for processing by the models. There will only ever be a limited number of BTC ‘coins’ in the world, a maximum of 21 million to be exact at the time of writing, this feeds into the logic for using X / Google searches (Rotman, 2014). The more people that are talking about it or adopting it, the more the price will go up due to people trying to buy it, simple supply and demand logic (Investopedia, 2021). Outside of actual trading forums X is a good place to capture news and/or talk of cryptocurrency and generally speaking rapid developments can be caught well on this platform.

At the conclusion of the above data selection, the resulting 10 variables at the outset of model development were the following, these features were deemed as potentially having a positive impact on model performance for predicting BTC price prior to model development; Open, High, Low and Close BTC prices, BTC Trading Volume, Gold price, USD price, VIX price, Google Trends and X Mentions, 10 variables in total to predict the target variable. These features will be analyzed during the model building process, most notably their contribution to model performance will be examined and different groups of features used to gauge their importance to model score improvement.

3.2 Data Acquisition

The time stamps will be one day intervals, different methods were used to acquire each input variable.

Technical Price Data: The Yahoo finance library API was utilized to make a call using a stock's ticker symbol to retrieve its historical data from the site (Yahoo Finance, 2024). A ticker symbol is a unique set of characters assigned to a stock for trading purposes, i.e., X is 'TWTR' (Investopedia, 2023). The BTC, Gold, USD and VIX price history (open, close, high, low, volume) were retrieved using this method. All the indicators were kept for BTC, only the close values were then kept for Gold, USD and VIX prices.

Google Trends Data: The interest over time function was employed from the Pytrend python library for the search topic "Bitcoin" over the time period previously mentioned (DeWilde, n.d.). Google Trends normalizes the data for a given range between 1 and 100 which wasn't an issue for this project as the data would have to be normalized pre input to the model (Rafiq et al., 2001).

X Tweets Data: The Bitinfo site needed to be used as the X API is limited because it doesn't allow the user search back more than two weeks in data. X API works by making a call to the search engine given the input parameters, which consisted of dates for search period, and "Bitcoin" or "BTC" for search terms as seen in Table 1. These are then summed up and incremented per day to give a figure for the amount of tweets that day which had the term "Bitcoin" mentioned in them, giving an accurate indicator of the tweeting volume. Bitinfo website was used for the initial historical data grab to manually input the X volumes per day for the aforementioned dates. The X API could then be used to update the mentions on a daily basis as needed. The API keys used are shown in Table 1.

Table 1. X (Tweets) API Keys

Key	Item Used
API Key	Access Token & Consumer Keys
Q (search term)	Bitcoin, BTC
Since (date)	Start Date
Until (date)	End Data

3.3 Data Pre-Processing

BTC is a 24/7 traded cryptocurrency, therefore the data for BTC will have 365 days in the year where it can be traded. Google trends and X mentions had the same number of rows as BTC, there were no issues with these variables and the datasets did not have any missing values. Some of the NYSE stocks are not traded on certain days, e.g., bank holidays so therefore some of these values had to be imputed on these given days to match the number of BTC rows overall. As this is a time series problem, visualizing the distribution and removing outliers isn't necessary as all points are reflective of what happened on a given day of trading and need to be caught (Li et al., 2018).

The value would need to be imputed, as opposed to marking and deleting the rows due to the sequential nature of the time series data. The rows were marked and either forward filled or backfilled depending on the days of the week where data was missing (Zou & Qu, 2020). As can be seen from Figure 1 the USD is not volatile from one day to the next, the same method applied for USD was then applied to the Gold and VIX columns.

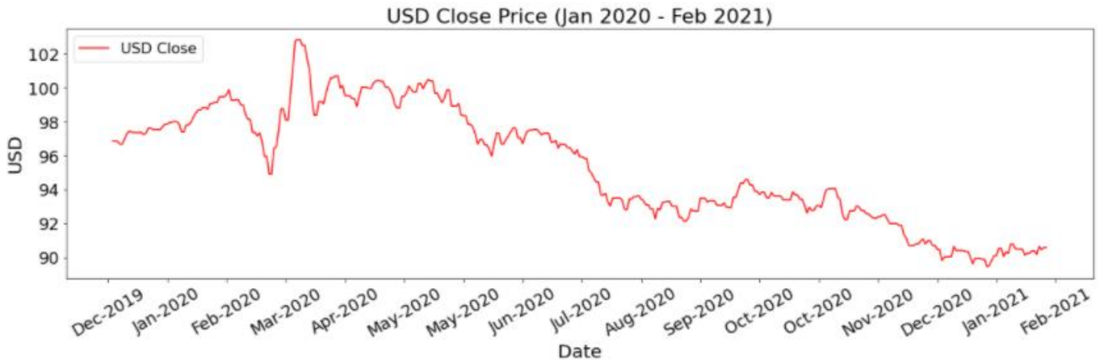

Fig. 1. USD Price (post imputing).

Once the data was imputed, it was then normalized to be inputted into the model as this improves model performance and also helps reach convergence better. A lambda function was used instead to normalize each column using the min/max method as not all variables had a gaussian distribution (Zou & Qu, 2020), the results of which can be seen in Figure 2.

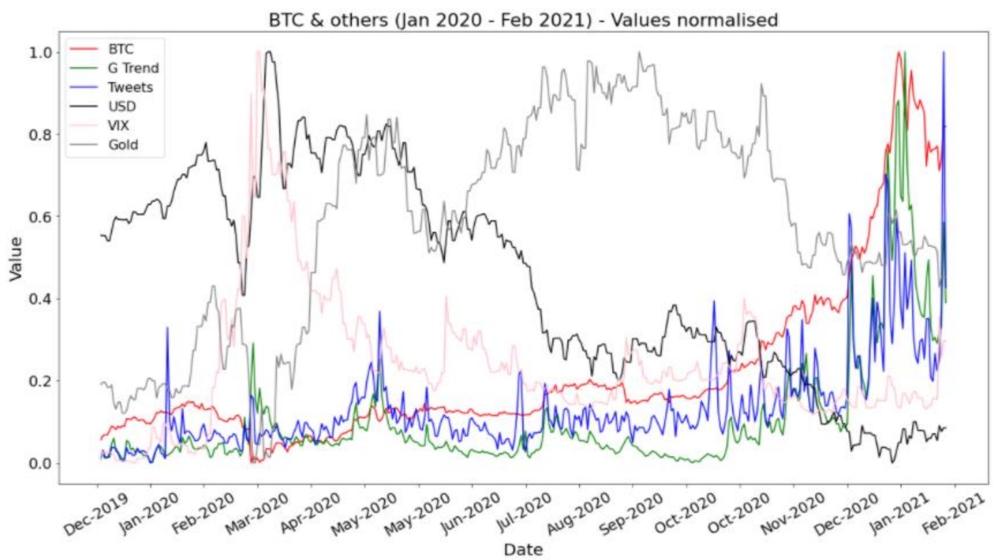

Fig. 2. Dataset Values Normalized.

The features in Figure 2 were all deemed to have a potentially good correlation to the price of BTC prior to the commencement of data collection in this study as a result of the literature review. RFE method was used as the feature selection technique, to select the most correlated features. RFE works by ranking features by importance, removing the least important ones and then re-fitting the model until the desired number of features remains (Chen & Jeong, 2007). It recursively considers smaller and smaller sets of features and prunes the least important ones; it will then give a ranking between these features. The scikit sklearn library was utilized for this RFE analysis, the results of this analysis can be seen in Table 2.

Table 2. RFE Method analysis

Variable	RFE Ranking
BTC Open	1
BTC Close	1
BTC High	1
BTC Low	1
BTC Trading Volume	6
Gold Price	7
VIX Price	4
USD Price	5

Google Trends	3
X Mentions	2

The opening, low, high and close would have the best correlation as they are all from the BTC price chart itself. For the RFE method it ranked X Mentions, Google Trends and the VIX as the highest ranks (2, 3, and 4 after the BTC technical indicators), which correlated with what the conception was from the outset (Shen et al., 2019), these were then used in the final model for testing in the below groupings as seen in Table 3. Group 1 acted as a control group, as these are what are used in traditional technical analysis. Group 2 was a result of the RFE analysis. Group 3 were resulting from studies on X & Google's influence on BTC price (Abraham et al., 2018). Group 4 is all the variables pertaining to the literature review.

Table 3. Feature Selection Groupings

Group Number	Number of Variables	Variables
Group 1	5	Close, High, Low, Open, Volume (stock technical indicators only)
Group 2	8	Close, High, Low, Open, Volume, VIX, Google trends, X mentions
Group 3	7	Close, High, Low, Open, Volume, Google trends, X mentions
Group 4	10	Close, High, Low, Open, Volume, VIX, Gold price, USD price, Google trends, X mentions

3.4 Tool / ML Algorithm Selection

Regression models can capture the relationships between input variables, both dependent and independent, to predict a continuous outcome. Linear Regression is a very popular model, as it finds parameters that minimise Mean Squared Error (MSE) between the target variable and the predicted variable. However, given that linear relationships are an assumption of this model it can struggle with non-linear relationships trends and noise within the data (Poole & O'Farrell, 1971). Linear Regression does not understand sequences of events or historic data, often resulting in losing chunks of information in that case. Autoregressive Integrated Moving Average Model (ARIMA) uses lagged moving averages to smooth time series data to predict future values based on previous outputs. It is prevalent in modelling time series forecasts but can have limitations in that it is difficult to model non-linear relationships between variables and there is also an assumption around constant standard deviation errors which may not always be true in practice (Siami-Namini et al., 2018). Given the ARIMA model is a simple model, hence its popularity, this will be used to gauge the LSTM's performance on just the BTC closing price, and the validity in using extra features for performance increases in the LSTM. Support Vector Regression (SVR) is an adaption of machine learning based classification theory of support vector machines, which can pick up non-linear trends in the dataset (Lu et al., 2009), therefore it will be used in this study for comparison versus the deep learning models.

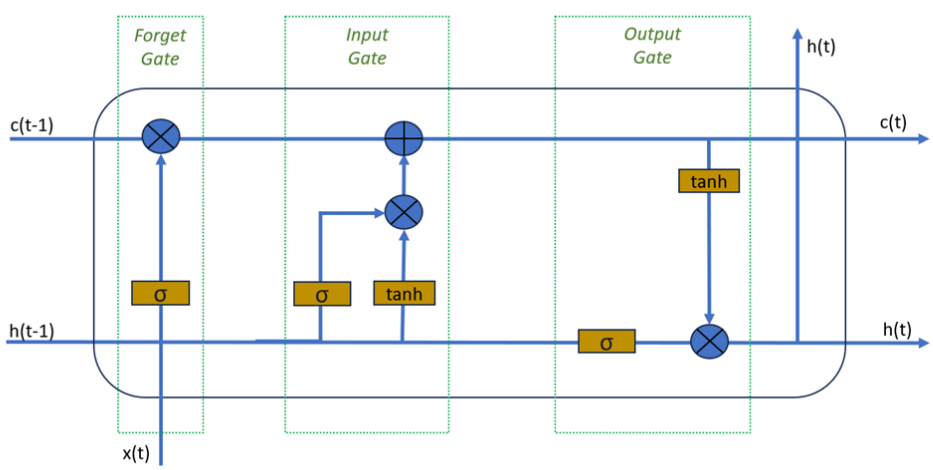

Fig. 3. LSTM Architecture

The sequential nature of time series prediction implies that an explicit order must be considered in the formulation of these prediction problems. ANN is a deep learning model capable of modelling complex relationships using forward and back propagation during model learning (Zupan, 1994). However, ANN is unaware of temporal structure as time steps are modelled as input features, meaning the network has no explicit understanding of the order between observations. Recurrent Neural Networks (RNN) are more suited to difficult sequence problems, using loops at hidden layers which essentially add memory to the network, and adding past values as part of the current values. By having the capability of the output of the network feeding back to the input of it with the next input vector, this allows the model to learn broader abstractions from the input sequences (Siami-Namini et al., 2018). LSTM is a form of a RNN, which is traditionally used in predicting sequential data. A LSTM consists of a cell and several gates as seen in Figure 3 (forget, input & output). They are sequential networks typically used to learn long-term dependencies and are commonly used in time series prediction.

LSTM models use a memory cell in the hidden layer to retain values over intervals, controlled by input, output, and forget gates as seen in Figure 3. This facilitates prediction in time series models like stock prices. Google's TensorFlow & Keras libraries provided APIs for building, training, and outputting predictions. During the model's development the MSE loss metric gauged performance, considering epochs to avoid over-training and acceptable noise from batch size. RMSE assessed model performance for stock price prediction, deemed more suitable than R^2 for feature selection. RMSE is a measure frequently used to gauge the accuracy of prediction a model makes, it measures the residuals between actual and predicted value and penalises large errors. A big advantage of RMSE is interoperability as it evaluates error in the same scale of the target variable, measuring the accuracy of the model which aligns with the model's primary purpose (Ağbulut et al., 2021). The Adam optimiser was used for this research as it builds upon RMSProp and adds momentum (Zou et al., 2018).

3.5 Variable Creation

The model developed was multivariate, thus consisting of multiple inputs (X), as mentioned in the data selection section, which were used to predict a single output variable (target variable). This target variable value would be the next day's value for the closing price of BTC. In order to create the target variable for this research the 'sliding window' methodology is put in place. This is often a method used in time series forecasting whereby prior time steps are used to predict the proceeding time step, i.e., all the input variables are used to predict next day's BTC price. Time series data can vary drastically during the period of observation and therefore become highly non-linear. For accurate predictions of highly non-linear stock data an average of stock data, or a window, might be more suitable (Hota et al., 2017).

The look back is the amount of previous time steps used to predict the next time step. Lookbacks of 1 to 10 days, 1 & 2 weeks were used on this model during testing. Due to stock volatility, the sliding window method to predict the next day price more so than a long trend garnered more success (Nelson et al. 2017; Selvin et al., 2017). When working with cryptocurrencies such as Bitcoin, the price change can vary a lot in a day, let alone a week. Cryptocurrency stocks are much less stable than more traditional stocks, hence why a 2-week window for example, would not work in most circumstances and that is why a 1-day look back was settled on for this paper.

3.6 Model Development

Different permutations of topologies were investigated to identify the best RMSE scores for the LSTM model. As the LSTM is known to not require a large network to get good results in the model development stage two versions of the model were tested, a single LSTM model and then a stacked LSTM model (Zou & Qu, 2020). LSTM models expect a 3D input to run consisting of [samples, time step, features], the inputs were shaped appropriately to conform to this. The different model topologies were tested with some sample results in Table 4 and 5. The method applied here was to use Neurons starting with a smaller network and expanding until there was no improvement in results seen, once no improvement in results were observed the other hyperparameters such as batch sizes and epochs were tweaked to get the best configuration.

Table 4. Stacked LSTM Topologies

Neurons (layer 1)	Neurons (layer 2)	Batch Size	Epochs	Train RMSE	Test RMSE
12	4	1	25	341	1197
12	4	4	25	348	1145
12	4	4	25	348	1378
8	4	1	20	343	1464
8	2	1	25	326	3535
8	6	1	25	350	1210

10	6	1	20	340	1290
10	6	2	20	341	2206
12	6	2	20	334	3613
12	6	2	20	376	1858
14	6	1	20	328	1626
14	4	1	20	418	2636
14	4	4	20	340	1515
6	2	1	20	339	5207

The fully connected layer which followed the LSTM layer to output the predictions was the Dense layer. A linear activation function was implemented as this is a regression problem. As can be seen from the different configurations in Tables 4 and 5, using 12 neurons worked best in both of them. Similar outcomes as this model were seen in other research, where the stacked LSTM model did not perform better than the normal LSTM model (Zou & Qu, 2020). This can be seen in Tables 4 and 5, where the single LSTM models outperformed the stacked versions.

Table 5. Single Layer LSTM Topologies

Neurons (layer 1)	Batch Size	Epochs	Train RMSE	Test RMSE
8	2	25	334	1277
12	1	25	352	1610
12	2	25	389	1122
12	2	15	332	1203
12	4	15	343	1167
4	2	25	338	1456
4	1	25	421	1135
12	1	25	240	1172
6	1	20	367	2275
6	2	15	342	3403
8	1	15	341	1115
8	2	15	348	1296
8	4	15	443	2939
14	1	20	466	1552

The hyperparameters were then tested to see the effect on noise/loss and RMSE seen in the model. Increasing the batch size did smooth out the noise in model development, batch sizes of 1 to 4 were utilised in the process with not much noise variation seen between the lower and higher values. Ultimately the smaller batch sizes performed better as an increase in noise in the loss gradient was not seen, therefore the stochastic approach was decided as appropriate to use for this model. The number of epochs did not have much bearing due to the relatively small network size, as seen in Figure 4 convergence was reached after a small number of epochs in the model of approximately five. There were no issues with the updates getting stuck in local minima here.

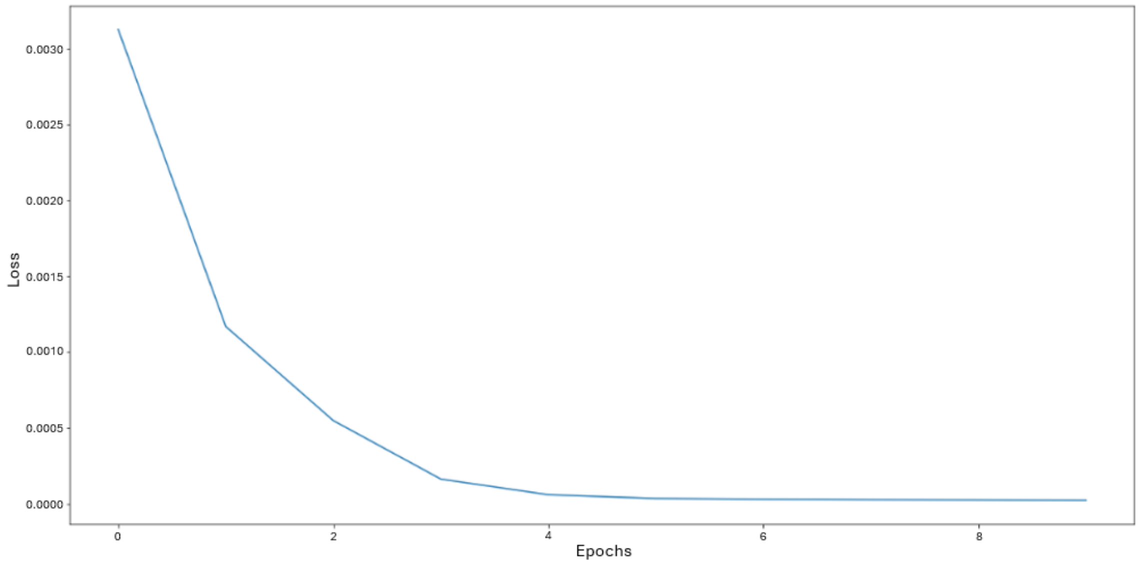

Fig. 4. Loss metric versus number of epochs

The sliding window remained as a 1-day interval (Figure 5), the model performance didn't improve when using a larger window, using a larger window would also decrease the number of samples possible for training the model. Longer periods of data can have success as seen in other studies but the authors had 7 years of daily data to utilise (Nelson et al. 2017). Using a window such as two weeks in this case would not see much success, and this is mostly due to the volatile nature of the stock itself.

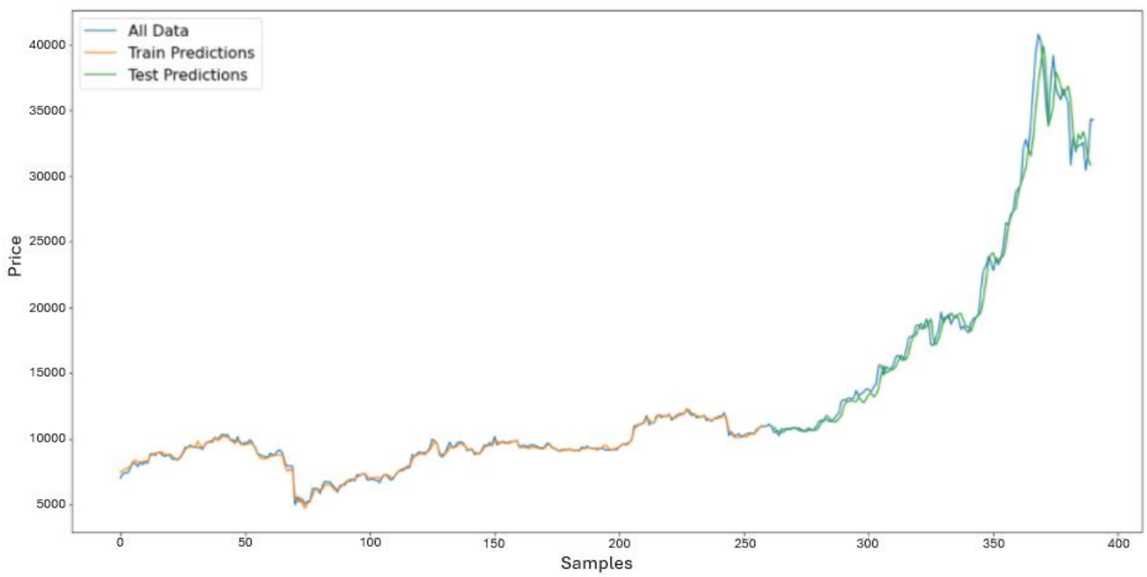

Fig. 5. Train & Test prediction of LSTM model in development

4 Results

To monitor the performance of the LSTM versus the ANN, SVR & RNN models both of them were fed with a range of variable groupings as mentioned in model development. This was performed to determine which model was better at learning the signal between the variables and not the dataset, ultimately ending in a better convergence and RMSE score. To do this, the 4 groupings of variables from the data pre-processing stage are utilised (see Tables 6-9).

As mentioned in the methodology, the RFE method was used to select the features which are most relevant in picking the target variable. This selection algorithm gave back the results seen in Table 2 for the most relevant features, apart from the technical indicators it chose X mentions, Google trends, then VIX as the most relevant in that order. As a result, groups 2 and 3 were created as they use these selections along with the technical indicators, trends and mentions were put on their own as these tend to be very strong in predicting BTC price (Shen et al., 2019). Group 1 would be used as a control for seeing if the addition of the extra variable other than the technical indicators made the model perform any better and Group 4 is all the data columns in the dataset. Tables 6-9 present the groups performances.

Table 6. Group 1 – RMSE values

Model	Train RMSE	Test RMSE	All Data
LSTM	117.78	1073.66	691.04
ANN	340.83	1312.51	842.67
RNN	355.97	5340.18	871.62
LSTM	360.73	2372.12	656.50

Table 7. Group 2 – RMSE values

Model	Train RMSE	Test RMSE	All Data
LSTM	143.78	892.84	742.58
ANN	148.06	6395.24	847.30
RNN	137.78	2271.48	722.12
LSTM	384.54	3733.60	666.55

Table 8. Group 3 – RMSE values

Model	Train RMSE	Test RMSE	All Data
LSTM	230.33	662.23	268.83
ANN	157.85	750.53	736.92
RNN	121.74	3193.93	416.16
LSTM	385.66	2880.3	662.15

Table 9. Group 4 – RMSE values

Model	Train RMSE	Test RMSE	All Data
LSTM	145.57	1460.86	700.87
ANN	249.31	6119.42	773.04
RNN	389.16	4052.62	762.03
LSTM	409.47	4742.29	659.96

The models that used the top two features (group 3) selected by the RFE method were the best performing for the ANN, RNN and LSTM when all the data was used on the models. These are the features which reflect the fundamental analysis of BTC price in the market, this corresponds to similar findings in other research (Abraham et al., 2018). The SVR did perform marginally better on the technical indicators (group 1) than others, but its results were consistent across all feature groups. The models have varying degrees of success when comparing the train and test sequences. Again, all models performed best with the data in group 3, but the ANN can be seen performing better in group 2 for train, and LSTM better in groups 1, 2 and 4 for train.

Similar analysis was performed on NYSE stocks where the LSTM model used technical indicators, and fundamental indicators such as Profit Margin, Earnings Per Share and more were inputs to the model (Zou & Qu, 2020). A selection of the Tech stocks prices are chosen for comparison to the LSTM BTC results in this research, and for increased interpretability the RMSE is shown as a percentage of target value to compare, given the different variables used in both studies. The single LSTM models and inputs used achieved RMSE percentage errors of 16%, 2.6%, 3.7% and 21.3% for Microsoft, IBM, AAPL and Google respectively (Zou & Qu, 2020). The BTC LSTM used in this research had RMSE percentage error of 1.75%, however the BTC target values in this research had a range of 38,000 while the highest range observed in the selected stocks from the comparison study is 41. Similar research was performed where the LSTM seen a RMSE percentage error of 6.4% on the NASDAQ stock index price using just the technical indicators (Siami-Namini et al., 2018). This indicates the fundamental inputs used in the BTC research (volume of tweets and Google searches) are an effective indicator of price movements for BTC.

The inclusion of the extra features is also vindicated as the LSTM model has a RMSE of 667.03 for univariate inputs versus a RMSE of 268.83 when using multivariate inputs. The LSTM also outperforms the ARIMA model output of 742.83 RMSE and the RNN model of 754.71 RMSE when using the same univariate input. The LSTM performance improving on the ARIMA has been documented before (Zou & Qu, 2020; Siami-Namini et al., 2018), these results are also observed in this research, again validating the selection of the LSTM model for stock price prediction over other more traditional models. In time series datasets there is a sequence of dependence between the input variables, the LSTM is able to handle this complexity better than the other models.

Table 10. Forecasting RMSE shown per number of days look back

Days	Train RMSE	Test RMSE	All Data
1	230.33	662.23	268.83
2	390.22	1412.54	744.57
3	446.77	2656.14	685.63
5	480.35	3974.58	770.00
7	449.70	1279.58	910.58
10	431.33	1330.33	808.41
14	443.88	2874.06	678.83

The results in Table 10 indicate the volatility in the stock seen had a negative impact on the forecasting accuracy as the day intervals were increased. There was some improvement seen in train/test for the 10-day interval compared to smaller days but ultimately there was no improvement seen from the initial 1-day interval, an expected result for a volatile stock. It should be noted that the price range in this dataset is 4,000 to approx. 42,000, it has an effect on the RMSE value obtained. The training data didn't see the spikes seen in the test data so naturally the RMSE for the test is going to be larger, not an indication overfitting of the data but showing how volatile this stock actually is.

5 Conclusion

The observed outcomes can be attributed to several factors. Notably, there is a prominent peak towards the end of the dataset, coinciding with a substantial BTC price increase. This introduces complexity to the model's learning process, imparting a skew to the test set not present in the training set. From tables 6 to 9, it is evident that the LSTM model consistently outperformed the ANN, RNN, and SVR models, showcasing superior performance, particularly in handling the more volatile unseen data in the test set. This substantiates the idea that combining less traditional fundamental analysis methods with traditional technical features yields the most effective model for stock price prediction.

The wide range of variables used, and the inclusion of extra technical and fundamental variables not typically seen in stock price analysis, coupled with the use of a LSTM model for such a volatile stock is a novel approach in the area of time-series stock prediction. The LSTM model was able to deal with the non-linear and complex data relationships that traditional models such as ARIMA cannot manage. It also confirmed there was value to be learned from the data sequence as it performed better than the ANN model. The addition of a memory cell in the LSTM architecture with gates to control the flow of data for long- and short-term memory feedback makes for an improved score in the LSTM model's performance.

The results affirm that the LSTM model effectively captured the signal of the data better than other models across all variable groupings. The RMSE of 228.83, considering prediction values ranging from 4000 to approximately 42,000 in this dataset, underscores the accuracy and ability of the LSTM model to comprehend not just the pattern but also the nuances of the dataset's signals. The inclusion of additional fundamental analysis variables is justified, enhancing the model's convergence compared to relying solely on traditional technical analysis variables. The utilization of LSTM's long/short-term memory, coupled with the appropriate input variables, positions it as a potent tool for addressing time series problems, as demonstrated in this study.

Potential areas for further research involve applying the same methodologies to different stocks of companies in the NYSE. While BTC's price volatility presented a formidable challenge, the LSTM model showcased high performance, suggesting that with more standard stocks exhibiting less variation, the LSTM model and methodologies applied in this study could achieve even greater accuracy.

However, a limitation of this study lies in the robustness of the data, resulting in challenges when forecasting BTC prices for longer intervals, such as 1, 2 weeks, or 3 days. A problem that arises with multivariate data such as the data used in this research is the issue that occurs with predicting future lookaheads for longer intervals. This is an inherent difficulty with multivariate data when attempting to increase forecasting intervals to more than one day. There are also external factors which can have an impact on this such as political and economic issues, hence the reason for stock market volatility indicator inclusions. The complexity of the data used and how the data evolves and interacts over time can incur a decrease in accuracy if the forecasting period is increased. This is due to the dynamism of the variables in the model in this research, and the increased volatility of the BTC stock makes it difficult to accurately predict long-term, for this reason the validation of the model is prioritised over future predictions.

While the LSTM model demonstrated impressive accuracy in this research compared to other models, exploring potentially more complex models could further investigate the best performance, given the selected inputs. Although the RFE method was employed for feature selection, alternative methods such as Principal Component Analysis (PCA) could have provided different groupings for feature selection. While attempts were made by the authors in the research to bring as many attributes together as possible for model evaluation, additional variables could be explored. More variables can be exhausted with this model to test the impact on the price prediction accuracy of BTC. For example, items such as 3, 5, 100-day moving averages can be used when performing stock analysis and other sources of fundamental data could also be explored which may have had an additional impact on model accuracy.

References

Abraham, J., Higdon, D., Nelson, J., Ibarra, J., & Nelson, J. (2018). Cryptocurrency Price Prediction Using Tweet Volumes and Sentiment Analysis. SMU Data Science Review, 1(3). https://scholar.smu.edu/cgi/viewcontent.cgi?article=1039&context=datasciencereview

Ağbulut, Ü., Gürel, A. E., & Biçen, Y. (2021). Prediction of daily global solar radiation using different machine learning algorithms: Evaluation and comparison. *Renewable and Sustainable Energy Reviews*, 135, 110114. https://doi.org/10.1016/j.rser.2020.110114

Alberg, J., & Lipton, Z. C. (2017). Improving factor-based quantitative investing by forecasting company fundamentals. arXiv preprint arXiv:1711.04837.

BitInfoCharts. (2009). Retrieved November 12, 2023, from https://bitinfocharts.com

Chen, X., & Jeong, J. C. (2007). Enhanced recursive feature elimination. *Sixth International Conference on Machine Learning and Applications (ICMLA 2007)*. https://doi.org/10.1109/icmla.2007.35

Google. (n.d.). Google Trends. Retrieved November 14, 2023, from https://trends.google.com/trends

Hota, H.S., Handa, R., & Shrivas, A.K. (2017). Time Series Data Prediction Using Sliding Window Based RBF Neural Network. *International Journal of Computational Intelligence Research* 13(5), 1145–1156.

Investopedia. (2021). What is bitcoin? how to mine, buy, and use it. Retrieved February 16, 2024, from https://www.investopedia.com/terms/b/bitcoin.asp

Investopedia. (2023, January 5). Stock Symbol (Ticker). Retrieved January 17, 2024, from https://www.investopedia.com/terms/s/stocksymbol.asp

Investopedia. (2023, August 28). Introduction to Supply and Demand. Retrieved January 11, 2024, from https://www.investopedia.com/articles/economics/11/intro-supply-demand.asp

Kazem, A., Sharifi, E., Hussain, F. K., Saberi, M., & Hussain, O. K. (2013). Support vector regression with chaos-based firefly algorithm for stock market price forecasting. *Applied Soft Computing*, 13(2), 947–958. https://doi.org/10.1016/j.asoc.2012.09.024

Klein, T., Pham Thu, H., & Walther, T. (2018). Bitcoin is not the New Gold – A comparison of volatility, correlation, and portfolio performance. *International Review of Financial Analysis*, 59, 105–116. https://doi.org/10.1016/j.irfa.2018.07.010

Kraaijeveld, O., & De Smedt, J. (2020). The predictive power of public Twitter sentiment for forecasting cryptocurrency prices. *Journal of International Financial Markets, Institutions and Money*, 65, 101188. https://doi.org/10.1016/j.intfin.2020.101188

Li, S., Li, W., Cook, C., Zhu, C., & Gao, Y. (2018). Independently Recurrent Neural Network (IndRNN): Building A Longer and Deeper RNN. *ArXiv (Cornell University)*. https://doi.org/10.48550/arxiv.1803.04831

Libman, D., Haber, S., & Schaps, M. (2019). Volume Prediction With Neural Networks. *Frontiers in Artificial Intelligence*, 2. https://doi.org/10.3389/frai.2019.00021

Lu, C.-J., Lee, T.-S., & Chiu, C.-C. (2009). Financial time series forecasting using independent component analysis and support vector regression. *Decision Support Systems*, 47(2), 115–125. https://doi.org/10.1016/j.dss.2009.02.001

Nelson, D. M. Q., Pereira, A. C. M., & de Oliveira, R. A. (2017). Stock market's price movement prediction with LSTM neural networks. *2017 International Joint Conference on Neural Networks (IJCNN)*. https://doi.org/10.1109/ijcnn.2017.7966019

Poole, M. A., & O'Farrell, P. N. (1971). The Assumptions of the Linear Regression Model. *Transactions of the Institute of British Geographers*, 52(52), 145. https://doi.org/10.2307/621706

PyPI. (n.d.). pytrends: Pseudo API for Google Trends. Retrieved November 11, 2023, from https://pypi.org/project/pytrends

Rafiq, M. Y., Guido Bugmann, & Easterbrook, D. J. (2001). Neural network design for engineering applications. *Computers & Structures*, 79(17), 1541–1552. https://doi.org/10.1016/s0045-7949(01)00039-6

Rotman, Sarah, (2014), Bitcoin Versus Electronic Money, No 18418, World Bank Publications - Reports, The World Bank Group, https://EconPapers.repec.org/RePEc:wbk:wboper:18418.

Selvin, S., Vinayakumar, R., Gopalakrishnan, E. A., Menon, V. K., & Soman, K. P. (2017). Stock price prediction using LSTM, RNN and CNN-sliding window model. *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI).* https://doi.org/10.1109/icacci.2017.8126078

Shen, D., Urquhart, A., & Wang, P. (2019). Does twitter predict Bitcoin? *Economics Letters*, 174, 118–122. https://doi.org/10.1016/j.econlet.2018.11.007

Siami-Namini, S., Tavakoli, N., & Siami Namin, A. (2018). A Comparison of ARIMA and LSTM in Forecasting Time Series. *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA).* https://doi.org/10.1109/icmla.2018.00227

Vargas, M. R., de Lima, B. S. L. P., & Evsukoff, A. G. (2017). Deep learning for stock market prediction from financial news articles. *2017 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA)*. https://doi.org/10.1109/civemsa.2017.7995302

Yahoo Finance. (2024). Yahoo Finance - Business Finance, Stock Market, Quotes, News. Retrieved November 17, 2023, from https://finance.yahoo.com

Zhuge, Q., Xu, L., & Zhang, G. (2017). LSTM Neural Network with Emotional Analysis for Prediction of Stock Price. *Engineering Letters*, 25(2), 167–175.

Zou, F., Shen, L., Jie, Z., Zhang, W., & Liu, W. (2018). A Sufficient Condition for Convergences of Adam and RMSProp. *ArXiv (Cornell University).* https://doi.org/10.48550/arxiv.1811.09358

Zou, Z. B., & Qu, Z. (2020). Using LSTM in Stock prediction and Quantitative Trading. CS230: Deep learning, winter (2020): 1-6 https://cs230.stanford.edu/projects_winter_2020/reports/32066186.pdf

Zupan, J. (1994). Introduction to Artificial Neural Network (ANN) Methods: What They Are and How to Use Them. *Acta Chimica Slovenica*. 41(3), 327–327.