Understanding BITCOIN Market Mechanics Using Feature Engineering, Data Modeling, and Forecasting Methods

Ibrahim, Ahmed

https://doi.org/10.34737/wvx46

Understanding BITCOIN Market Mechanics Using Feature Engineering, Data Modeling, and Forecasting Methods

Ahmed Ibrahim

A thesis by published work
submitted to the Department of Computer Science and the committee and Graduate studies of the University of Westminster in partial fulfillment of the
requirements for the degree of
Doctor in Philosophy
in Computer Science

London, UK, 2024

Academic Supervisors: Dr. Panagiotis Chountas and Dr. Hamzah Alzubi, University of Westminster

For more information, please contact:

University of Westminster
School of Computer Science and Engineering 115 New Cavendish Street
London UK
W1W 6UW
E-mail: A.Ibrahim5@westminster.ac.uk

STATEMENT OF AUTHENTICATION

This thesis is submitted to the University of Westminster in fulfilment of the requirements for the Doctor of Philosophy Degree.

I declare that this thesis is composed entirely by myself. It was not submitted in whole or part to any previous application for a degree. The presented work is my own and was self-funded.

The thesis writing was guided under the supervision of Dr. Panagiotis Chountas and Dr. Hamzah Alzubi and is, to the best of my knowledge, original except as acknowledged in the text.

Ahmed Ibrahim

MASc. (Waterloo 2014), MSc,. (AAST, 2005), BEng .(AAST, 1995).

Understanding BITCOIN Market Mechanics Using Efficient Feature Engineering, Data Modeling, and Forecasting Methods

Abstract

Bitcoin (BTC) has emerged as a groundbreaking and influential cryptocurrency, revolutionizing the financial landscape. Traders operating in the Bitcoin market encounter numerous challenges when it comes to making informed decisions due to the inherent volatility of the cryptocurrency market. Given the challenges posed by the volatile nature of the Bitcoin market, this thesis focuses on understanding the market mechanics (i.e., the underlying factors influencing price movements) to assist traders in making well-informed and profitable decisions in the unpredictable cryptocurrency market. This includes the development of prediction models that can utilize both structured (such as trading data) and unstructured data (such as social media posts) to anticipate the direction of Bitcoin's price movements and support decision-making, especially in unstable markets (e.g., the COVID-19 pandemic). The thesis represents a compendium of published papers. Article 1 provides a literature review and comparative analysis of state-of-the-art time series prediction models. In Article 2, the BTC market mechanics are simulated using a feature set of endogenous and exogenous variables. It is necessary to recognize patterns within images of time-series data charts using deep learning, as shown in Article 3. Existing forecasting models fall short of providing a robust model that handles unstructured data while providing accurate forecasting results. Thus, in Articles 4 and 5, an efficient forecasting model using ensemble and consensus learning, respectively, are proposed, which accurately analyzes the trend of BTC during the COVID-19 pandemic using Twitter posts using labeled and unlabeled data. Collectively, this thesis has contributed new insights into the BTC market. Future research could build on these findings to focus on three key areas: 1) Obtaining a greater understanding of other cryptocurrencies and stock data, 2) varying the adopted baseline models, and 3) including federated learning to handle the large size of the social datasets.

Acknowledgments

First and foremost, I would like to sincerely thank my supervisor, Dr. Panagiotis Chountas and Dr. Hamzah Alzubi, whose knowledge and expertise were invaluable to me throughout my study at The University of Westminster, particularly in conducting this thesis. Their insightful feedback and, guidance, and confidence in my research pushed me to sharpen my thinking and brought this work to a whole new level.

I would also like to thank my wife, my sons Adam and Yusuf, and my parents for their unconditional support and encouragement throughout my study.

Original Peer-reviewed Publications

This thesis comprises the following peer-reviewed journal and conference publications:

1. **Ibrahim**, A., Kashef, R., & Corrigan, L. (2021). "Predicting market movement direction for Bitcoin: A comparison of time series modeling methods." Computers & Electrical Engineering, V.89, pp. 106905-106915.
2. **Ibrahim**, A. Kashef, R., Li, M., Valencia, E., Huang (2020). "Bitcoin network mechanics: Forecasting the BTC closing price using vector auto-regression models based on endogenous and exogenous feature variables."Journal of Risk and Financial Management, 13(9), pp. 189-210.
3. **Ibrahim**, A. F., Corrigan, L., & Kashef, R. (2020). "Predicting the Demand in Bitcoin Using Data Charts: A Convolutional Neural Networks Prediction Model."In 2020 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE) (pp. 1-4). IEEE.
4. **Ibrahim**, A. (2021, April), (, "Forecasting the Early Market Movement in Bitcoin Using Twitter's Sentiment Analysis: An Ensemble-based Prediction Model."In 2021 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS) (pp. 1-5). IEEE.
5. **Ibrahim**, A. (2021, October), "Analyzing BTC's Trend During COVID-19 Using A Sentiment Consensus Clustering (SCC)". In 2021 IEEE 12th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON) (pp. 0460-0465). IEEE.

My Publications – The following are electronic copies of my publications, listed in order as they will appear in my thesis, where" J" refers to a Journal and "C" refers to a Conference.

- (J1) https://doi.org/10.1016/j.compeleceng.2020.106905
- (J2)* https://doi.org/10.3390/jrfm13090189
- (C1) http://doi.org/10.1109/CCECE47787.2020.9255711
- (C2) http://doi.org/10.1109/IEMTRONICS52119.2021.9422647
- (C3) http://doi.org/ 10.1109/IEMCON53756.2021.9623182

Acronyms and Abbreviations

BTC	Bitcoin
C-C-C-V	Close-to-Close volatility
H-L-V	High-Low volatility,
OHLCV	Open-High-Low-Close volatility)
MA	Moving Average
SMA	Smoothing Moving Average
EMA	Exponential Moving Average
SAR	Stop-and-Reverse (SAR)
MACD	Moving Average Convergence Deviance
RSI	Relative Strength Index
OBV	On Balance Volume
TI	Technical Indicators
SCC	Sentiment Consensus Clustering
CEPM	Composite Ensemble Prediction Model
ARIMA	Autoregressive Integrated Moving Average
RF	Random Forest
VAR	Vector-Auto-Regression
BVAR	Bayesian Vector-Auto-Regression
MLP	Multi-Layer Perceptron
MKPRU	Equilibrium closing price
MWNUS	The number of unique MyWallet users
TOTBC	The total BTC available in the market to date
AVBLS	Average Block Size
DIFF	Bitcoin Difficulty
NTRBL	Number of Transactions per Block
MIREV	Miner's Revenue

NADDU	Change in the Number of Unique Addresses
TRVOU	Total Output Volume
HRATE	Hash Rate
FPE	Forecast Prediction Error
CNN	Convolutional Neural Network
LR	Linear Regression
SVM	Support Vector Machines

Table of Contents

List of Figures

List of Tables

Chapter 1 Introduction

1.1 Motivation

Several issues have arisen in the new era of digital money, including evaluating market dynamics, price prediction, data modeling, and trend forecasting. Several drivers impacting the Bitcoin market, including supply, demand, social influences, and regulation, have caused a great need to design an in-depth understanding of what drives BTC and develop efficient prediction models, which can be helpful to numerous stakeholders. However, the complex nature of any financial market warrants a more sophisticated forecasting model with an optimal selection of individual influence factors. Traditional machine-learning methods and time-series forecasting fall short of better understanding the BTC mechanics and providing effective forecasting models for structured and unstructured data (i.e., social media data) in unprecedented market crash periods such as COVID-19. This research aims to improve upon traditional machine learning and time-series analysis methods by using simulation to dive deep into the mechanics of the BTC market and identify unique market makers. The goal of using structured and unstructured data is to develop efficient models to predict the direction of price and trend movement in the BTC market.

1.2 Background on this research

Cryptocurrencies are digital assets that use cryptography for secure financial transactions. They operate on decentralized networks, meaning they are not controlled by any central authority, such as a government or financial institution. Instead, transactions are verified by network nodes through a process called mining, in which computers solve complex mathematical problems to validate transactions and add them to the blockchain. This decentralized ledger records all cryptocurrency transactions. Bitcoin is a decentralized cryptocurrency that allows for secure, peer-to-peer financial transactions. It has gained popularity as both a form of electronic cash and an investment asset. As an investment

asset, Bitcoin has been highly volatile, with significant price fluctuations over time. This volatility can be attributed to a number of factors, including market speculation, regulatory changes, and the overall maturity of the cryptocurrency market. The value of Bitcoin and other cryptocurrencies can fluctuate significantly, and investors may lose a significant portion or even all of their investments. It is important for potential investors to carefully consider the risks and potential rewards of investing before making any investment decisions. Thus, there is a need to understand the BTC market mechanics and factors that impact the BTC market. In addition, with the fast amount of data available online, both structured (e.g., trading data) and unstructured, such as social media posts, there is a need to develop robust and efficient forecasting models that can help traders invest at the right time. This is crucial, especially in market crash periods (e.g., COVID-19) when there are tremendous changes and fluctuations in the cryptocurrency market. Several approaches can be used to forecast the price and trend of Bitcoin and other cryptocurrencies.

- Technical analysis: this is based on the idea that historical price patterns can provide insight into future price movements. Technical analysis of cryptocurrency involves using various tools and techniques to analyze market data and identify trends and patterns that may provide insight into future market movements. Some measures that technical analysis may look for include Price trends, Moving averages, Oscillators, Chart Patterns, and Market Volume. However, this approach has its limitations. For example, it does not consider fundamental factors such as economic conditions or market news, which can impact prices. In addition, technical analysis has a subjective element, as different analysts may interpret the same chart patterns differently.
- Fundamental analysis: Fundamental analysis of cryptocurrency involves using various tools and techniques to evaluate the intrinsic value of a cryptocurrency by examining a range of economic, financial, and other qualitative and quantitative factors. Some measures that fundamental analysis may consider include demand and supply dynamics, the regulatory environment, economic conditions, market trends and sentiment, technological developments, and competition. However, this

approach can be limited by the availability and accuracy of data. It can also be challenging to predict how these factors will change in the future and how they will impact the price of Bitcoin.

- Time-series prediction models are statistical methods used to forecast future values of a time-series based on past data. A time series is a series of data points that are collected at regular intervals over a period of time. The price of Bitcoin can be highly volatile, making it difficult to forecast future values using these methods accurately. In addition, Time-series models are based on past data, so they may not be able to take into account unexpected events or changes in market conditions that could affect the price of Bitcoin. For example, the current war in Europe and how it impacts the Bitcoin price volatility and market volume.
- Machine learning algorithms: Traditional machine learning algorithms can help build predictive models, but they are only as good as the data they are trained with. If the data is incomplete or of poor quality, the resulting model may not be accurate, which can make it challenging to evaluate their reliability and interpret their results.

1.3 Problem Statement

The price of Bitcoin has experienced significant fluctuations in recent years. Thus, it can be challenging to predict how it will behave in the future, particularly in the face of unexpected events or shifts in market conditions that could impact the price of Bitcoin. This makes it difficult for investors and analysts to decide whether to buy or sell Bitcoin.

This thesis aims to gain a deeper understanding of the Bitcoin market and to create more accurate forecasts of Bitcoin's price and trends using robust forecasting models on structured and unstructured data. The Bitcoin market is complex and constantly evolving, and it can be challenging to predict how it will behave in the future. By analyzing structured and unstructured data and building forecasting models considering a wide range of factors, we hope to improve our understanding of the market and develop more accurate predictions about the Bitcoin market. We will also evaluate the performance of

different models and identify any limitations or challenges in forecasting the price of Bitcoin to identify ways to overcome these challenges.

1.4 Research Questions

Here are a few of the most important questions that motivated my research.

Question #1: *Which machine learning model best predicts BTC movement in the short term?*

Question #2: *How can feature engineering be used to optimally select endogenous and exogenous variables of interest for accurate BTC price prediction?*

Question #3: *Can we create an alternative method of modeling the Bitcoin time series to improve price prediction*?

Question #4: *How can social media help predict early cryptocurrency market movements*?

Question #5: In the absence of labeled data, which model can *invoke social media while predicting early cryptocurrency market movements*?

Question #6: *How can the proposed models be compared to existing methods?*

1.5 Thesis Objectives

The main objectives of this thesis are to examine and analyze the six specific research questions in depth and to present the findings of the analysis in a clear and organized manner. To summarize, the main objectives of this thesis are as follows:

- **To gain a deeper understanding of the Bitcoin market:** The main goal of this research is to gain a more thorough understanding of the Bitcoin market. This includes understanding the various factors influencing the market and how they interact. By gaining a deeper understanding of the market, we can better predict its future behavior and make more informed decisions about investing in or trading Bitcoin.

- **To analyze structured data in the Bitcoin market:** In this research, we will be analyzing structured data from the Bitcoin market and provide better data representations as images to make more accurate forecasts about the future price of Bitcoin.
- In this thesis, we will be **using unstructured data from social media** on the Bitcoin market to perform an analysis that will help us better understand the market during unexpected events or changes in market conditions that could affect the price of Bitcoin.
- **To develop more accurate forecasts of Bitcoin's price using robust forecasting models**: A key objective of this research is to develop robust forecasting models that can accurately predict the future price of Bitcoin. We will use structured and unstructured data and consider various factors influencing the market.
- **To evaluate the performance of different forecasting models for the Bitcoin market**: In this research, we will evaluate the performance of different forecasting models for the Bitcoin market. This will involve comparing the accuracy of different models and identifying any strengths or weaknesses of each model. By doing this, we aim to identify the most effective forecasting models for the Bitcoin market.
- **To identify limitations and challenges in forecasting the price of Bitcoin and develop strategies to overcome them:** Forecasting the price of Bitcoin can be challenging due to the complex and constantly evolving nature of the market. In this research, we will identify any limitations or challenges that we encounter in forecasting the price of Bitcoin and develop strategies to overcome these challenges. By doing this, we aim to improve the accuracy of our forecasts and better understand the market.

1.6 Thesis Contribution

The main contributions of this thesis can be summarized as

- A comprehensive review and analysis of the state-of-the-art time-series

prediction models are presented.

- A deep understanding of the BTC drivers using Endogenous and Exogenous Feature analysis and modeling.
- A novel method for analyzing time-series BTC using data charts to detect small and imperceptible patterns within images of time-series data charts. The proposed method has been shown to have considerable results.
- A novel ensemble model is developed to anticipate early cryptocurrency market moves, namely CEPM (Supervised model) using social media data like tweets. The proposed model outperformed the state-of-the-art prediction models using Twitter datasets acquired during the COVID-19 era.
- A novel sentiment consensus clustering (SCC) algorithm, based on the idea of cooperative learning, is proposed to predict the BTC trend in an unsupervised model. During and after the COVID-19 epidemic, the consensus model performed admirably in anticipating the BTC trend.

1.7 Thesis Outline

This thesis presented herein consists of six chapters: Introduction (Chapter 1); Literature Review (Chapter 2); Research Gap, Novelty, and Methodology (Chapter 3); Bitcoin Network Mechanics: Forecasting the BTC Closing Price Using Endogenous and Exogenous Feature Variables (Chapter 4); Predicting the Demand in Bitcoin Using Data Charts (Chapter 5); Forecasting the Early Market Movement in Bitcoin Using Sentiment Analysis (Chapter 6); Analyzing BTC Trends Using Sentiment Consensus Clustering (Chapter 7); General conclusions and recommendations (Chapter 8). Concluding remarks are provided at the end of each chapter (except for Chapter 1) to highlight and summarize the key outcomes of each chapter. A brief description of each chapter discussed in this thesis is presented below.

Chapter 1 is the introductory chapter of this thesis. It starts by discussing the motivation behind this research, background, and problem statement on cryptocurrency prediction for structured and unstructured data. The aim and specific objectives, the scope of the

study, the significance of the study, and the thesis outline were also presented in this chapter. The introduction section also summarizes the publications presented in this thesis. The published work demonstrates the significance and contribution of the thesis in understanding the BTC market mechanics and its demand while analyzing various types of structured and unstructured datasets.

Chapter 2 presents the state-of-the-art literature review on time-series prediction modeling for BTC trend prediction. This chapter covers Article 1, which discusses various statistical-based and machine-learning-based. The first key research of this thesis is presented in Chapter 2, which shows the efficiency of using neural network-based models in short-term forecasting.

Chapter 3 focuses on addressing the research gap, highlighting the novelty of the thesis, and presenting the methodology employed to achieve its objectives. The primary aim of this thesis is to gain a comprehensive understanding of the market mechanics of Bitcoin and to develop prediction models that assist traders in making informed decisions within the highly volatile cryptocurrency market.

The second key research of this thesis is presented in **Chapter 4,** which focuses on investigating the individual factors of influence on the BTC market. In this chapter, Article 2 is shown, in which the BTC market mechanics are broken down using vector autoregression prediction models. These models proved useful in simulating past BTC prices using a selected feature set of exogenous variables.

The idea of forecasting trends from image representations of data is presented in **Chapter 5**. In this chapter, a key contribution of this thesis, as published in Article 3, is by proposing an advanced deep learning architecture that shows a significant improvement in analyzing time-series data charts instead of traditional feature-based time-series data.

Posts on social media platforms such as Twitter, Facebook, and Reddit can influence the perceptions and expectations of traders and investors and potentially impact the price of

Bitcoin. **Chapter 6** provides a sentiment analysis-based approach to the unstructured Twitter dataset and proposes an efficient classification algorithm that effectively predicts the BTC trend while handling unstructured data. The key findings in this chapter are published in Article 4.

To handle unstructured posts such as tweets while labels are absent, in **Chapter 7**, we proposed an unsupervised-based forecasting model that uses the notion of consensus clustering while adopting sentiment analysis to analyze the unstructured posts. The proposed model is published in Article 5.

In **Chapter 8**, The significance and key findings of the conducted research are outlined in this chapter, followed by recommendations for potential future research directions.

Overall, the thesis structure follows a logical sequence that builds upon earlier concepts and approaches to comprehensively understand the topic.

Chapter 2 – Literature Review

2.1 The Objective of The Chapter

This chapter reviews, discusses, implements, and compares various Bitcoin price prediction models with multiple strategies to help traders decide how to best act over the changes in Bitcoin prices over short timeframes by creating a model that can predict the direction of price movement.

2.2 Published Article 1

Ibrahim, A., Kashef, R., & Corrigan, L. (2021). Predicting market movement direction for Bitcoin: A comparison of time series modeling methods. Computers & Electrical Engineering, 89, 106905. ISSN 0045-7906, https://doi.org/10.1016/j.compeleceng.2020.106905.

2.3 The Article Body of Knowledge

The subsequent sections are directly excerpted from the paper "**Predicting market movement direction for Bitcoin: A comparison of time series modeling methods.**" All credits and rights are attributed to the original authors and the source publication.

2.3.1 Introduction

Over the last six months, Bitcoin and cryptocurrencies have been a significant topic in the news. In late 2017, headlines about people becoming overnight millionaires by investing in these "digital currencies" created a massive market bubble as an influx of new investors rushed to try and get into the game. This caused Bitcoin's price to reach an all-time high of nearly 20,000 USD in December 2017 before the price crashed to around 7,000 USD (at the time of writing this article). With all the speculation occurring in the Bitcoin market, price fluctuations of over 10% daily are common. Price changes in the range of 1-3% frequently happen within very short timeframes – just several minutes. These price fluctuations create massive opportunities for people to profit from trading Bitcoin over

short periods and have sparked a wave of people known as "day traders" who try to capture profit from these short periods of price fluctuation. The world of Bitcoin is following in the footsteps of other financial markets, and the use of algorithmic trading bots is becoming a common practice [1]. With over 60% of trading volume estimated to be attributed to these bots, it is getting harder for human traders to make any profit trading over short periods [1]. These bots are being supported using increasingly sophisticated artificial intelligence based on complex machine learning models, often supported by deep learning [2]. Besides, due to the large volume of data obtained to represent cryptocurrencies as time-series datasets, big data fusion is a challenging task both in the preprocessing and modeling stages [2]-[4]. The goal is to develop a model that can assist an algorithmic trading bot in making trade decisions to maximize the chance of making profitable returns when trading Bitcoin against USD pairs. In this paper, we discuss, implement, and compare various Bitcoin price prediction models with multiple strategies to help traders decide how to best act on the changes in Bitcoin prices over short timeframes by creating a model that can predict the direction of price movement.

The rest of this paper is organized and presented as follows: Section 2 presents the background of the Bitcoin market, and Section 3 discusses related work on cryptocurrency prediction models. Experimental analysis and results are presented and discussed in Section 4. Finally, conclusions and future directions are summarized in Section 5.

2.3.2 Bitcoin Market Background

While some advocates hail the invention of Bitcoin as other cryptocurrencies as a new world currency, in its current form, Bitcoin and cryptocurrencies appear to be more closely related to stocks than currencies. Cryptocurrencies also experience much higher volatility than fiat currencies, making them most akin to penny stocks. Currently, Bitcoin and other cryptocurrencies are traded using websites that offer digital exchange platforms, offering similar trading options to stock trading platforms, including put/call options and stop-limit orders. Bitcoin has an average daily trading volume of around 5 billion USD across these major exchange platforms1. Compared to the United States (US) equity market, with a

daily trading volume of 55 billion USD, the Bitcoin market shows no small player. There is some serious money already flowing around this young cryptocurrency space. While the market is still young, big institutional money is starting to trickle in. Bitcoin currently holds a market capitalization of over 110 billion USD. The Chicago Mercantile Exchange (CME) started issuing Bitcoin futures in January 2018. While the Securities And Exchange Commission (SEC) is yet to approve a Bitcoin Exchange-Traded Fund (ETF), many people believe it is only a matter of time before trading Bitcoin becomes accessible to mainstream investors [5]. Some analysts call Bitcoin a bubble, drawing parallels to the Dutch Tulip Bubble of the 1600s or the Dot-Com Bubble in 2000. Long-term speculators hope that an influx of institutional money will come with the approval of an ETF and provide a significant price increase. Although the future of cryptocurrency remains unclear, there exists a real opportunity to profit from trading Bitcoin at this current point in time. Bitcoin as blockchain currencies are not as liquid as other forms of currency; thus, understanding the Bitcoin market's behavior draws insights on how to capitalize on this asset over time [6][7].

2.3.3 Related work on Cryptocurrency prediction Models

Cryptocurrencies have already been extensively analyzed for suitability as trading instruments. Carpenter and Chen have discussed the benefits of using cryptocurrencies to augment traditional portfolios [8]. Zhengyao Jiang has gone a step further in investigating crypto-only portfolios [9]. Others have had similar ideas about the efficacy of treating cryptocurrencies as investment objects. Price prediction has been another increasingly hot topic in the field of cryptocurrency research. Shah has applied Bayesian regression [10], and Madan has implemented deep learning and tree-based regression techniques [11]. In terms of similar research, research into price prediction models on stock data has been summarized nicely in the review of state-of-the-art stock prediction techniques [12]. In a similar up/down classification problem, in [13], a top accuracy of 56% across the tests using PCA dimensionality reduction techniques was achieved by choosing a wide set of technical indicators. While a lot of new research uses deep learning for stock

price analysis, some serious debate remains about the performance gained by these complex models over their simpler linear counterparts [14]. This has created an incentive for the tests within the scope of this paper. Comparing linear models with deep learning models, especially within financial markets, is an interesting field of research [15]. The tested models include ARIMA, Prophet (by Facebook), Random Forest, Random Forest Lagged-Auto-Regression, and Multi-Layer Perceptron (MLP) Neural Networks.

2.3.3.1 Autoregressive Integrated Moving Average

The Autoregressive Integrated Moving Average (ARIMA) models are the general class of models for forecasting a non-stationary time series. The integrated part of the model indicates the different steps over the time series data to eliminate the non-stationary trend. The ARIMA model has two different types: seasonal and non-seasonal. $\mathrm{ARIMA}(\mathrm{p},\mathrm{d},\mathrm{q})(\mathrm{P},\mathrm{D},\mathrm{Q})_{\mathrm{m}}$ denotes the seasonal model, where *m* refers to the number of periods considered in a season, the smaller case p,d,q refers the number of autoregressive, non-seasonal difference, and moving average terms, and the upper case P,D,Q refers the number of seasonal autoregressive, seasonal difference, and seasonal moving average terms. Mathematically, $\mathrm{X_t}$ is a non-seasonal $\mathrm{ARIMA}(\mathrm{p},\mathrm{d},\mathrm{q})$ if $\nabla^{\mathrm{d}}\mathrm{X_t}$ is $\mathrm{ARMA}(\mathrm{p},\mathrm{q})$ $\beta(\mathrm{B})\nabla^{\mathrm{d}}\mathrm{X_t} = \theta(\mathrm{B})\varepsilon_{\mathrm{t}}$, where B is a Backshift operator and ∇ is a Difference operator. The price value of Bitcoin is temporal and has a variable trend. Hence, the ARIMA model is suitable for predicting the movement of Bitcoin prices. Azari in [16] has applied the ARIMA model approach to predict the future value of Bitcoin by considering the dataset consisting of the Bitcoin price for three years from 2015 to 2018. They reported a minimum Residual Sum of Squares (RSS) of 0.02. The non-linear deep learning methods outperformed the ARIMA approach, as shown in [17]. The ARIMA achieved 50.05%, whereas Long Short-Term Memory (LSTM) marked 52.78% accuracy. In [18], the ARIMA model is compared with the Prophet, multi-layered perceptron, and LSTM to predict the cash flow. The other models outperformed the ARIMA model in the long-term forecast. The seasonal ARIMA underperformed them due to squaring errors for seasonal and holiday effects. The ARIMA model is efficient for short-term prediction if data has a consistent pattern. Since ARIMA models are primarily backward-looking, the long-term

forecast eventually goes to be a straight line and is poor at predicting series with turning points. ARIMA models are the industry standard when it comes to time-series regression problems. The regression model attempts to predict the return amount for the next period. For the cryptocurrency classification problem, only the direction of price movement is considered (positive or negative), and the magnitude of the prediction is ignored.

2.3.3.2 The Prophet

Prophet is an algorithm developed by Facebook for time-series forecasting. It uses Bayesian-based curve methods for forecasting and is generally considered a competitor for ARIMA models, often achieving slightly better results [18]. The prophet model utilizes the closing price values as parameters. Similar to the ARIMA model, a sliding training window was employed so that predictions were being made only one period into the future. Facebook designed the Prophet forecasting model to handle the characteristic features of business time series, such as multiple strong seasonality, trend changes, outliers, and holiday effects. It is an additive model in which non-linear trends fit with different seasonality and holidays. The model is robust to missing data, shifts in the direction, and significant outliers. The model's implementation is made available as an open-source software distribution in Python and R [19]. In this approach, a time-series model is decomposed into three model components: trend, seasonality, and holidays.

$$y(t) = g(t) + s(t) + h(t) + \varepsilon_t \quad \text{(2-1)}$$

In the equation, g refers to trend, $s(t)$ represents periodic change, and $h(t)$ indicates the effects of holidays. The error term ε_t provides the changes that are not accommodated by the model. In the ARIMA model, the measurements must be regularly spaced, and it does not consider the outliers, whereas the Prophet model does include the outliers and data that need not be periodic. Hence, the Prophet model provides lesser error values in comparison with the ARIMA model. Yenidoğan has compared the Bitcoin price prediction done by ARIMA and Prophet models [19]. The Prophet model provided a prediction near the correct price with 94.5% precision, whereas the ARIMA model showed only 68% precision. The forecasting of cash flow, a time-series data presented in [20], also provides

a similar result to that of [19]. The Prophet model performed better than the ARIMA model as it considers the seasonality and holiday effects. However, the Multilayer Layer Perceptron (MLP) and Long Short-Term Memory (LSTM) models achieved lesser error than the Prophet model.

2.3.3.3 Random Forests

Many decision trees form the random forest classification algorithm. Random forest uses bagging and feature randomness to build each tree to create an uncorrelated forest of trees whose prediction by committee is more accurate than that of a single decision tree. Hence, the average of the predictions from individual trees, which are reasonably good models to produce a prediction that better estimates the original hypothesis. The basic principle of decision trees is the recursive partitioning of the feature space. The decision tree arrives at a single class node called a leaf node by splitting every child node. Random forest uses an ensemble approach of many such decision trees to reduce over-fitting, each tree in a random forest grown on a random subset of the feature space. The $m = \sqrt{M}$ features are randomly selected to grow each tree if each sample in a dataset has the M features. Random Forests are preferred over decision trees because of voting-based conclusions. Research work in [21] predicts Bitcoin's daily and five-minute interval price using the high-dimensional features of the Bitcoin dataset. The random forest, a machine learning model, was adopted to forecast the Bitcoin price movement. The model produced 51% accuracy and 61.2% F1 score while predicting daily Bitcoin price. On the other hand, the 64.8% accuracy and 75.8% F1 score predict Bitcoin's five-minute interval price. The Long Short-Term Memory (LSTM) model performed better in predicting a five-minute interval price with 67.2% accuracy. The Logistic Regression (LA) and Linear Discriminant Analysis (LDA) predicted the daily price with an accuracy of 66%. Suryoday had worked on predicting the direction of the stock market using tree classifiers [22]. The random forest is used to study the advantage ensemble technique to forecast medium to long-run stock value. The stock values of ten companies were predicted with a trading window range of 3 to 90 days. Random Forest achieved 59.31 % accuracy for three days' price while 94.44% accuracy for 90 days' value of the Facebook stock price. While not

traditionally used for time-series forecasting, random forests are a prevalent classification method in the world of machine learning. Two random forest models were tested to see if this technique might still have some value in predicting Bitcoin price movements. Random forest models in this paper were developed using the distributed machine learning framework H_2O. This framework was chosen because it provides a grid search method that helps select the optimal hyper-parameters. It allows for parallel processing to take advantage of the computing power on the server.

2.3.3.4 Multilayer Perceptron Deep Neural Network

The Multilayer Perceptron (MLP) is a feed-forward Artificial Neural Network (ANN). An input layer, a hidden layer, and an output layer configure an MLP. The hidden and output layers consist of nonlinearly activating nodes. The features of MLP, such as multiple layers and non-linear activation, distinguish MLP from a linear perceptron. The MLP learns by changing connection weights after processing data, depending on the amount of error in the output compared to the expected result. The objective of an MLP is to approximate some function. For instance, for a classifier,$y = f^*(x)$ maps an input x to a category y. An MLP defines a mapping $y = f(x; \theta)$ and learns the value of the parameters θ that result in the best function approximation [22]. The authors of [22] have used machine learning techniques to predict the direction, maximum, minimum, and closing prices of daily Bitcoin exchange rates. MLP, which had 20 nodes in the first layer and trained for 100 epochs, provided 58.84% accuracy. Another configuration of MLP with five nodes in the first layer and ten nodes in the second layer trained for 500 epochs showed 62.91% accuracy. The research presented in [20] compared the performance of MLP and Long Short-Term Memory (LSTM) with Autoregressive Integrated Moving Average (ARIMA) and Prophet in predicting cash flow. In the ARIMA and Prophet combination, Prophet complemented the holiday effect and changing trend issues of ARIMA. An MLP accounts for a single event at a given time and assumes all inputs to be temporarily independent of each other.

A time-series data, like cash flow or Bitcoin price value, is related to past data. A Recurrent Neural Network (RNN) node preserves information from past time stamps. Hence, the combination of MLP and LSTM is used. The accuracy of MLP and LSTM was significant in

comparison with ARIMA and Prophet. The H_2O framework also provided a deep learning framework using a multilayer perceptron neural network. Similar to the distributed random forest models, a grid search helped optimize the hyper-parameters used in the final model to achieve the highest accuracy.

2.3.4 Experimental Analysis and Results

2.3.4.1 Data Collection

The primary data for this paper was collected from Coinbase, the most popular and longest-running North American Bitcoin exchange (Coinbase. (2018), https://www.coinbase.com/). The data was collected in the form of tick-data – data tracking every individual trade dating back to 2014 – using the Application Programming Interface (API) provided by the platform. Bitcoin data dating back to September 2017 was also collected from the popular exchange Poloniex to create a second test set [23]. This data was collected using the exchange's trading API and was already in the format of a 5-minute Open High Low Close (OHLC) plus volume. Since it is assumed that Bitcoin trading resembles stock trading, some stock data was collected to test the predictive power of the models further on data for which they had not specifically been trained. The 5-minute OHLC data for Apple, Facebook, Google, and Microsoft stocks was downloaded from the Google Finance API, dating back to January 2018.

2.3.4.2 Benchmark Strategies

A. Naïve Guessing

This prediction method used a random number generator to give a number between zero and 1. For each time period in the Coinbase test set, if the number was above 0.5, the predicted direction was UP; otherwise, it was DOWN. This strategy achieved an accuracy of 50.30 %, not very impressive. To make sure that a 50/50 guess makes sense, it was checked to see if the size of the two classes, up/down, was equal. The ratio of UP/ DOWN was 0.503427, indicating that the classes are, indeed, equal.

B. Momentum Strategy

This strategy guesses that the stock price will move in the same direction as it did in the previous time period. Surprisingly, this method achieved an impressive classification accuracy of 53.85%. This strategy will be the benchmark that all other models must beat. Table 2-1 shows that the momentum strategy outperforms the naïve strategy by a statistically significant amount.

Table 2-1: Naïve Guessing Vs. Momentum Strategy

<table>
<tr><th colspan="4">McNemar's Test</th></tr>
<tr><td>McNemar's chi-squared</td><td>96.255</td><td colspan="2" rowspan="2"></td></tr>
<tr><td>p-value</td><td>< 2.2e-16</td></tr>
<tr><td colspan="2"></td><td colspan="2">Momentum Strategy</td></tr>
<tr><td colspan="2"></td><td>Correct</td><td>Wrong</td></tr>
<tr><td>Naïve Strategy</td><td>Correct</td><td>8805</td><td>7541</td></tr>
<tr><td></td><td>Wrong</td><td>8796</td><td>7540</td></tr>
</table>

2.3.4.3 Data Transformation and Feature Engineering

The Bitcoin tick-data data was transformed into 5-minute intervals using the matplotlib python package. Back-filling was used to fill prices for any 5-minute period that had no trading activity (common before 2016). After the transformation, the data had opening, high, low, and closing price variables, as well as the trade volume for those 5 minutes. To augment the data, new variables were engineered using the most popular indicators used in the stock trading industry. In the following, we discuss the indicators that were used to augment the data.

Volatility: Volatility for the price was calculated for two periods: over the past 7 days and over the last 40 5-minute periods (200 minutes). This was done to capture a macro and micro representation of the market. Three different calculations were used to capture different types of price volatility. The Close-to-Close volatility, High-Low volatility, and OHLC volatility [25] are shown in Eq.2-2, Eq.2-3, and Eq.2-4, respectively.

$$the\ \sigma_{close-to-close} = \sqrt{\frac{N}{n-2}\sum_{i=1}^{n-1}(r_i - \bar{r})^2}\ where\ r_i = \log\left(\frac{C_i}{C_{i-1}}\right)\ and\ \ \bar{r} = \frac{r_1 + r_2 + \cdots + r_{n-1}}{n-1} \quad (2\text{-}2)$$

$$\sigma_{high-low} = \sqrt{\frac{N}{n}\sum\left[\frac{1}{2}\left(\log\left(\frac{H_i}{L_i}\right)\right)^2 - (2\log(2)-1)\left(\log\left(\frac{C_i}{O_i}\right)\right)^2\right]} \text{ (2-3)}$$

$$\sigma_{open-high-low-close} = \frac{N}{n}\sum\left[\left(\log\frac{O_i}{C_{i-1}}\right)^2 + \frac{1}{2}\left(\log\frac{O_i}{C_{i-1}}\right)^2 - (2\times\log 2 - 1)\left(\log\frac{C_i}{O_i}\right)^2\right] \text{ (2-4)}$$

Moving Average: Moving averages were calculated on the close prices and volumes for 7-, 14-, and 28-day periods. Two types of moving averages were calculated, the Smoothing Moving Average (SMA) (Eq.2-5) and Exponential Moving Average (EMA) (Eq.2-6).

$$SMA_t = \frac{\sum_{i=1}^{n} C_{t-i}}{n},\ where\ C\ is\ closing\ price, n\ is\ period \text{ (2-5)}$$

$$EMA_t = \frac{\sum_{i=1}^{n}(1-\alpha)^{1-i}\times C_i}{\sum_{i=1}^{n}(1-\alpha)^{1-i}},\ where\ C\ is\ close\ price, n\ is\ period \text{ and } \alpha = \frac{2}{N}+1 \text{ (2-6)}$$

Stop-and-Reverse (SAR): The Parabolic Stop-and-Reverse calculates a trailing stop and indicates when positions should be changed from long to short.

$$SAR_{long} = SAR_{t-1} + A(H - SAR_{t-1}) \text{ (2-7)}$$

$$where\ A = 0.2 + \sum_{i=2}^{n}\begin{cases}0.2, & if\ C_i > C_{i-1}\\ 0, & otherwise\end{cases} \text{ (2-8)}$$

$$and\ H = \max(C_i),\ for\ i \in (1,\dots,n) \text{ (2-9)}$$

$$SAR_{short} = SAR_{t-1} + A(L - SAR_{t-1}) \text{ (2-10)}$$

$$where\ A = 0.2 + \sum_{i=2}^{n}\begin{cases}0.2, & if\ C_i < C_{i-1}\\ 0, & otherwise\end{cases} \text{ (2-11)}$$

$$and\ L = \min(C_i),\ for\ i \in (1,\dots,n) \text{ (2-12)}$$

Moving Average Convergence Deviance (MACD): The MACD is said to reveal changes in the strength, direction, momentum, and duration of a price trend. The MACD was calculated over 5-minute intervals as well as long-term, over days, to capture short- and long-term market trends.

$$MACD_{12,26,9} = EMA_9(EMA_{12} - EMA_{26}) \text{ (2-13)}$$

Relative Strength Index (RSI): RSI compares the magnitude of recent gains and losses over a specified time period; in this case the period chosen was 14 days and 200 minutes (i.e., 40- of 5-minute periods). The RSI is a measure of speed and change of price movements.

$$RSI = 100 - \left(\frac{100}{1} + RS\right) \quad (2\text{-}14)$$

$$where\ RS = \frac{\sum_{i=1}^{n} \begin{cases} C_i, if\ price\ increase \\ 0, otherwise \end{cases} / n}{\sum_{i=1}^{n} \begin{cases} C_i, if\ price\ decrease \\ 0, otherwise \end{cases} / n} \quad (2\text{-}15)$$

$$where\ \ n = trading\ periods$$

On Balance Volume (OBV): OBV predicts future stock momentum based on volume flow. The theory is that if volume increases sharply without a significant change, the price will eventually jump upward, and vice versa.

$$\begin{aligned} &\text{If } close > close_{[-1]} \text{ then} \\ &\quad OBV = OBV_{[-1]} + volume \\ &\text{elseIf } close < close_{[-1]} \text{ then} \\ &\quad OBV = OBV_{[-1]} - volume \\ &\text{else} \\ &\quad OBV = OBV_{[-1]} \end{aligned} \quad (2\text{-}16)$$

These technical indicators were chosen because they are considered the most popular indicators used by day traders to help make trading decisions and cover a wide range of characteristics regarding price and volume. Together these indicators capture the acceleration, momentum, trend, magnitude, and volatility of price changes both in the short and long term (over 200 minutes and 7-28 days). The trading volume effect is also incorporated to try and create a complete picture of the market when training the algorithms. To avoid including colinear or strongly correlated variables in the models, permutations of volatility, moving average, and various time periods of 7, 14, and 28 days were tested, and the final variables were selected based on the combination that gave the respective model the highest accuracy.

2.3.4.4 Rolling Value Calculations and Lagged Time Periods

Variables representing date and time were created and formatted as categorical variables to improve the model's training speed. The time variables that are created are minute of hour, Hour of day, Day of the week, Day of the month, Day of the year, Week of the year,

and Year.

Following the techniques used in the paper, events, like the price moving up or price changing by greater than 1% in 5 minutes, are encoded as binary variables. Rolling tallies can then be used to track the number of sequential events. These entity encodings create distance measures for categorical variables. These tallies can then, in turn, be encoded as categorical variables to help improve model speed and accuracy when compared with on-hot encoding. The features engineered using this distance-measuring technique are shown in Table 2-2.

Table 2-2: A List of used features using Distance Measuring

Variable	Description
UP DOWN	1 or 0 based on whether the price went up or down in that time period.
Since_UP or Since_DOWN	The number of periods since the price has gone in the respective direction.
Since_XX	The number of periods since the percentage price change was greater than: 0.1%, 0.25%, 0.5%, 1%, or 2%.
Since_XX_UP Since_XX_DOWN	The number of periods since the percentage price change was greater than: 0.1%, 0.25%, 0.5%, 1%, or 2% in the specified direction.

Using rolling calculations over a given number of periods, the following variables capture short-term trends for occurrence rates for events. These variables were calculated as rolling sums or rolling means, turned into integers, and then encoded as categorical variables for performance purposes, as shown in table 2-3. These variables were engineered with the Recurrent Neural Network (RNN) model.

2.3.4.5 ARIMA, Prophet, Random Forest, and MLP Comparative Results

The tested models include ARIMA, Prophet, Random Forest, Random Forest Lagged-Auto-Regression, and Multi-Layer Perceptron (MLP) Neural Networks.

Table 2-3: List of Variables for Short-Term Trend

Variable	Description
Rolling_10_seq_up Rolling_20_seq_up Rolling_10_seq_down Rolling_20_seq_down	Looking back either 10 or 20 periods, how many price movements have been seen for the given direction. $Rolling\_20\_seq\_up_t = \sum_{i=1}^{20} UP_{t-i}$
Rolling_10_volume Rolling_20_volume	The total volume over the last 10 or 20 periods. This is similar to the information given by the OBV. $Rolling\_20\_volume_t = \sum_{i=1}^{20} volume_{t-i}$
Rolling_10_volume_over_max Rolling_20_volume_over_max	This scales the previous calculation using the maximum volume observed over the past 10 or 20 periods. This changes the most when large variations in volume. $Rolling_{20_{volume\_over\_max_t}} = \frac{\sum volume_{t-i}}{\max(volume_{t-i})}$ $for\ i \in \{1, 2, \dots, 20\}$

A. ARIMA

The stationarity of close price returns was tested using the dicky fuller test. After differencing returns once, stationarity was achieved. The final parameters for the model were determined to be of order = (4, 1, 4) when training on the first 60% of data. The best prediction will be the immediate next time period. To get the best prediction power, a sliding window method was employed. Starting with 60% of the data, an ARIMA (4, 1, 4) model was trained. This model was used to predict one 5-minute period into the future. After prediction, the training window was extended to include the next period in the

training data. (The actual return value of the period that had just been predicted). A new model was retrained for each prediction to ensure that predictions were only made one period into the future. Repeatedly retraining the model for every single prediction was very resource intensive. The accuracy was 51.77%. This is technically significantly better than the guessing strategy for this sample size; however, it is also significantly worse than the momentum strategy. It is doubtful that many traders would see 51.77% as a large enough value to guide a trading strategy. Breaking the data into its constituent parts did not reveal any seasonality to the data, so the chosen ARIMA (4, 1, 4) model, without a d parameter, should be appropriate, as shown in Tables 2-4 and 2-5.

Table 2-4: Accuracy of the ARIMA (4, 1, 4) Model using Naïve Strategy

McNemar's Test against Naïve Strategy					
McNemar's chi-squared	10.303			ARIMA(4, 1, 4) Model	
				Correct	Wrong
p-value	0.001328	Naïve Strategy	Correct	8063	8283
			Wrong	7874	8462

Table 2-5: Accuracy of the ARIMA (4, 1, 4) Model using Momentum Strategy

McNemar's Test against Momentum Strategy					
McNemar's chi-squared	111.76			ARIMA(4, 1, 4) Model	
				Correct	Wrong
p-value	< 2.2e-16	Momentum Strategy	Correct	4396	13205
			Wrong	11541	3540

B. Prophet

This model has achieved 52.60% accuracy, which is better than the ARIMA model; however, it still falls short of the momentum strategy by a significant amount ($p < 0.01$) as in Tables 2-6 and 2-7.

The first random model uses technical indicators and embedded encodings from one time period to predict the future time period. This model relies on these variables to capture

the time-based trend in the data since it cannot see lagged variables in the way that the Prophet and ARIMA models can. In an attempt to hack a random forest into working with time-series data, a data frame with time-lagged close prices, and percentage values for high, low, and return values, dating back 40 time periods was created. The accuracies achieved were 50.51% and 50.89% for the Random Forest and the Random Forest with Lagged-Auto-Regression, respectively.

Table 2-6: Accuracy of the Prophet Model using the Momentum Strategy

McNemar's Test against Momentum Strategy					
McNemar's chi-squared	10.73			Prophet Model	
				Correct	Wrong
p-value	0.001054	Momentum Strategy	Correct	9562	8039
			Wrong	7628	7453

Table 2-7: Accuracy of the Prophet Model against ARIMA (4,1,4)

McNemar's Test against ARIMA(4, 1, 4)					
McNemar's chi-squared	93.421			Prophet Model	
				Correct	Wrong
p-value	< 2.2e-16	Naïve Strategy	Correct	8174	7763
			Wrong	9016	7729

C. Random Forest

These results were not statistically any better than the 50/50 guessing strategy. This was not a surprising result, as random forests are not generally used for time-series classification. Random forests have the benefit of being robust to large numbers of variables. For this reason, collinearity was the only main concern when selecting which technical indicators to include in the model. Several random forests were trained, and the selection of variables was chosen based on the combination that provides the highest accuracy on the test set. The final selection of variables include:

- Rolling_20_xxx variables rather than Rolling_10_xxx

- EMA rather than SMA for 7-day and 200-minute periods
- OHLC volatility for 7-days and 200 minutes
- RSI for 14-day and 200-minute periods

Tables 2-8 and 2-9 show that neither the technical indicator model nor the lagged prices model achieved significant results under the threshold of p-value < 0.01 for their respective McNemar tests. These two models seem to perform approximately equal to the guessing strategy; thus, the stock prices are random walks, and it should not be possible to predict their movement.

Table 2-8: Accuracy of the Random Forest Model using Technical Indicators

McNemar's Test against Naïve Strategy					
McNemar's chi-squared	1.5712			Momentum Strategy	
				Correct	Wrong
p-value	0.21	Naïve Strategy	Correct	8280	8066
			Wrong	8227	8109

Table 2-9: Accuracy of the Random Forest Model using Lagged prices

McNemar's Test against Naïve Strategy					
McNemar's chi-squared	4.9522			Momentum Strategy	
				Correct	Wrong
p-value	0.02606	Naïve Strategy	Correct	8345	8001
			Wrong	8286	8050

D. Multilayer Perceptron Deep Neural Network

The model is trained using the same set of technical indicators used by the Random Forest. A random hyper-parameter search was done using H_2O to find the best values for the learning rate, annealing rate, dropout ratio, and the number of hidden layers. The training took 3 days running on an 8-core i7 7700k server. The network achieved the best results with 54.09% accuracy compared to any of the tested models and strategies; however, the accuracy improvement is not significantly higher than the momentum strategy's accuracy

of 53.85% as shown in Table 2-10.

Table 2-10: Accuracy of the MLP Deep Learning Model

McNemar's Test against Prophet Model					
McNemar's chi-squared	14.669			MLP Deep Learning	
				Correct	Wrong
p-value	0.0001281	Prophet Model	Correct	9496	8105
			Wrong	8182	6899
McNemar's Test against Momentum Strategy					
McNemar's chi-squared	0.35464			MLP Deep Learning	
				Correct	Wrong
p-value	0.5515	Momentum Strategy	Correct	9350	7840
			Wrong	8328	7164

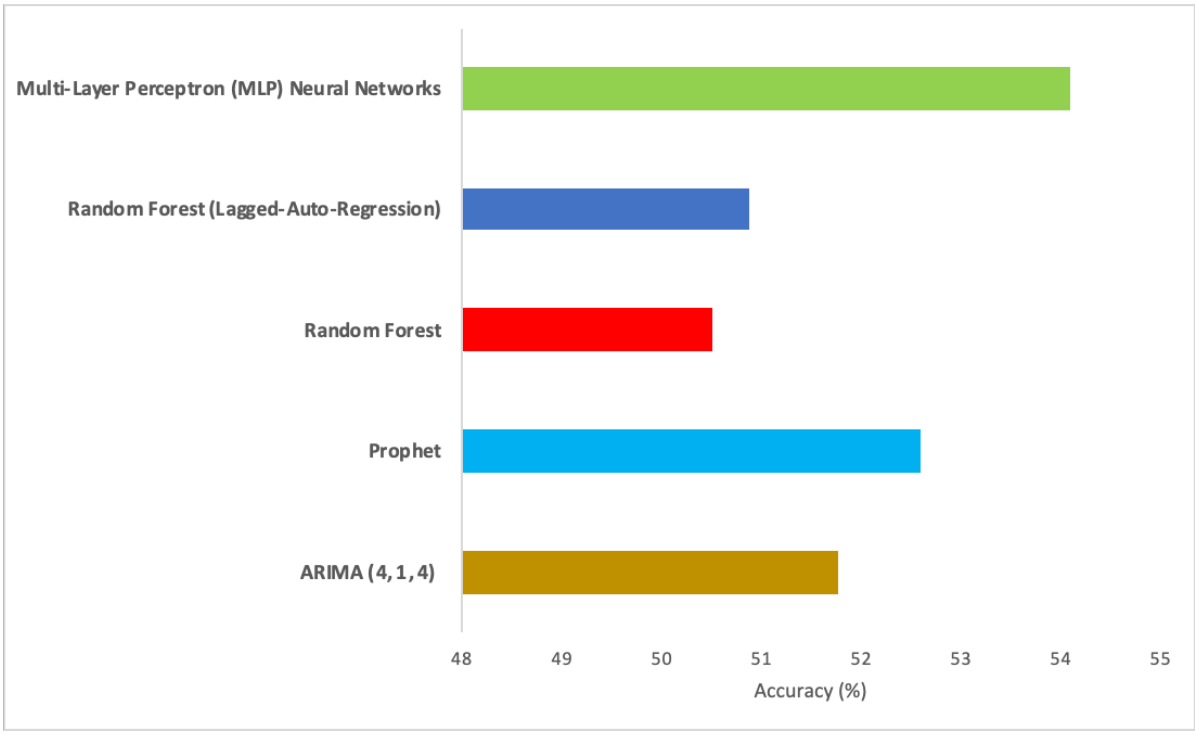


Figure 2-1: Accuracy of the Prediction Models

Figure 2-1. shows a comparison between the Autoregressive Integrated Moving Average (ARIMA), Prophet, Random Forest, Random Forest Lagged-Auto-Regression, and Multi-Layer Perceptron. In conclusion, we can observe that the MLP outperforms all of the tested models, while the Prophet model achieves better accuracy than the ARIMA and Random Forest models.

2.3.5 Conclusion and Future Work

The Bitcoin market mechanics dynamically change with the fluctuation of the financial trade market and the accuracy of the predictive models. Thus, a comparative analysis needs to be completed to flesh out the similarities and differences between various financial investments using cryptocurrencies. As time passes and more data becomes available, it is likely to train a more accurate market movement prediction model. In this paper, various machine learning prediction models are introduced to predict Bitcoin's market movement, i.e., the Up/Down binary classification problem. One of the most significant findings was that the momentum strategy was one of the best-performing models that could guide a Bitcoin trading bot. The best overall model was the MLP deep neural net with 54% accuracy. A slight increase in efficiency would encourage financial traders to gain massive profits when dealing with large numbers of probabilistic events. If a Bitcoin trading bot adopts this strategy for binary options trading on the 5-minute level, there is a potential increase in revenue.

2.3.6 References

The references for this article are detailed in Appendix B.

2.4 The Impact of the Article

This article was published in the "Computers & Electrical Engineering' journal by Elsevier, with an impact factor (IF) of 4.152. On Google Scholar, in June 2023, this article received around 34 citations. In ResearchGate, the article has 322 reads and 30 citations.

2.5 Key Findings in The Article

One of the most significant findings was that the momentum strategy was one of the best-performing models that could guide a Bitcoin trading bot. The best overall model was the MLP deep neural network with 54% accuracy. A slight increase in efficiency would encourage financial traders to gain massive profits when dealing with large numbers of probabilistic events. If a Bitcoin trading bot adopts this strategy for binary options trading on the 5-min level, there is a potential increase in revenue.

2.6 The Contributions of The Chapter

The Bitcoin market mechanics dynamically change with the fluctuation of the financial trade market and the accuracy of the predictive models. Thus, a comparative analysis needs to be completed to flesh out the similarities and differences between various financial investments using cryptocurrencies. In Article 1, various time-series and machine learning prediction models are introduced to predict Bitcoin's market movement, i.e., the Up/Down binary classification problem. This chapter has covered related work and background on trends and price prediction for BTC. The first key research of this thesis presented in Chapter 2 shows the efficiency of using neural network-based models in short-term forecasting, which will be further expanded in Chapter 3 when adopting different data representations while enabling accurate and efficient forecasting.

2.7 The Summary of The Chapter

This chapter aims to provide a comprehensive overview of the current state of knowledge on time-series prediction models and identify gaps in the forecasting market that need to be addressed to pave the way for the discussion in the following chapters and the main

thesis contributions elements. In this chapter, various models have been reviewed, implemented, and compared, including time-series and machine learning-based models such as ARIMA, Prophet (by Facebook), Random Forest, Random Forest Lagged-Auto-Regression, and Multi-Layer Perceptron (MLP) Neural Networks.

For the comparative analysis, the primary data was collected from Coinbase[1]. The data was collected in the form of tick-data – data tracking every individual trade dating back to 2014. Bitcoin data dating back to September 2017 was also collected from the popular exchange Poloniex[2] to create a second test set in the format of 5-min Open High Low Close (OHLC) plus volume. As Bitcoin trading resembles stock trading, some stock data was collected to test further the predictive power of the models on data for which they had not specifically been trained. The 5-min OHLC data for Apple, Facebook, Google, and Microsoft stocks was downloaded from the Google Finance API, dating back to January 2018.

For data transformation, the Bitcoin tick-data data was transformed into 5-min intervals, and Back-filling was used to fill prices for any 5-min period with no trading activity. After the transformation, the data had opening, high, low, and closing price variables, as well as the trade volume for those 5 min. To further augment the data, new variables were engineered using the most popular indicators used in the stock trading industry, such as Price Volatility (including Close-to-Close volatility, High-Low volatility, and OHLC volatility), Moving Average (including Smoothing Moving Average (SMA) and Exponential Moving Average (EMA)), Stop-and-Reverse (SAR), Moving Average Convergence Deviance (MACD), Relative Strength Index (RSI), and On Balance Volume (OBV). These technical indicators (TI) were selected because they are considered the most popular indicators used by day traders to help make trading decisions and cover a wide range of characteristics regarding price and volume.

---

1 – Coinbase: https://www.coinbase.com/
2 – Poloniex: https://poloniex.com/support/api/

A detailed discussion and definition of each of these indicators are presented in Article 1. Variables representing date and time were created and formatted as categorical variables to improve the model's training speed.

The experimental work shows that the AIRMA has archived an accuracy of 51.77%, and The Prophet model has an accuracy of 52.60%. The accuracies achieved were 50.51% and 50.89% for the Random Forest and the Random Forest with Lagged-Auto-Regression, respectively. Finally, the MLP network achieved the best results with 54.09% with a proper choice of the learning rate, annealing rate, dropout ratio, and the number of hidden layers.

Chapter 3 – The Proposed Methodology, Research Gap, and Novelty

This thesis aims to understand the market mechanics of Bitcoin and develop prediction models to assist traders in making informed decisions in the volatile cryptocurrency market. The methodology employed in this thesis involved a multi-article approach to investigate various aspects of the Bitcoin market. Each article contributed to the overall understanding of market mechanics and the development of prediction models.

3.1. Research Gap

The literature surrounding the cryptocurrency market, particularly Bitcoin, has witnessed significant growth in recent years. However, several research gaps remain that this thesis aims to fulfill. One notable gap pertains to the need for comprehensive analysis that incorporates both structured and unstructured data sources. While traditional financial markets have well-established models for analysis, the unique characteristics of the cryptocurrency market, including its inherent volatility and influence of social media, require novel approaches. Existing research often focuses on either structured data, such as trading data, or unstructured data, such as social media posts, without effectively integrating the two. This thesis addresses this gap by developing prediction models that leverage both types of data, providing a more comprehensive understanding of Bitcoin's market mechanics.

Another research gap in the literature relates to the challenges posed by market crashes and their impact on cryptocurrency price movements. The COVID-19 pandemic highlighted the vulnerability of the cryptocurrency market to external shocks. However, existing prediction models often fail to adequately capture the dynamics of market crashes. This thesis aims to fill this gap by developing robust models that handle market crashes and provide accurate forecasts during periods of extreme volatility. By incorporating ensemble learning techniques and unsupervised sentiment analysis of social

media data collected during the pandemic, this research contributes to a better understanding of how market sentiment and external events impact Bitcoin's price movements.

3.2. The Proposed Methodology

The research process in this thesis is comprised of the following key methodology steps:

- **Comparative Analysis**: The initial methodology involved conducting a comprehensive literature review to gather existing knowledge about time series prediction models and their application to cryptocurrency markets. This review served as the foundation for selecting appropriate models for further analysis. A comparative analysis was performed to assess the performance of different models and identify the most effective ones [Article 1].
- **Analysis of BTC Market Mechanics:** We considered a range of exogenous variables to simulate past BTC prices and understand the significant drivers of price changes. To gain insights into the underlying factors influencing Bitcoin price movements, vector autoregression (VAR) and Bayesian vector autoregression (BVAR) models were employed [Article 2].
- **Modified Deep Learning Model for Time Series Modeling**: Recognizing that traditional time series modeling techniques may not capture all patterns, a modified deep learning model using RESNET was proposed. The main methodology is targeted at uncovering subtle and potentially undetectable patterns within images of time-series data charts. By leveraging these patterns, the model demonstrated promising results for improving the accuracy of Bitcoin price prediction [Article 3].
- **Efficient Forecasting Model Using Ensemble Learning and Social Media Data**: The impact of social media, particularly during market crash periods like the COVID-19 pandemic, was explored. An efficient forecasting model using a Composite Ensemble Prediction Model (CEPM) that utilizes sentiment analysis to make predictions was developed, incorporating ensemble learning techniques and analyzing Twitter posts. This model effectively handled unstructured data and

provided accurate BTC trend analysis during the pandemic [Article 4].

- **Unsupervised Sentiment Model Using Consensus Clustering and Social Media Data:** To address the challenge of limited labeled data for sentiment analysis, an unsupervised sentiment model using sentiment consensus clustering (SCC) was proposed. This model utilized consensus clustering and examined Twitter posts collected during the COVID-19 timeframe. By uncovering underlying sentiment in online posts, the model provided valuable insights for forecasting early Bitcoin movements following the outbreak [Article 5].

3.3. Statement of Novelty

The thesis contributes novel methodologies, models, and insights to the literature. The novel approaches presented in this thesis fill existing research gaps and provide valuable tools for traders operating in the volatile cryptocurrency market.

- The main novelty of this thesis lies in its contributions to the understanding and prediction of Bitcoin's price movements. Firstly, the thesis proposes a modified deep-learning model that recognizes subtle patterns within images of time-series data charts. This approach represents a novel method for time series modeling and provides insights that may not be apparent using traditional numerical feature-based techniques. By leveraging image-based representations, the model enhances the forecasting process and improves the accuracy of Bitcoin price predictions.
- Secondly, the thesis tackles the challenge of handling both structured and unstructured data for cryptocurrency analysis. By developing efficient prediction models that effectively utilize both types of data, this research fills a critical gap in the literature. The integration of structured data, such as trading data, with unstructured data, such as social media posts, allows for a comprehensive analysis of Bitcoin's market mechanics. This novel approach enables a deeper understanding of the factors driving price movements and provides traders with more informed decision-making tools.
- Furthermore, the thesis addresses the specific challenge of market crashes, as

exemplified by the COVID-19 pandemic. Existing prediction models often struggle to handle extreme market volatility and the influence of social media during such periods. The thesis develops robust models that successfully analyze social media data and provide accurate BTC trend analysis during market crashes. By incorporating ensemble learning techniques and unsupervised sentiment analysis, the models capture the underlying sentiment in online posts and offer valuable insights into early Bitcoin movements following significant events.

Chapter 4 – Simulating the Bitcoin Market

4.1 The Objective of The Chapter

In Chapter 2, we have summarized the state-of-the-art time series and machine learning-based algorithms for predicting the BTC movements in the short term. However, several drivers are impacting the Bitcoin market, such as the total number of Bitcoins available, the difficulty of Bitcoin mining, and the average blockchain size that needs to be analyzed. Therefore, determining essential endogenous and exogenous drivers in BTC markets is critical. In this chapter, the BTC market is simulated to determine the optimal set of endogenous (independent) and exogenous (dependent) variables to draw insights into how one could capitalize on this asset over time. This chapter aims to present a feature-selection strategy for identifying influential factors in the cryptocurrency market to improve the accuracy of BTC prediction models.

4.2 Published Article 2

Ibrahim, A., Kashef, R., Li, M., Valencia, E., & Huang, E. (2020). Bitcoin network mechanics: Forecasting the Bitcoin closing price using vector auto-regression models based on endogenous and exogenous feature variables. Journal of Risk and Financial Management, 13(9), 189, https://doi.org/10.3390/jrfm13090189

4.3 The Article Body of Knowledge

The subsequent sections are directly excerpted from the paper titled "**Bitcoin network mechanics: Forecasting the Bitcoin closing price using vector auto-regression models based on endogenous and exogenous feature variables."** All credits and rights are attributed to the original authors and the source publication.

4.3.1 Introduction

Bitcoin (BTC) is a digital currency alternative to real currency and is the most popular among cryptocurrencies. The BTC was created by a cryptologist known as “Satoshi Nakamoto”, whose real identity is still unknown (Nakamoto 2014). As blockchain currencies are not as liquid as other forms of currency, understanding the behavior of this market draws insights as to how one could capitalize on this asset over time. Especially as society becomes more digitally inclined, the viability of a blockchain currency such as BTC to become a common currency seems like a possible reality. There are both winners and losers in the context of each capital market transaction. There are several drivers impacting the Bitcoin market, such as the total number of Bitcoins available, the difficulty of Bitcoin mining, and the average blockchain size. Therefore, determining the essential endogenous and exogenous drivers in BTC markets is a critical task. Each of these endogenous and exogenous variables can be treated as a time series, and therefore, suitable multivariate time series forecasting models are needed.

Vector autoregression (VAR) is one of the most widely used stochastic process models to analyze interdependencies of multivariate time series, and it has proven to be a useful model for describing the behavior of economic and financial time series and for forecasting (Campbel et al. 1996). The VAR model is an extension of the univariate autoregression model to multivariate time series data. In the VAR structure, each variable is a linear function of past lags of itself and the past lags of the other variables. However, the limited length of standard economic datasets may produce over-parameterization problems (Koop and Korobilis 2009); thus, the Bayesian vector autoregression (BVAR) model was introduced in (Litterman 1980) to solve this problem. The BVAR model uses Bayesian methods to estimate a vector autoregression. In comparison with the standard VAR models, the BVAR model treats input parameters as random variables, and prior probabilities are then assigned. A feature selection of the cryptocurrency drivers is needed to enhance the performance of a multivariate time-series (e.g., BTC) prediction model. In this paper, we applied direct forecasting using VAR and BVAR models to simulate the BTC market to understand the behavior of market participants as well as their most and least

favorable market conditions according to the closing price of BTC based on an optimal set of exogenous variables. The simulated BTC market includes forecasting the endogenous variables, such as the equilibrium closing price of the market for BTC as denominated by the US dollar (MKPRU), the number of unique MyWallet users (MWNUS), and the total BTC available in the market to date (TOTBC). Experimental analysis over 7-year and 10-year timeframes shows the efficiency of the VAR and BVAR models in predicting the set of endogenous variables compared to traditional autoregression and Bayesian regression models using the optimal selected set of exogenous variables. The rest of this paper is organized as follows: Section 2 introduces the background of Bitcoin; Section 3 focuses on the related work; Section 4 describes the prediction models for Bitcoin closing price; Section 5 presents and discusses the results of the prediction models; and Section 6 outlines the conclusions and future works.

4.3.2 Background on Bitcoin

Bitcoin is a unique digital currency with the potential to change the nature of the transactions that people conduct in digital space. Bitcoin enables consumers for the first time to make electronic transactions from person to person without the need for an intermediary between them, like cash ((Brito 2014). Transactions conducted in the digital space with BTC allow individuals to push payments directly to the merchants without having to share personally identifiable information, which could be intercepted by cybercriminals for fraud. One of the greatest concerns for BTC as a commonly accepted currency is security, as there is no intermediary to ensure the coverage of stolen BTC, should theft occur (Brito 2014). As the value of the asset appreciated 63% YTD in 2016, and 87% YTD in 2020, identifying historical patterns of behavior could help in understanding how the BTC security (and the security of similar cryptocurrencies) is likely to behave from inception.

4.3.2.1 Bitcoin Ledger

Each block in the Bitcoin blockchain contains a summary of all transactions in the block using a Merkle tree (aka binary hash tree) such that each transaction is first put into a pool

of pending transactions. Then, they are put into the transaction chain (blockchain) (Antonopoulos 2014). Each block is linked in a chain by a reference to a previous header hash in which the addition of a transaction into the chain is through a “mathematical lottery” (United States Securities and Exchange Commission 2017). The miner solves the math problem (cryptographic hashing) and puts the transaction into the chain. The math helps everyone with a wallet know the order of transactions as well as all past transactions.

4.3.2.2 Bitcoin Development Process

As other cryptocurrencies aim to perform the same computer-distributed task, there are risks that any new digital currency faces from inception until maturity. There are three primary characteristics that a digital currency must satisfy to be deemed a sound form of currency. The following are the key success factors (Barski and Wilmer 2015):

- The network effect;
- Cryptocurrency volatility;
- Cryptocurrency-pegging technology.

A. The Network Effect

The simple concept of money is that people will be willing to use the currency (medium of exchange) so long as someone else is willing to accept it as a form of payment. Without an appropriate network for the payment mechanism, it is unlikely that people will desire to use the specific cryptocurrency if it turns out to be illiquid.

B. Cryptocurrency Volatility

For any cryptocurrency that is getting newly established as a payment method, the “fair” established value must be stable for consumers to be comfortable purchasing with the digital currency. As BTC is a newly available asset, the price discovery mechanism requires that the group of buyers and sellers exchanging the currency come to an agreed-upon value for the underlying asset (Pagnottoni and Dimpfl 2019). As the value of a BTCUSD in November 2016 was roughly $740, the currency was far from stable at the time. Seeing prices as high as $1200 in 2013, $15,000 in 2020, and as low as $355 in 2016 for BTCUSD, a true concern for consumers is to make a purchase with an asset that has varied

so much in value. However, there are many cases of money being just as volatile. One famous example is the Zimbabwe hyperinflation, where the currency experienced 80 billion percent inflation in a single month.

D. Cryptocurrency-Pegging Technology

As the supply for the total BTC is limited to 21,000,000, more users have begun to use the BTC, which has modestly reduced volatility. The advantage of BTC over other cryptocurrencies is that it has been established and generated credibility for a sufficient network of users to adopt the use of the coin. Primarily, this has helped BTC outpace other digital currencies to normalize volatility. For any potential new e-coin that could enter the cryptocurrency market, it would make sense that the coin merges its stability according to a more stable cryptocurrency such as BTC.

4.3.2.3 Market Participants

The following are the market participants worthy of further analysis, accompanied by a brief description of their role in the market:

- Miners—The market participants who are proactively adding transaction records to Bitcoin's public ledger of past transactions or blockchain and fueling the supply of BTC.
- Individual investors—Investors for the digital assets to purchase goods or services with the digital currency.
- Payment mechanism—Conduct business internationally as international payments are now available via BTC.
- Retail investors—Funds that are likely to pick up the currency as a portion of their portfolio to hedge, like gaining exposure to traditional currency markets.

4.3.2.4 Stakeholders

As digital currency changes the evaluated value of money and other financial assets, several stakeholder requirements and motives should be considered. The following are the stakeholders (formal and informal) affected by the adoption of cryptocurrencies: savers/bullish investors, government, other cryptographers, BTC exchanges/brokers, illegal black markets, BTC miners, and members of the public. As stakeholders desire

stability and strength with any medium of transaction, some stakeholders are opposed to the widespread adoption of BTC. Specifically, the government and other cryptographers may have an issue with the widespread adoption of the BTC as decentralized digital money where no government or single entity can control the price or value.

4.3.3 Related Work

In modeling and simulation of the economics of mining in the Bitcoin market (Cocco and Marchesi 2016), authors have discussed how a miner is impacted by BTC prices (Cocco and Marchesi 2016). The goal of this artificial market model is to model the economy of the mining process from the inception of the Graphics Processing Units (GPU) generation. The important findings for this computational experiment encompass the ability to reproduce the unit root property, the fat tail phenomenon, and the volatility clustering of the BTC prices (Cocco and Marchesi 2016). Research on Bitcoin price forecasting is mainly based on two approaches: machine learning and time series methods.

4.3.1.1 Machine Learning Prediction Methods

Felizardo et al. (2009) presented a comparative study of price prediction performance among several machine learning models: long short-term memory (LSTM), WaveNet, support vector machine (SVM), and random forest (RF). The results indicated that for time-series data, the LSTM model tends to perform better than other machine learning models. The research of Tandon et al. (2009) gave a similar conclusion. They applied three different machine-learning methods to forecast the Bitcoin price and compared their prediction ability. As a result, the RNN (recurrent neural network) with LSTM gave a lower mean absolute error than the random forest and linear regression models. Much research focuses on improving the LSTM model to increase forecasting accuracy. Wu et al. (2018) proposed an LSTM called LSTM with AR(2) model to forecast Bitcoin's daily price. The conventional LSTM model only considers the previous price to predict the current price; instead, the LSTM with AR(2) takes the previous two days' prices into account. The experimental results demonstrated that the proposed model with AR(2) achieved better forecasting accuracy with a lower mean squared error. Hashish et al. (2019) proposed the

addition of hidden Markov models (HMMs) to the conventional LSTM. The HMM was used to describe the historical movements of Bitcoin. The proposed hybrid of HMM and LSTM outperformed the traditional forecasting of LSTM by decreasing the mean squared error from 49.089 to 33.888. The main drawback of the machine-learning models is that they need high computational capacity, so the execution time of the forecasting process is very time-consuming. Thus, in this paper, we focus on time-series prediction models. Support vector machine, latent source, and multilayer perceptron models work better for classification problems. The LSTM model performs well in solving long-term dependency problems, which means it is suitable for price prediction. However, the LSTM model needs a long computation time and has a large memory requirement.

4.3.3.2 Time-Series Prediction Methods

Bakar and Rosbi (2017) proposed the autoregressive integrated moving average model (ARIMA) to forecast the exchange rate between Bitcoin and the US dollar. In this method, the upcoming price depends upon autoregression, integration, and moving average, respectively. They believed the ARIMA model could be a reliable model for forecasting the volatile characteristics of Bitcoin. Both Roy et al. (2018) and Anupriya and Garg (2018) applied the ARIMA model to predict Bitcoin's price. The experimental result demonstrated the strong forecasting ability of the ARIMA. The mean error between the actual prices and the predicted prices was less than 6% for most values. Roy et al. (2018) also compared the performance of the ARIMA model with the autoregressive model (AR) and moving average model (MA), and the ARIMA model resulted in better accuracy than the other two models. However, the ARIMA model's shortcoming is that it can give a more accurate prediction for short-term data based on the research result of Ariyo et al. (2014). Rane and Dhage (2019) introduced nine approaches for Bitcoin price prediction and discussed each methodology in their research. The ARIMA model targets to forecast uncertainty time-series data within a short-term period, but class imbalance can bias it. Linear regression is unsuitable for predicting Bitcoin prices as time series data.

The strength of the vector autoregression (VAR) model and the Bayesian vector autoregression (BVAR) model in estimating currency and exchange rate fluctuations has

been demonstrated in recent research. VAR has been used widely by financial theorists and economists in predicting time series economic variables in systems that involve supply and demand (Ito and Sato 2006; Wang et al. 2017; Carriero et al. 2009; Alquist et al. 2013; Sims 1993). We found several papers that use VARs to estimate currency and exchange rate fluctuations, notably Koray and Lastrapes, who use a VAR model to estimate the exchange rate on a series of macroeconomic variables (Koray and Lastrapes 1989). Additionally, Ito and Sato performed VAR research on the exchange rate of post-crisis Asia (Ito and Sato 2006). Wang et al. (2017) established a VAR model to analyze the impact of exchange rate volatility on economic growth. Furthermore, there is some research on forecasting using the Bayesian vector autoregression (BVAR) method. For example, Carriero, Kapetanio, and Marcellino demonstrated that the BVAR model produced better forecasting for exchange rates (Wang et al. 2017). In the econometric/finance community, (Catania and Ravazzolo 2019) and (Bohte and Rossini 2019) have studied the forecasting performance of cryptocurrencies by vector autoregression with and without time-varying volatility. (Bianchi, forthcoming) has investigated the possible relationship between returns on cryptocurrencies and traditional asset classes. Bianchi et al. (2020) discussed the relationship between the returns on stable-coins and major cryptocurrency pairs within the context of a large Bayesian vector autoregression model. The BVAR model extends the classical VAR model by using Bayesian methods to estimate a vector autoregression. The BVAR model treats input parameters as random variables, and prior probabilities are then assigned. Current related work to both VAR and BVAR models in forecasting BTC prices does not focus on selecting the set of endogenous and exogenous variables and drivers that control the BTC market, which is the primary focus of this paper.

4.3.4 BTC Closing Price Prediction Models

Both VAR and BVAR models are used in this paper to forecast the Bitcoin price and simulate the BTC market to understand market participants' behavior as well as the market conditions according to the closing price of BTC.

4.3.4.1 Endogenous and Exogenous Variables

An autoregressive model is typically used to develop predictions and understand the trend of a time series. However, in financial and economic data, several factors are affecting the time series, such as supply, demand, and regulation. The complex nature of any financial market warrants a more sophisticated model. The performance of the VAR and BVAR forecasting models depends on the optimal selection of the set of endogenous variables of interest. Several variables were tested as proxies to represent the price, demand, and supply of the BTC market, respectively, after trying out numerous iterations of VARs and BVARs and using sensitivity analysis with different variables, lags, and time frames. The final set of endogenous variables is defined in Equation (4-1). Let Y_t be a vector of the endogenous variable of interest such that:

$$Y_t = \{\text{MKPRU, MWNUS, TOTBC}\} \quad (4\text{-}1)$$

$$t_1 = [\text{04-01-2009, 22-11-2016}] \quad (4\text{-}2)$$

$$t_2 = [\text{01-01-2011, 01-08-2020}] \quad (4\text{-}3)$$

where MKPRU represents the equilibrium closing price of the market for BTC as denominated by the US dollar (Figures 2-1 and 2-2). MWNUS is the number of unique MyWallet users, and TOTBC is the total BTC available in the market to date, as there is a limited amount of BTC available at 21,000,000. Our time frames are across two intervals, the first one is [04-01-2009, 22-11-2016] (Figure 2-1), and the second period is [01-01-2011, 01-08-2020] (Figure 2-2). The decision-making process uses reasonable metrics deemed viable drivers of the endogenous variables, where the following were selected as the exogenous variables: Average Block Size in, MB (AVBLS), Bitcoin Difficulty (DIFF), Number of Transactions per Block (NTRBL), Miner's Revenue (MIREV), Change in the Number of unique addresses (NADDU), Total Output Volume (TRVOU), and Hash Rate (HRATE). A majority of the factors selected were those that had been a result of the BTC network's transaction behavior and how the fundamental mechanics influenced the closing price. The variables AVBLS, DIFF, TRVOU, and HRATE were taken as the variables that dictated the difficulty of accessing and supplying BTC to the market. NTRBL considers

the growing number of transactions occurring per block of BTC as a measure of transaction volume per available block of BTC. The NADDU variable considers the changing number of unique addresses performing BTC transactions to understand behavior trends over time. TRVOU measures the exchange trade volume of USD within the BTC market, which serves as a guideline as to how the market reacts to changes in value when buying or selling BTC. Finally, X_t, as the list of exogenous variables, is defined in Equation (4-4) as:

$$X_t = \{\text{MIREV, NTRBL, AVBLS, DIFF, NADDU, TRVOU, HRATE}\} \quad (4\text{-}4)$$

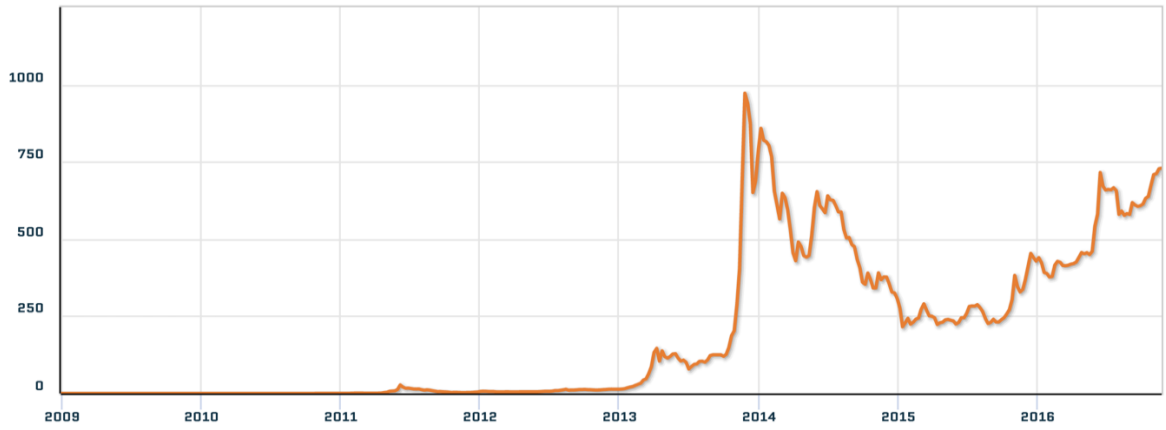


Figure 4-1: Bitcoin closing price in USD (MKPRU), [04-01-2009, 22-11-2016]

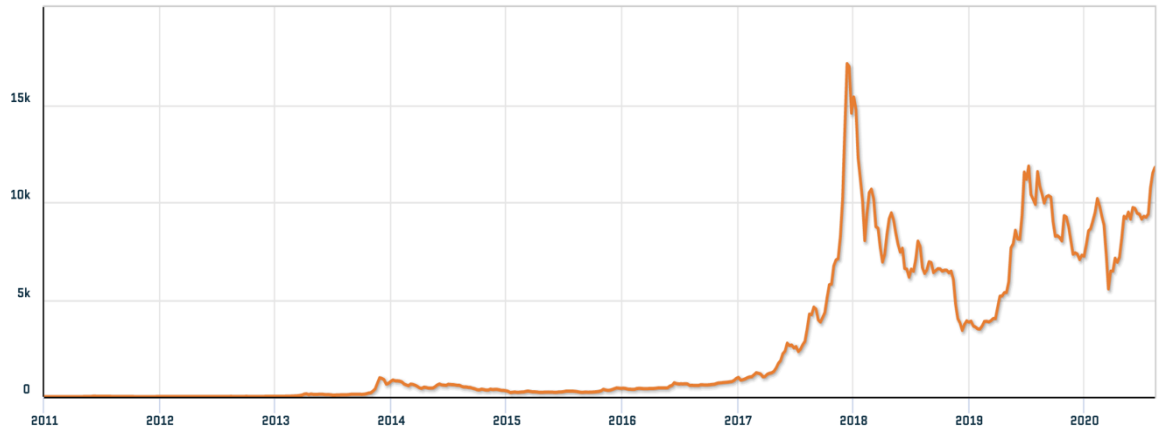


Figure 4-2: Bitcoin closing price in USD (MKPRU), [01-01-2011, 01-08-2020]

4.3.4.2 Vector Autoregression (VAR) model

A Vector autoregression (VAR) (Sims 1993) , (Kuschnig et al. 2020), and (Kuschnig and

Vashold 2019) model was developed to understand the relationship between the system of variables that are of interest (Equations (4-1) and (4-4)). Thus, the VAR of interest is as follows:

$$Y_t = \beta_0 + \beta_1 Y_{t-1} + \beta_2 Y_{t-2} + \beta_3 Y_{t-3} + \cdots \beta_n Y_{t-n} + \beta_{n+1} X_t + \epsilon_t \quad (4\text{-}5)$$

where the betas ($\beta_t's$) are vectors of constants and coefficients representative of the relationship between the variables, where n is the number of lags used in the VAR model. The purpose of selecting this model is to use the model coefficients to simulate a certain period of BTC endogenous variable (Equation (4-1)) given the exogenous variables (Equation (4-4)). Furthermore, one could ideally forecast out the BTC price behavior over time, such that there are verified and validated forecasts of the exogenous variables.

A. Model Assumptions

A few assumptions were made in this VAR model in an effort to use real market data to forecast just over six months. First, the model assumes that the relationship between the variables is static. A variety of timelines were tested accordingly in order to understand differences in behavior. The following are the timeframes selected for analysis:

Experiment A: Full timeframe: [04-01-2009, 22-11-2016], Post-boom timeframe: [10-12-2013, 22-11-2016], the Year of 2016 timeframe: [01-01-2016, 22-11-2016]. Experiment B: Full timeframe: [01-01-2011,01-08-2020], Post-boom timeframe: [01-01-2017, 01-08-2020], the Year of 2020 timeframe: [01-01-2020, 01-08-2020]. For both Experiments A and B, the second assumption made in the model was the segregation of endogenous and exogenous variables. The decision-making process yielded a qualitative and intuitive measure for the variables.

B. Model Validation and Verifications

The process of validating the model was among the most difficult tasks throughout the entire process. Ultimately, the selected set of endogenous variables contained the BTC exchange rate, a variable for supply, and a variable for demand. Collectively, these variables help represent the market mechanics of Bitcoin. Based on the selected endogenous and exogenous variables, the following parameters were used:

- lag.max=366—to accommodate a full year of seasonal behavior and trends;
- type='both'—to evaluate the deterministic regressors.

The resulting selection timeframe was selected according to the Akaike Information Criterion (AIC), Schwarz Criterion (SC), Hannan Quinn (HQ), and Forecast Prediction Error (FPE). This screening process served as a deterministic selection of the timeframe for the forecasting by encompassing summary statistics such as p-value and R2 to verify the accuracy of the relationship that was being estimated. Additionally, other combinations of variables were attempted with exceptionally poor results. Most of the other variables that were included as an aggregate to those used in the model projected dramatic market crashes with negative asset value.

4.3.4.3 Bayesian Vector Autoregression (BVAR) Model

The classical VAR model may have over-parameterization problems because of the large number of parameters and limited availability of time-series datasets (Sims 1980); alternatively, the Bayesian vector autoregression model can be used. The BVAR model applies Bayesian methods to estimate a VAR and treats the VAR model parameters as random variables. It also assigns and updates the prior probabilities of both observed and unobserved parameters based on available data (Miranda-Agrippino and Ricco 2018). The BVAR model in this paper uses the same variables of interest in the VAR model as described in Section 4.2. Let Y_t be a list of variables used in this BVAR model, such that:

$$Y_t = \{\text{MKPRU, MWNUS, TOTBC}\} \quad (4\text{-}6)$$

As in the VAR model, the BVAR model also assumes the chosen variables have static relationships and uses several different timelines to observe forecasting outputs. The BVAR model uses the same timeframes (Experiment A and Experiment B) used in the VAR model in order to compare their forecasting abilities.

Prior Specification

In the BVAR model, the informative prior probability distribution of the VAR coefficients ($\beta_t's$ in Equation (4-5)) can be assigned before observing the sample data. The Minnesota

prior was introduced and developed by Robert Litterman and other researchers at the University of Minnesota (Litterman 1980) and was chosen in our BVAR model. This prior is based on the behavior of most macroeconomic variables, which is approximately a multivariate random walk model with drift. The parameters of the Minnesota prior are set as follows:

- Parameter λ with max = 5 and min = 0.0001, to control the tightness of the prior.
- Parameter α with max = 3 and min = 1, to manage variance decay with increasing lag order.
- var = 10,000,000, to set the prior variance on the model's constant.

4.3.5 Experimental Analysis

Real datasets of the Bitcoin market in three different timeframes were used in this paper across two different time periods, Experiment A, t1 = [04-01-2009, 22-11-2016], and Experiment B, [01-01-2011, 01-08-2020]. For Experiment B, the data were normalized using the logarithm of each return variable. Both the VAR and BVAR models were applied and tested on these datasets to forecast the Bitcoin market price. The forecasting results were analyzed to evaluate the performance of our models.

4.3.5.1 Experimental Dataset

The primary source of data and information was the Quandl Dataset, which was sourced from Blockchain.com (Quandl 2020). The source contains up to 32 datasets, including the BTC market price. Each dataset contains a time series for a variable. The secondary dataset was the average OHLC (open-high-low-close) candlestick values across multiple exchanges scraped from Bitcoin charts.com (Bitcoin Charts 2020). Additionally, any of the transforms accepted were denoted upfront before the variable. In the circumstance of the BTC simulation, the Quandl transform applied was "diff", which implied the change over time depending on the frequency (i.e., daily frequency data would be sampled as daily frequency change of that variable). The OHLC candlestick chart data (Figure 4-3) were sourced directly from Bitcoin charts.com, consolidating the average OHLC candle according to a number of varying exchanges that trade BTC and similarly pegged altcoins. One of the major difficulties encountered upon sourcing the data was to get a consistent

market price from BCHAIN, which would match the OHLC charts sourced. The difference appeared to be according to when the different data sources selected their end-of-day settlement. Bitcoin charts.com was selected, as the close price difference was roughly around ($1–$2).

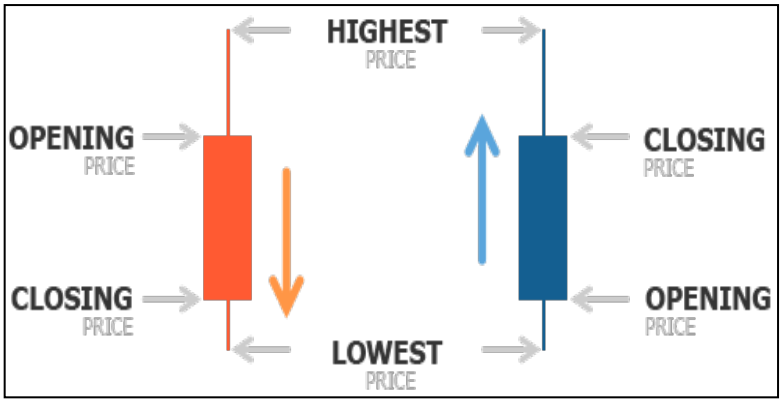


Figure 4-3: OHLC (open-high-low-close) candlestick

4.3.5.2 Forecasting Results

Both VAR and BVAR models were tested with three timeframes in two different experiments. Experiment A: For the 2016 timeframe, values of variables described in Sections 4.1.1 and 4.2.2 between 01-01-2016 and 30-09-2016 were imported as input to the two models. For the Post-boom timeframe, data from 10-12-2013 to 30-09-2016 were imported as input. Both models forecasted the Bitcoin price in USD for the period 01-10-2016 to 30-10-2016 and compared the forecasting results with the actual Bitcoin price. For the Full timeframe, the time period selected to forecast was the last 199 days [05/08/2016–11/22/2016] to evaluate the effectiveness of these two models. Experiment B: For the 2020 timeframe, input and output variables between 01-01-2020 and 01-08-2020 were used for both the VAR and BVAR models. For the Post-boom timeframe, data from 01-01-2017 to 01-08-2020 were used. For the Full timeframe, the time period selected for forecasting was the last six months [01-02-2020, 01-08-2020] to evaluate the effectiveness of these two models.

A. Results of the VAR Model: Experiment A

The model selects the most suitable coefficients, where the outcome minimizes FPE.

Figures 4-4, 4-5, and 4-6, respectively, show the evaluation of the Full, Post-boom, and the Year of 2016 timeframes forecasting in comparison to the BTCUSD OHLC candle from Bitcoin charts.com, where “fcst” is the forecasted closing price, “lower” is the lower bound (95% CI), and “upper” is the upper bound (95% CI). The endogenous variables were simulated from the estimated VAR, as shown in Figures 4-7, 4-8, and 4-9 for three different timeframes. The simulated exogenous variables were the real datasets taken from Quandl for the aforementioned timeframe. Ultimately, by evaluating the results of different timeframes, the full timeframe using the VAR model showed the best forecasting performance. The Full timeframe represents the most data available and incorporates the relationships over different timeframes. Although the significance of the relationship between these variables may change over time, the 7-year timeframe surely aided in modeling the market behavior.

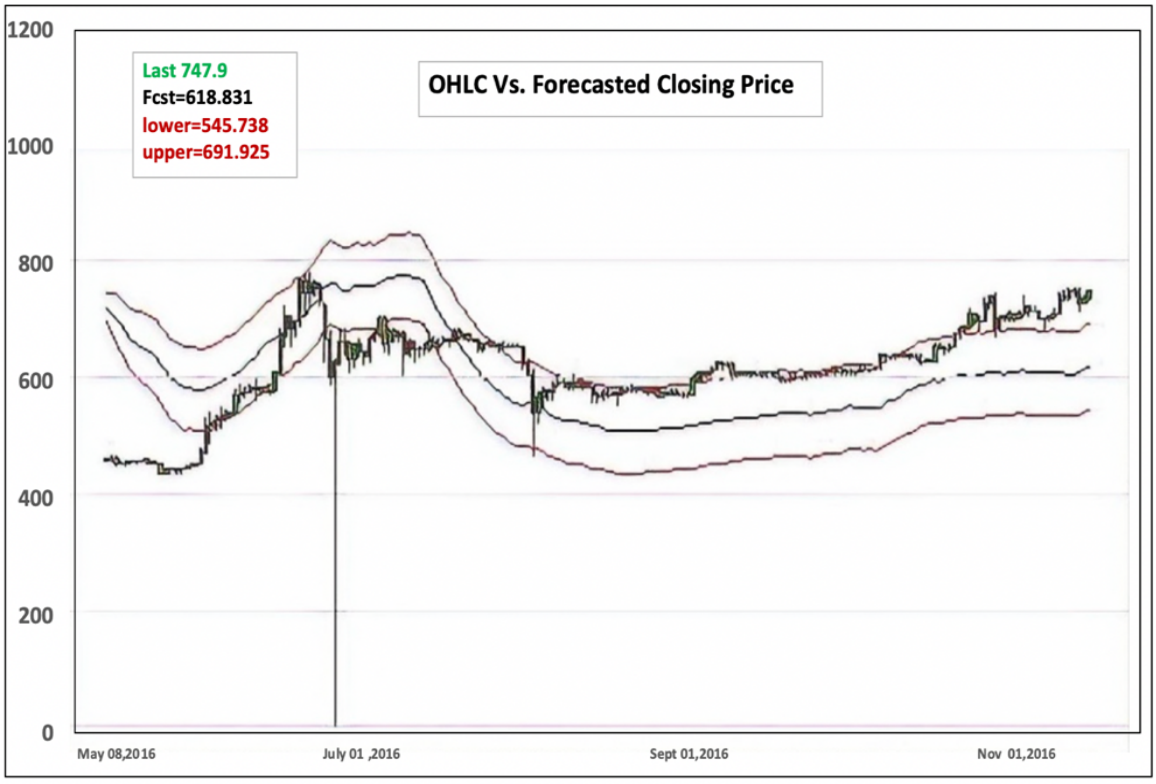


Figure 4-4: Forecasting Bitcoin closing price using Full timeframe. Data Vs. BTC OHLC

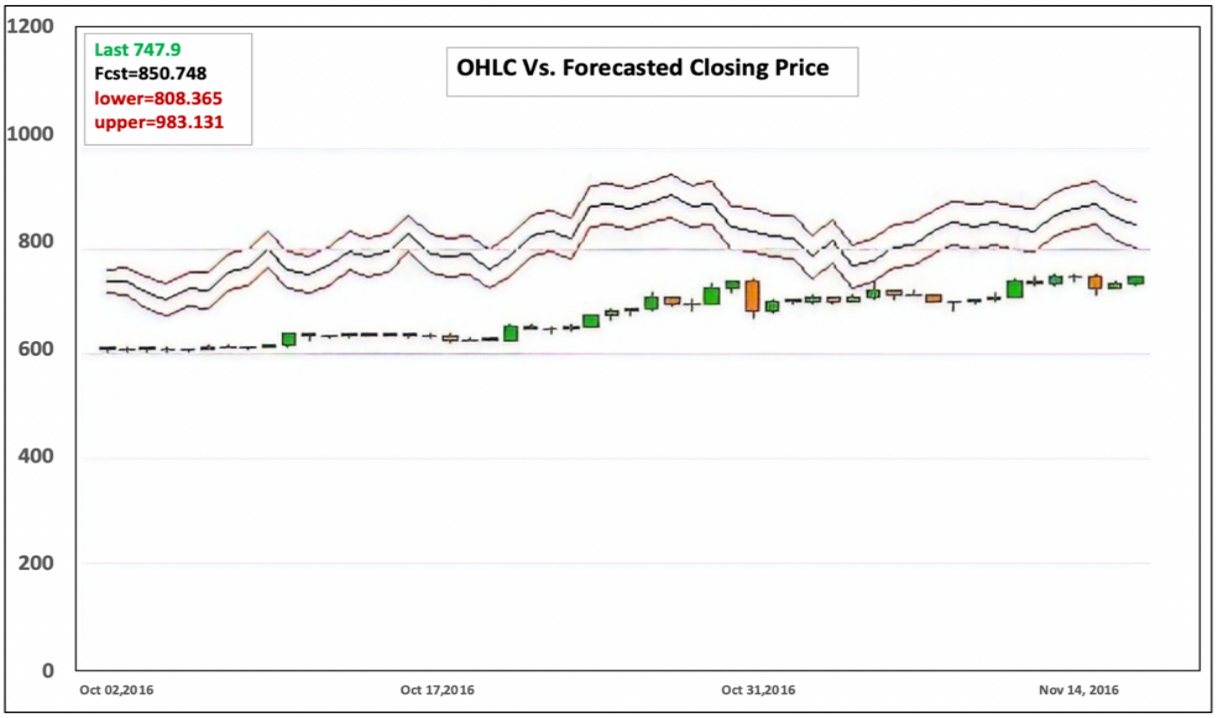


Figure 4-5: Forecasting Bitcoin closing price using Post-boom timeframe. Data Vs. BTC OHLC

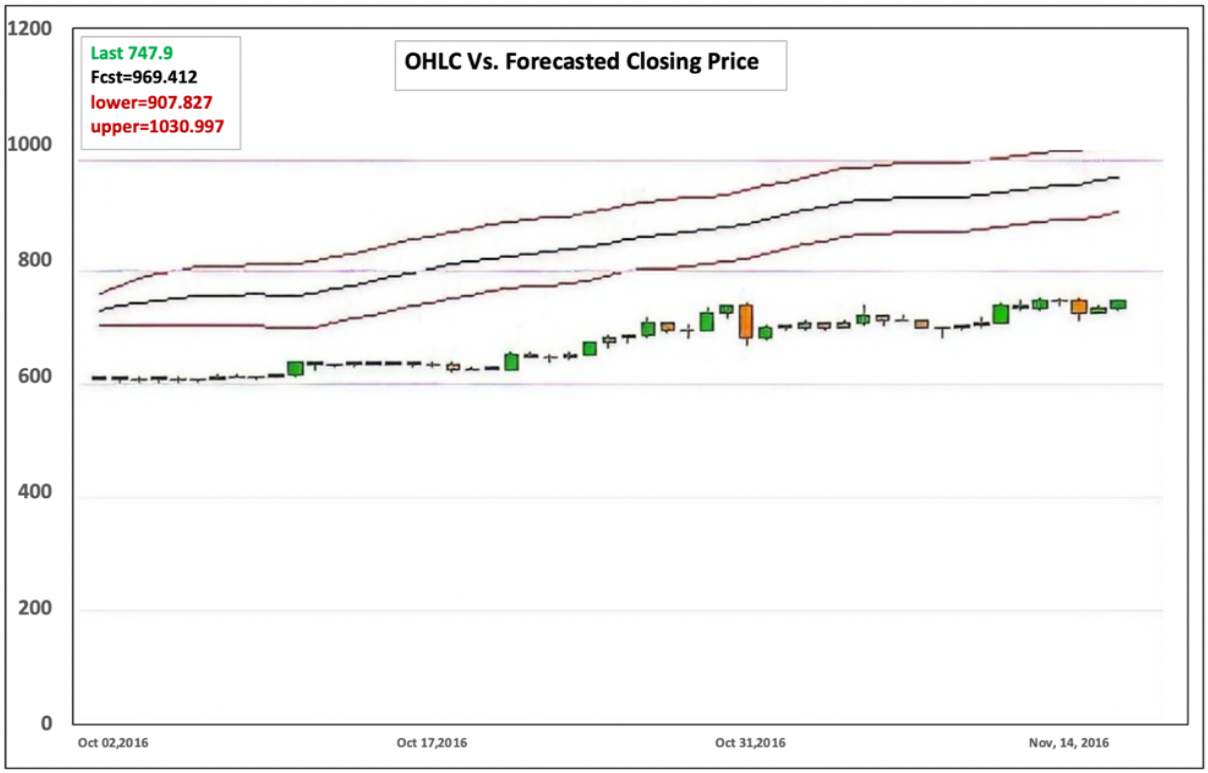


Figure 4-6: Forecasting Bitcoin closing price using the Year 2016 timeframe. Data vs. BTC OHLC

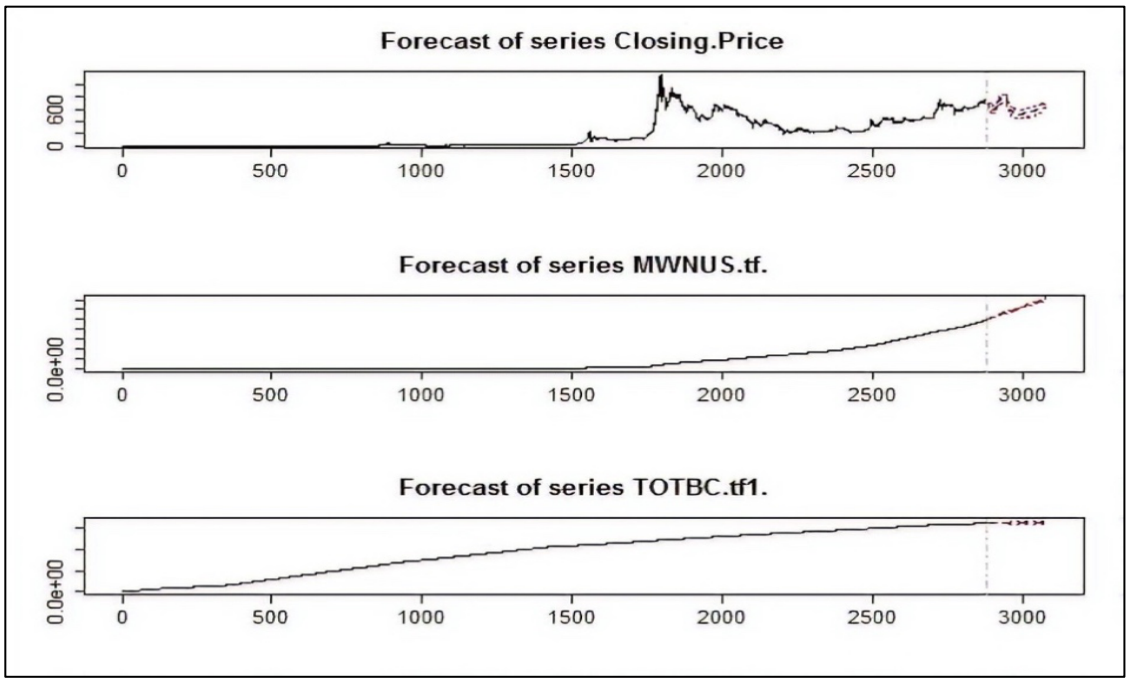


Figure 4-7: Forecasting the endogenous variables using Full timeframe data (VAR)

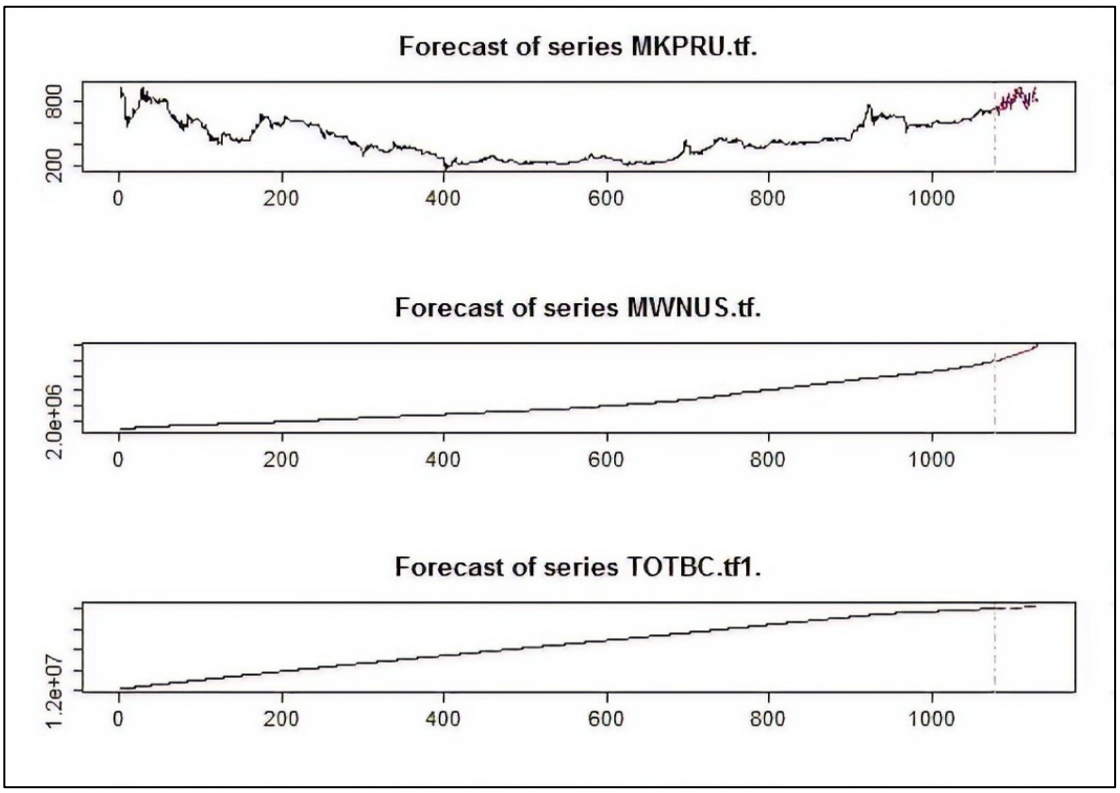


Figure 4-8: Forecasting the endogenous variables using Post-boom timeframe data (VAR)

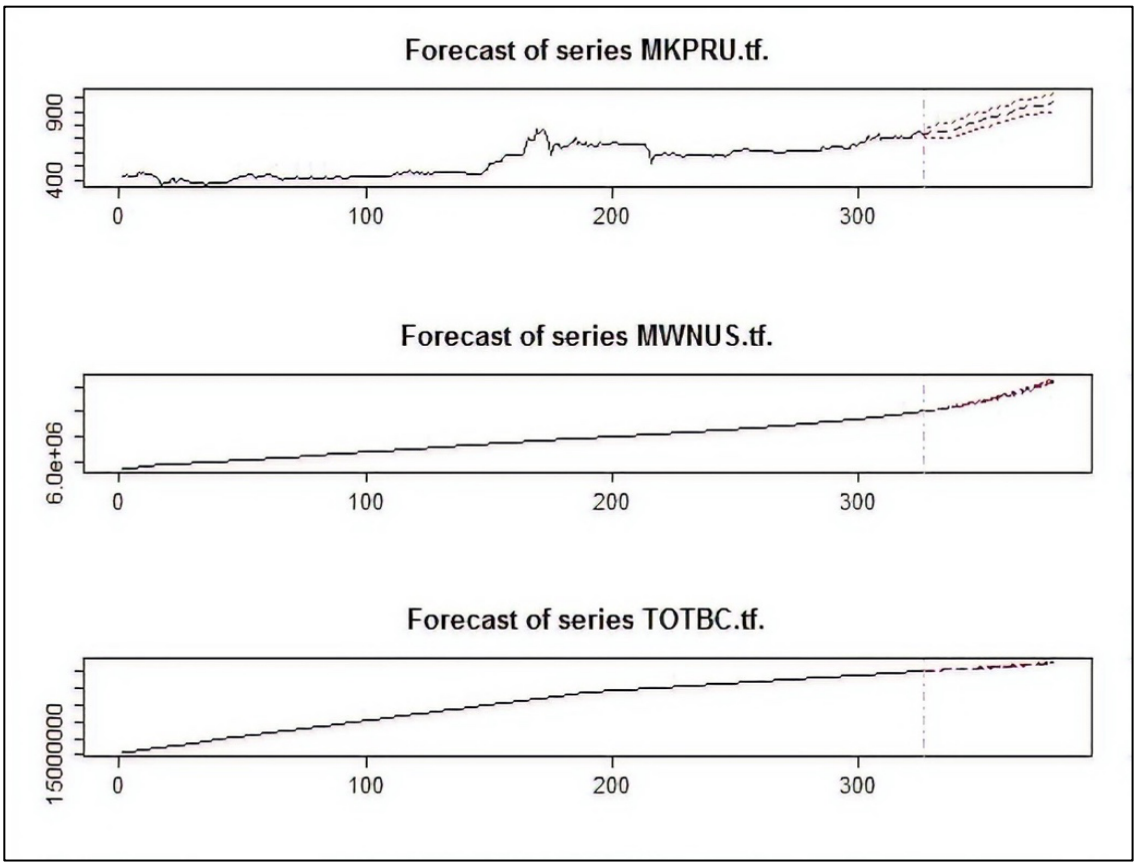


Figure 4-9: Forecasting the endogenous variables using the Year 2016 timeframe data (VAR)

B. Results of the VAR Model: Experiment B

In this experiment, we evaluated the performance of the VAR model using the period [January 2011–August 2020] Full timeframe data, Post-boom timeframe data [January 2017–August 2020], and the Year of 2020 timeframe data [January 2020–August 2020]. We can observe that the VAR model could effectively predict the prices of the BTC using the three timeframes for the variables MKPRU, MWNUS, and TOTBC, as shown in Figures 4-10 to 4-13, with the best performance obtained for the Full timeframe period.

C. Results of the BVAR Model: Experiment A

The forecasting results of Bitcoin price in USD for Full, Post-boom, and the Year of 2016 timeframes are shown in Figures 4-13 to 4-15, respectively. The red lines in each plot are from the BTC Market Price dataset (MKPRU) of Quandl. The mean absolute percentage error (MAPE) of each forecasting result was calculated to evaluate the model performance.

The forecasting of the Year 2016 and Post-boom timeframes gave good performances, as the result of the Year 2016 timeframe has a MAPE value of 2.38%, and the MAPE value of the Post-boom timeframe result is 2.85%. However, forecasting price using the Full timeframe resulted in the largest MAPE value, 19.88%. The BVAR model provided high forecasting accuracy with fewer data available or shorter timeframe in the period of [January 2009–November 2016].

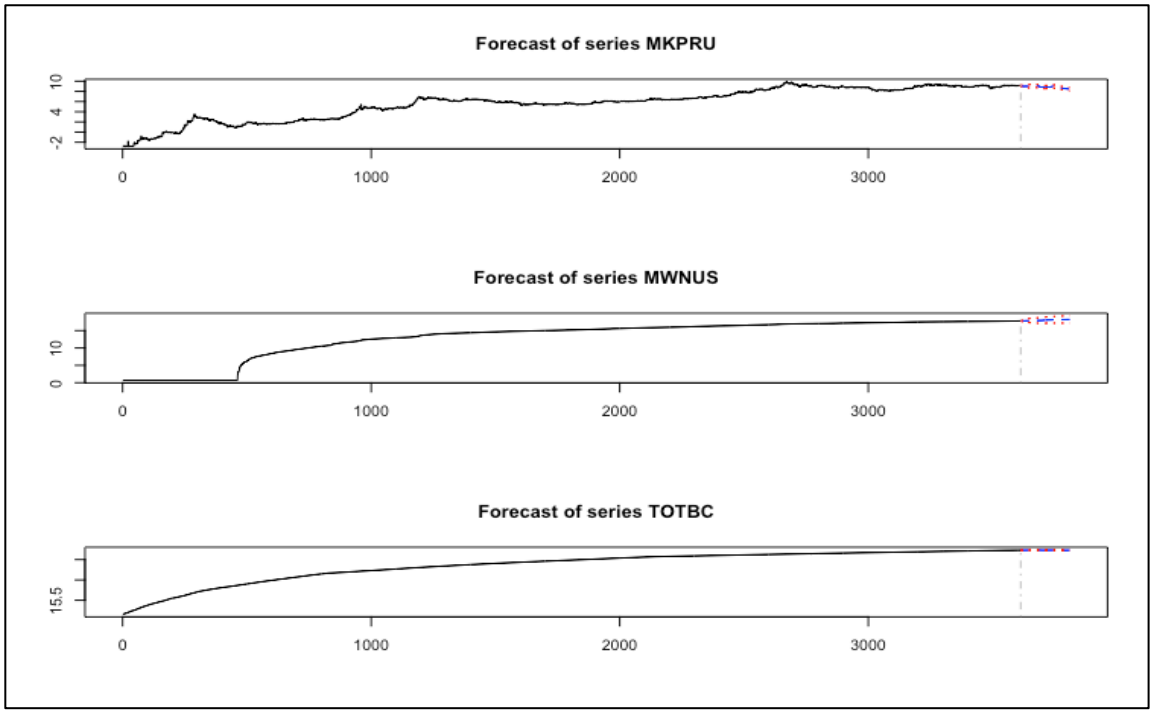


Figure 4-10: Forecasting the endogenous variables using Full timeframe data (VAR)

D. Results of the BVAR Model: Experiment B

In this experiment, we evaluated the performance of the VAR model using the period [January 2011–August 2020] Full timeframe data, Post-boom timeframe data [January 2017–August 2020], and the Year of 2020 timeframe data [January 2020–August 2020], as shown in Figures 4-16 to 4-18. We can observe that the BVAR model could predict the values of the two endogenous variables (MWNUS and TOTBC) effectively for the Post-boom period and the Year 2020 only, while the MKPRU variable had its best prediction for the Year 2020 alone. This experiment confirms that the BVAR model achieves better forecasting performance for short time periods.

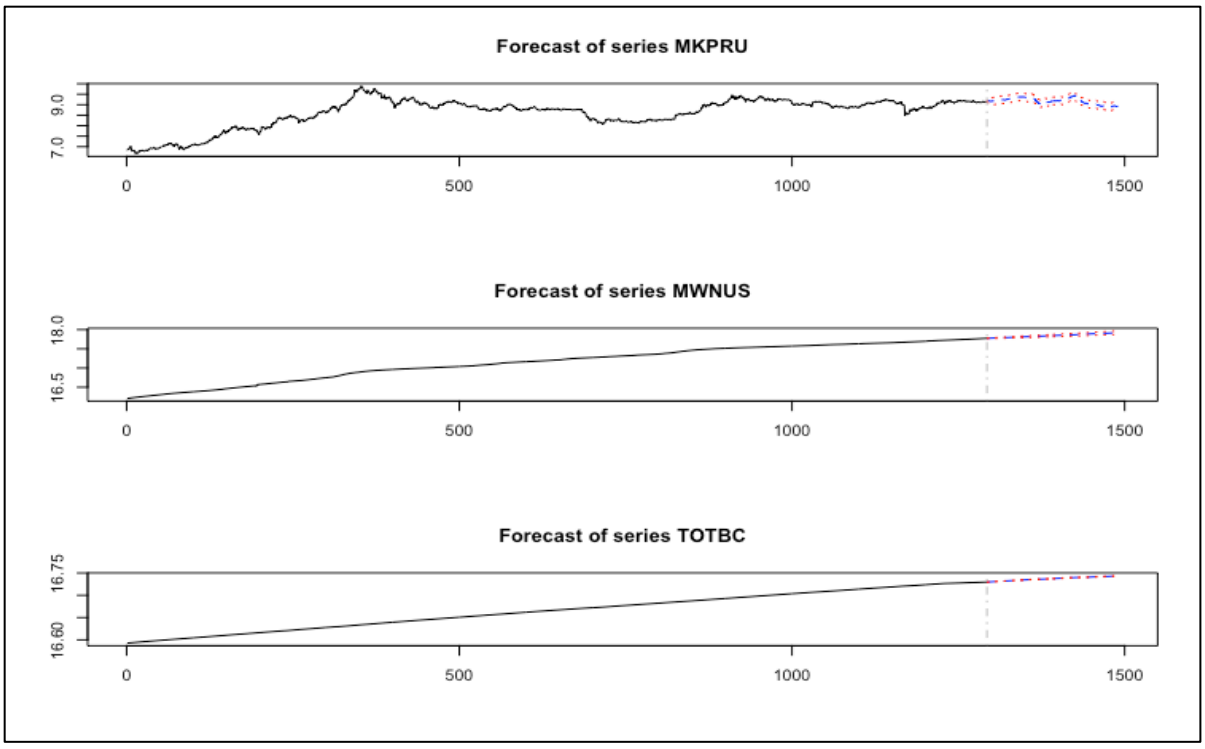


Figure 4-11: Forecasting the endogenous variables using post-boom timeframe data (VAR)

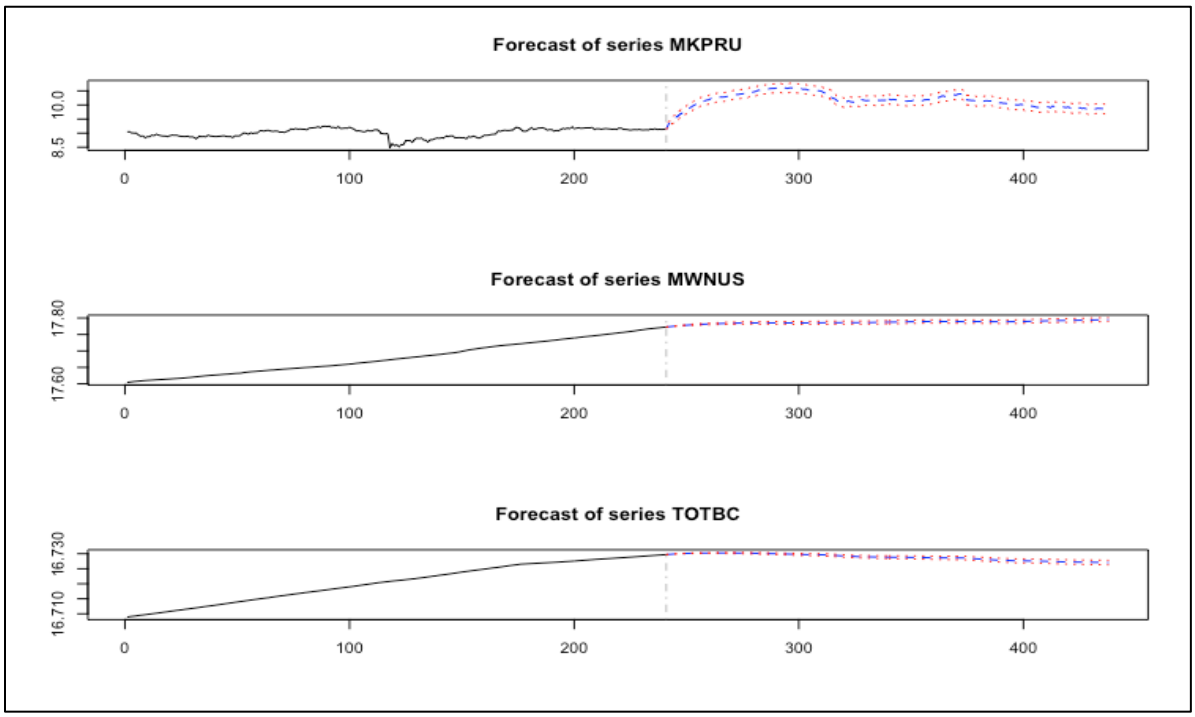


Figure 4-12: Forecasting the endogenous variables using Year of 2020 timeframe data (VAR)

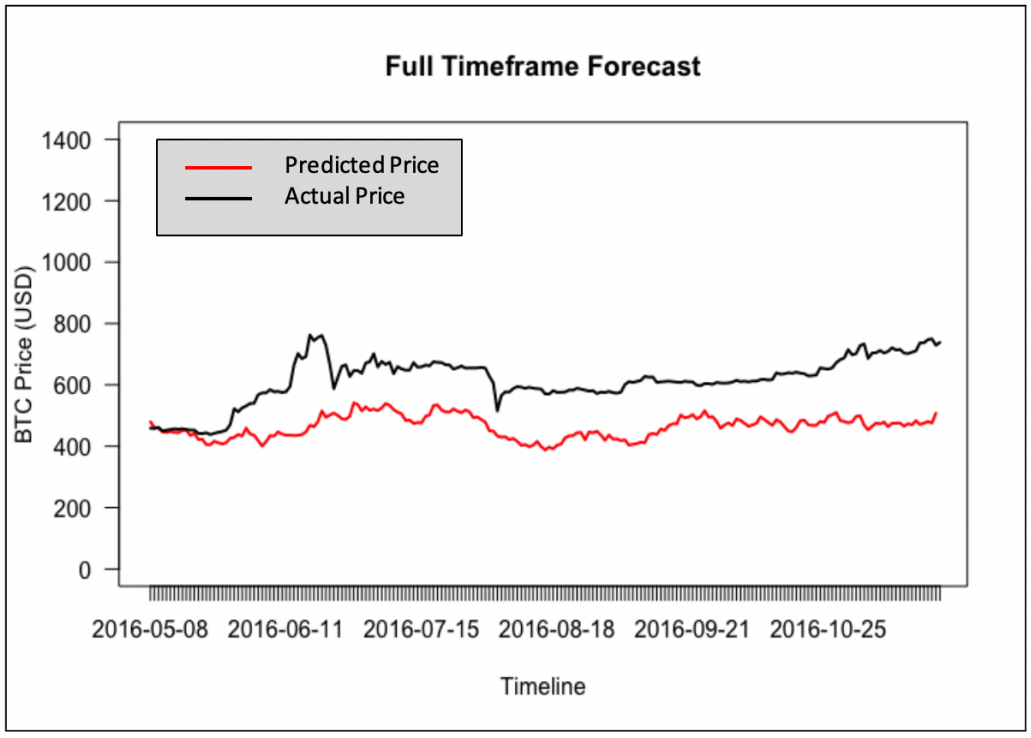


Figure 4-13: Forecasting Bitcoin closing price using Full timeframe data (BVAR)

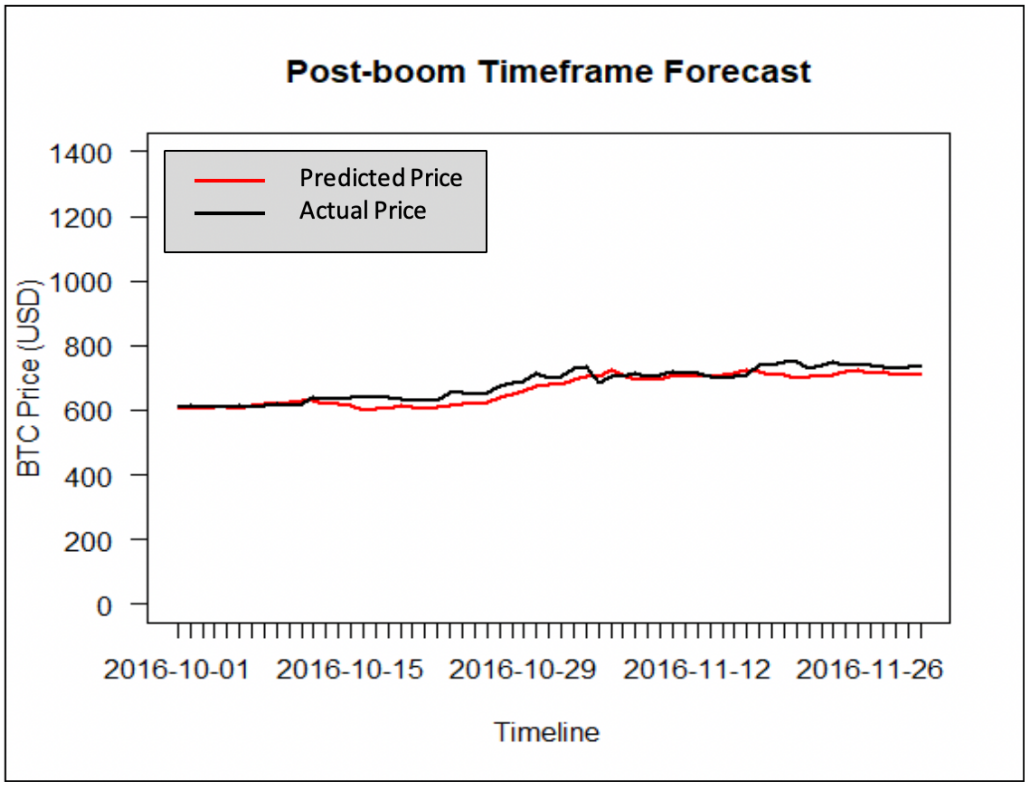


Figure 4-14: Forecasting Bitcoin closing price using post-boom timeframe data (BVAR)

E. Analysis and Discussion of Results

For the VAR model, the price of BTC was affected by the short-term lag itself as well as the number of MyWallet users. Surprisingly, it was not affected by the supply of BTC available on the market. One explanation for this could be that the supply of BTC is limited, and as such, this value is known by speculators beforehand as a market symmetric variable. The current BTC price was positively affected by 1, 2, 4, 5, 9, 11, 17, and 20-day lags of itself. It was negatively impacted by 7, 8, 10, 12, 16, and 18-day lags of itself, as shown in Table 4-1.

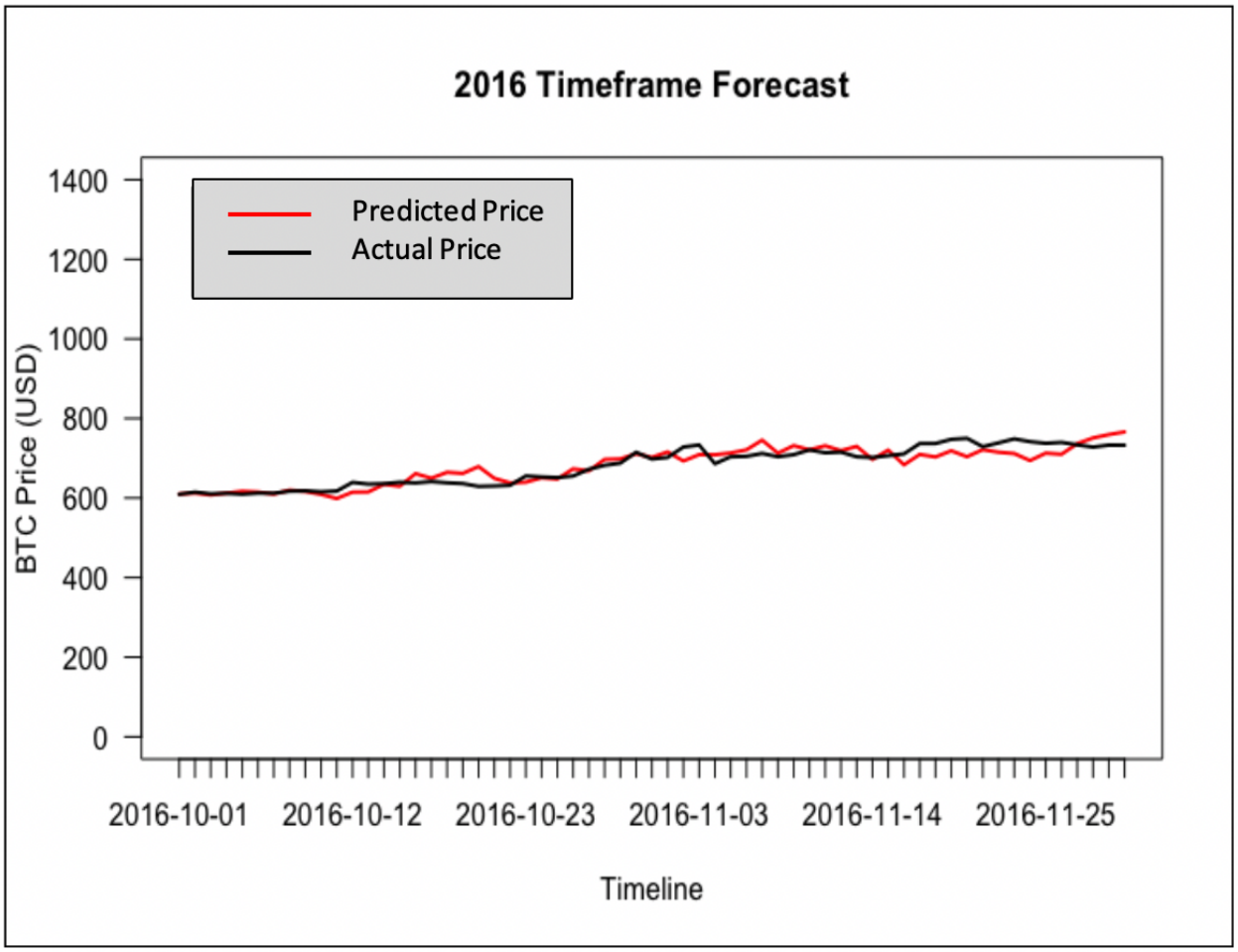


Figure 4-15: Forecasting Bitcoin closing price using the Year of 2016 timeframe data (BVAR)

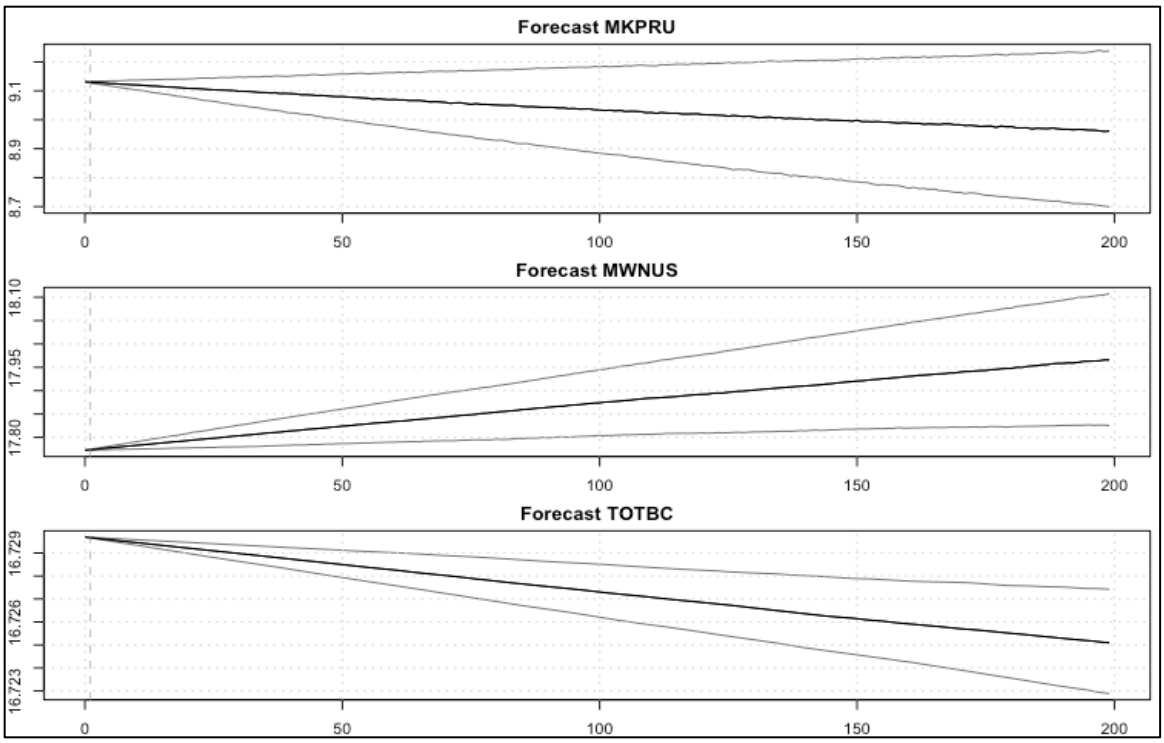


Figure 4-16: Forecasting the endogenous variables using Full timeframe data (BVAR)

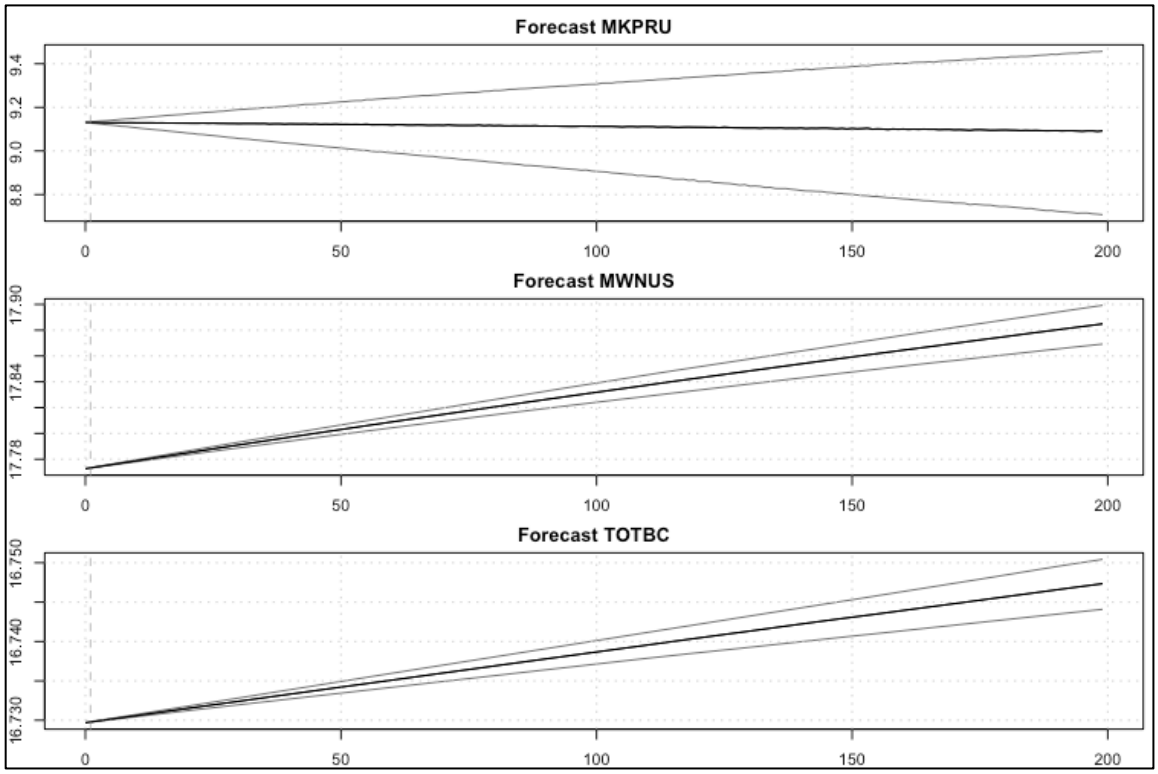


Figure 4-17: Forecasting Bitcoin closing price using the Year of 2016 timeframe data (BVAR)

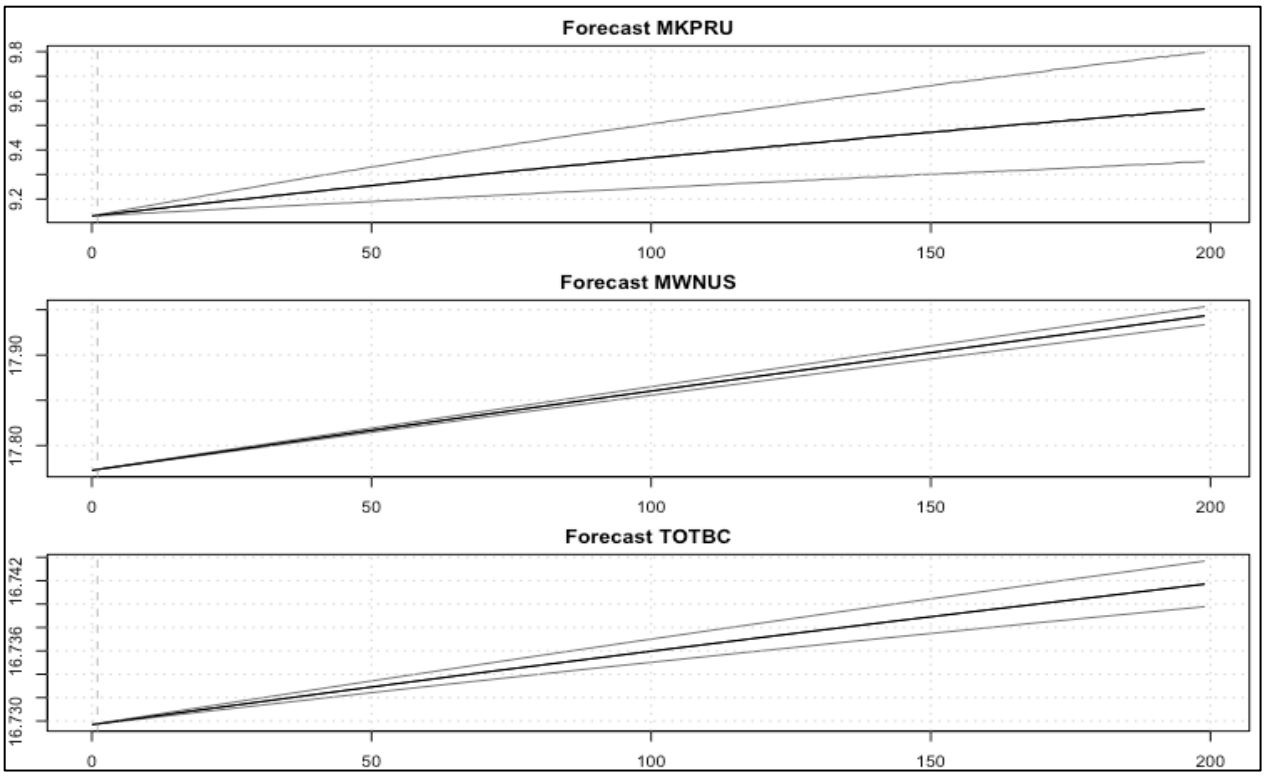


Figure 4-18: Forecasting the endogenous variables using Year of 2020 timeframe data (BVAR)

Table 4-1: Variables of significance and their effect.

Variables of Significance	Effect
1, 2, 4, 5, 9, 11, 17, 20-day lag of BTC	+
7, 8, 10, 12, 16, 18, day lag of BTC	–
1, 4, 6, 10-day lag of MyWallet users	+
2, 5, 12-day lag of MyWallet users	–
Miner's Revenue, BTC Difficulty, Change in the Number of unique addresses	+
Number of Transactions per Block, Hash Rate	–

The effects of MyWallet users on BTC price were slightly positive overall. In terms of exogenous variables, the Miner's Revenue (+), Number of Transactions per Block (–), BTC Difficulty (+), the Change in the Number of unique addresses used (+), and Hash Rate (–) all played a significant part in estimating BTC. The R2 of the model was above 99%, with F-Stats significant at a 99% confidence level, as shown in Table 4-2.

In addition to analyzing the individual factors that influence Bitcoin prices, the VAR model

predicted a great pattern of fluctuating prices. Compared with the forecasting price curves from the VAR model, the BVAR model gave a more accurate prediction of Bitcoin price to the actual values in general. Additionally, the availability and completeness of the input data played a significant role in the performance of the VAR model, while the BVAR model achieved a great forecasting result with a low percentage error rate while using only data from the years 2016 and 2020. The results demonstrate that the BVAR model performed well for a fairly limited number of observations.

Table 4-2: R2 and F-statistics

Variable	R^2	F-Statistics
BTC Price	99+%	99+%
MyWallet User	99+%	99+%
Total BTC	99+%	99+%

4.3.6 Comparative Analysis

In this section, we compare the performance of the VAR and BVAR models with some of the well-known autoregression and Bayesian regression algorithms, including the autoregression integrated moving average (ARIMA) (Chu et al. 2017; Hencic and Gouriéroux 2015) and Bayesian regression (BR) (Shah and Zhang 2014). ARIMA is a commonly used model to predict the price, and the model is a combination of three basic time-series models: autoregressive, moving average, and autoregressive moving average. Bayesian regression uses statistical analysis within the context of Bayesian inference rules. The comparison was made based on the values of the root mean squared Error (RMSE), the mean absolute error (MAE), and the mean absolute percentage error (MAPE) (Tan and Kashef 2019; Tobin and Kashef 2020). In this section, we focus on the data timeframe from Experiment B [January 2011–August 2020] and the variable of interest MKPRU (the equilibrium closing price of the BTC market as denominated by the US dollar). As shown in Tables 4-3 to 4-5, for the Full timeframe, the VAR model had the best performance. For the Post-boom timeframe, both the VAR and the BVAR models had the lowest RMSE, MAPE, and MAE values. Finally, for the Year 2020, the VAR and the BVAR models had

better performance than the ARIMA and BR models.

Table 4-3: Accuracy of forecasting models: Full Timeframe

	MAPE	**RMSE**	**MAE**
VAR	0.0249	0.3102	0.2260
ARIMA (2,2,1)	0.0421	0.3900	0.3258
BR	0.0362	0.3554	0.3826
BVAR	0.0286	0.3375	0.2501

Table 4-4: Accuracy of forecasting models: Post-boom timeframe

	MAPE	RMSE	**MAE**
VAR	0.0248	0.2708	0.2212
ARIMA (2,2,1)	0.0421	0.3900	0.3258
BR	0.0351	0.3693	0.2776
BVAR	0.0264	0.2806	0.2286

Table 4-5: Accuracy of forecasting models: Year of 2020 timeframe

	RMSE	**MAE**	**MAPE**
VAR	0.0123	0.1235	0.1023
ARIMA (2,2,1)	0.0143	0.1908	0.1262
BR	0.0129	0.1418	0.1158
BVAR	0.0130	0.1273	0.1247

4.3.7 Conclusions and Future Directions

In this paper, two VAR models were developed to analyze and understand the mechanics of the BTC market. The developed models were tested to predict the endogenous variables using selected features of exogenous variables. The two models were compared with the state-of-the-art forecasting models in order to show their efficiency. This research presents a powerful way to predict Bitcoin market price and an interesting look at what factors of this BTC network can shape new innovations in blockchain and the future of digital currency. As a new currency that is not administered by the government, there are many interesting behaviors that can be studied. From the perspective of miners, investors, or users of BTC, these findings may be useful for understanding the movements

of the price of the BTC and could help to understand what influence each of the exogenous factors has on the price of BTC. Future experiments for BTC prices will use a non-linear or dynamic VAR, which is suitable for BTC simulation. Dynamic VAR accounts for the change in a relationship by allowing the coefficients to change over time, which makes it much more challenging to analyze. The technical indicator could be extended as an exponential moving average or volume-weighted average price. Different priors can be suggested for future directions, such as the independent normal-Wishart. Additionally, analyzing the daily market returns in order to understand the distribution of daily behavior could provide insight into the classification of upward and downward trends. Incorporating the classification would enable research to understand price action in more depth with increasingly sophisticated machine-learning or nonlinear models. Finally, further investigation of machine-learning prediction models is recommended.

4.3.8 References

The references for this article are detailed in Appendix B.

4.4 The Impact of the Article

This article is published in the "Journal of Risk and Financial Management '(JRFM) journal by MDPI. On Google Scholar, in 2023, this article received around 27 citations. In ResearchGate, the article has 854 reads and 21 citations.

4.5 Key Findings of the Article

In Article 2, we applied direct forecasting using VAR and BVAR models to simulate the BTC market to understand the behavior of market participants as well as their most and least favorable market conditions according to the closing price of BTC based on an optimal set of exogenous variables. The simulated BTC market includes forecasting the endogenous variables, such as the equilibrium closing price of the market for BTC as denominated by the US dollar (MKPRU), the number of unique MyWallet users (MWNUS), and the total BTC available in the market to date (TOTBC). The performance of the VAR and BVAR forecasting models depends on the optimal selection of the set of endogenous variables of interest.

In Chapter 2, the focus was on using technical indicators to simulate the BTC market, while in this chapter, we expand this by selecting variables that are primally related to the BTC's network transaction behavior. Several variables were tested as proxies to represent the price, demand, and supply of the BTC market, respectively. After trying out numerous iterations of VARs and BVARs and using sensitivity analysis with different variables, lags, and time frames, only seven feature factors are selected, including the AVBLS, DIFF, NTRBL, MIREV, NADDU, TRVOU, and HRATE. These exogenous factors are selected because of a BTC's network transaction behavior and how the fundamental mechanics impact the closing price while minimizing the FPE.

The primary source of BTC market price data and information was the Quandl Dataset (https://www.quandl.com/data/BCHAIN). The secondary dataset was the average OHLC (open-high-low-close) candlestick values across multiple exchanges scraped from Bitcoin charts.com.

Experimental analysis over 7-year and 10-year timeframes shows the significant impact of selecting an optimal set of exogenous features to simulate the BTC market and the factor of influence. Key findings in this chapter conclude that by deploying these variables into multivariate time-series prediction models such as VAR and BVAR, it is shown that the efficiency of the VAR and BVAR models has been improved in predicting the set of endogenous variables compared to traditional autoregression and Bayesian regression models using the optimal selected set of exogenous variables for short-term forecasting. Another finding includes the impact and effect of each selected exogenous feature on the BTC price. For example, both the Miner's Revenue, BTC Difficulty, and the change in the number of unique addresses used have a positive effect on the price, while the number of transactions per block and the Hash Rate (−) have a negative impact on the BTC price. All of these selected exogenous factors played a significant part in estimating BTC.

4.6 The Contributions of The Chapter

This chapter provides a proper feature selection process to determine the optimal set of exogenous variables that drive the market movement of the BTC defined by selected endogenous variables. Both VAR and BVAR models are used in Article 2 to forecast the Bitcoin price and simulate the BTC market to understand market participants' behavior as well as the market conditions according to the closing price of BTC. From the perspective of miners, investors, or users of BTC, these findings may be useful for understanding the movements of the price of the BTC and could help to understand what influence each of the exogenous factors has on the price of BTC.

4.7 Summary of the Chapter

The chapter aims to improve the accuracy of Bitcoin (BTC) price prediction models by identifying optimal sets of endogenous (independent) and exogenous (dependent) variables that influence the BTC market. The cited article in the chapter uses Vector Auto-Regression (VAR) and Bayesian Vector Auto-Regression (BVAR) models for simulating the BTC market. The models forecast key endogenous variables like the equilibrium closing

price of BTC in USD, the number of unique MyWallet users, and the total BTC available in the market. Various factors were tested through sensitivity analysis, and ultimately, seven exogenous variables were selected, including Miner's Revenue, BTC Difficulty, and the number of unique addresses used.

The chapter demonstrates that using these selected variables significantly improves the efficiency of the VAR and BVAR models in predicting BTC prices in the short term, compared to traditional autoregression and Bayesian regression models. The research also provides insights into how each selected exogenous variable impacts the BTC price. For instance, Miner's Revenue and BTC Difficulty have a positive effect, while the number of transactions per block and the Hash Rate have a negative impact.

Published in a well-cited journal, the chapter's contributions are valuable for miners, investors, and general users of BTC as it provides a comprehensive feature selection process and insights into factors affecting BTC price movements.

Chapter 5 Predicting the Trend of Bitcoin Using Data Charts

5.1 The Objective of The Chapter

Traditional time series modeling techniques emphasize predicting cryptocurrencies using classically structured data representation as numerical features to present the time-series datasets. Rather than relying only on these numerical features to represent the time-series data, it is necessary to recognize patterns within images of time-series data charts. As we have seen in Chapter 2, neural network models have shown great performance in predicting the market movement in Bitcoin. The main objective of this chapter is to present a modified deep learning model using subtle and potentially undetectable patterns that may not be apparent using other time-series techniques using data charts as a novel representation of the time-series datasets towards enhancing the performance of the forecasting process.

5.2 Published Article 3

Ibrahim, A. F., Corrigan, L., & Kashef, R. (2020). Predicting the Demand in Bitcoin Using Data Charts: A Convolutional Neural Networks Prediction Model. In 2020 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE) (pp. 1-4). IEEE, doi: 10.1109/CCECE47787.2020.9255711.

5.3 The Article Body of Knowledge

The subsequent sections are directly excerpted from the paper titled "**Predicting the Demand in Bitcoin Using Data Charts: A Convolutional Neural Networks Prediction Model**". All credits and rights are attributed to the original authors and the source publication.

5.3.1 Introduction

In the world of stock trading, traders are looking for patterns like accessing triangles, head and shoulders, double tops, and Elliot waves. E-traders claim that these patterns can be used to predict future market movement and guide their trading strategy [1]. Deep Learning has demonstrated a significant ability to recognize subtle, undetectable patterns in various applications, including stock market prediction [2]-[8]. With the impressive ability of Convolutional Neural Networks (CNNs) to detect subtle, difficult to find, patterns in images, it is believable that the CNNs can detect those hidden patterns within images [9]-[13], especially candlestick charts [14] and use these patterns to predict future market movement. Wang and Oates [8] have applied CNNs in predicting product demand by encoding the time series into Gramian Angular Fields and Markov Transition Fields. Their model has shown very competitive results when compared to five state-of-the-art models. The idea of forecasting demand from image representations of data was motivated by [14][15], which involves the same up/down classification problem for Bitcoin. In [15], the Tensorflow and the AlexNet architecture [11][13] for the CNNs model is used. Experimental results in this work claimed to achieve over 70% accuracy on the up/down binary classification for Bitcoin prices. The work in [15] did not test the model on unseen data. It uses the validation set for accuracy, which creates cause for concern. In [10], the data creation process creates test and validation sets from random windows within the same time period, meaning the model can see possible validation images in the training set. This would give the model foresight during training, making the validation accuracy less meaningful. Inspired by [8], [10], [14], and [15], this paper aims to improve the prediction accuracy of cryptocurrencies' prices with a focus on Bitcoin using time-series data charts. In this paper, we are using a more advanced architecture, ResNet, and implementing stochastic gradient descent with restarts, and cyclical learning rate selection [10], [12]. This paper separates the test data from the training and validation data to better assess the accuracy. The proposed model has achieved an accuracy of 78.6%, which shows a significant improvement in analyzing time-series data charts instead of traditional feature-based time-series data.

The rest of the paper is organized as follows: In section 2, related work on cryptocurrency prediction models is presented. Section 3 introduces the CNNs. Section 4 discusses the proposed model using the modified architecture of the CNNs. Experimental results and validations are explained in Section 5. Finally, the conclusion and future directions are discussed in Section 6.

5.3.2 Related Work and Background

Cryptocurrencies, such as Bitcoin, Ethereum, and Litecoin, are an alternative class of digital assets that are primarily used as a medium of [12],[14],[17],[16]. Cryptocurrencies and Stock price predictions have been a heavily studied topic for decades. Traditional time series modeling has generally shown marginally positive results, at best. It can usually be concluded that the randomness of stock prices cannot be predicted using traditional machine-learning techniques. While more traditional methods like ARIMA [17][18][19][20] appear not to work when it comes to stocks, new techniques, and neural network architectures might prove to have greater predictive power than previous modeling techniques. Long Short-Term Memory (LSTM) [16], recurrent neural networks (RNN) [21] are one cutting edge architecture that has been showing a significant progress in the field of time-series predictions. With the incredible accuracy being achieved using deep learning, in particular, using CNNs, new research directions [2]-[8] have been investigated using deep neural networks that achieve incredible feats and breakthroughs, especially in complex image recognition applications [9]-[13].

5.3.3 The Proposed CNN Model using RESNET34

The CNNs operate by reacting to input, passing that reaction forward to further neurons, and training a receptive field to interpret the response and begin to make predictions. The CNNs are typically implemented in a series of alternating layers. These alternating layers are generally ordered in such a way that they have convolutional layers alternating with pooling layers. Pooling layers reduce the number of free variables at the end of the process that gets passed on to the receptive field, which is the trainable part of the network. If a

network applies pooling layers that shrink the depth of the problem in between these convolutional layers, it can be considered a local pooling layer. If the pooling happens at the end of the convolutions, it is called global. The more convolutional layers to the network, the “deeper” it is considered. CNNs are used to process visual data with the ability to interpret spatially linked data. Analyzing time-series datasets using the data chart is a recent effort in the last few years. In this paper, we show the importance of using a visual representation of data to provide a better prediction of those hidden patterns in Bitcoin trends using deep learning with a focus on CNN. The proposed model uses a CNN architecture known as resnet34 [11] and PyTorch [22], a dynamic numerical computation framework made by Facebook as its competitor to Google Tensor Flow. The process for model development involves pre-trained neuron weights calculations based on the winning ImageNet submission, determining of an optimal learning rate, training the last few layers of the network to get a base set of weights, and training the entire network until overfitting started to occur. The learning method optimized the log loss using stochastic gradient descent (SGD) with restarts [10] and a cosine annealing function.

A. Precomputed Weights

Not using data augmentation only gave the extra benefits of precomputing and saving layer outputs for each image, which helps improve future training steps. Each image in the test and validation sets were run through the network using the default set of weights associate with resnet34. Outputs coming out of the second last layer were saved as the precomputed inputs to the final layer.

B. Choose Learning Rate

Choosing the learning rate should be low enough to ensure convergence. However, if it is too low, there is a risk that the gradient optimization might get stuck in a local minimum. Finding the learning rate works by iteratively decreasing the learning rate until performance starts to degrade. The learning rate was then selected to be a magnitude of 10 larger than the optimal learning rate. In this case, the optimal learning rate was 10^{-5}, so the learning rate was set to 10^{-4}. This choice has been made to benefit the SGD[10] with restarts algorithm and help reduce overfitting during earlier training cycles. Cosine

annealing decreases the learning rate iteratively between mini batches. Between cycles, the learning rate is reset, such that the model escape from narrow valleys in the multi-parameter optimization space as shown in Fig.5-1. Wider yet shallower valleys lead to better generalization.

C. Partially Trained network

The network was initialized with the weights from the resnet34 architecture, trained on the ImageNet dataset. The training was focused on only the final layer for the first four cycles to leverage the pre-learned features present in the beginning layers. This helped the network train faster overall. This benefited from the pre-computed set of weights where the saved outputs from the second last layer were fed into the final layer for training, avoiding the need to rerun the entire network on each image.

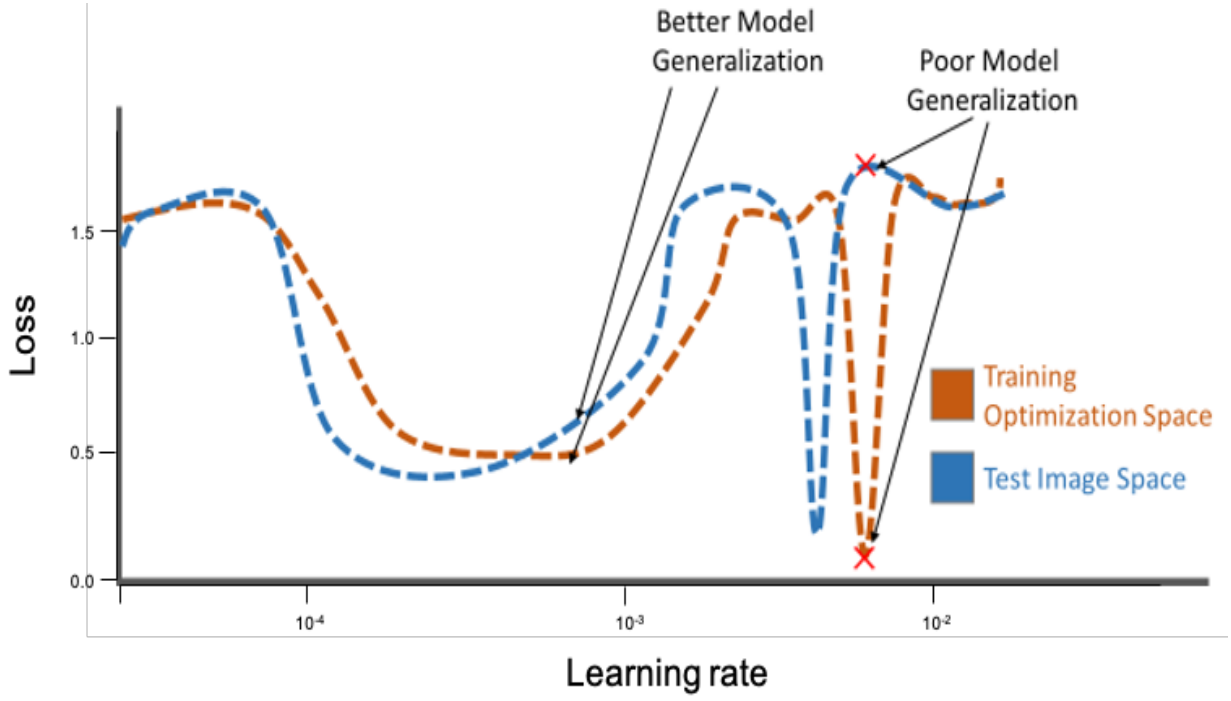


Figure 5-1: Wide Valleys Lead to Better Model Generalization

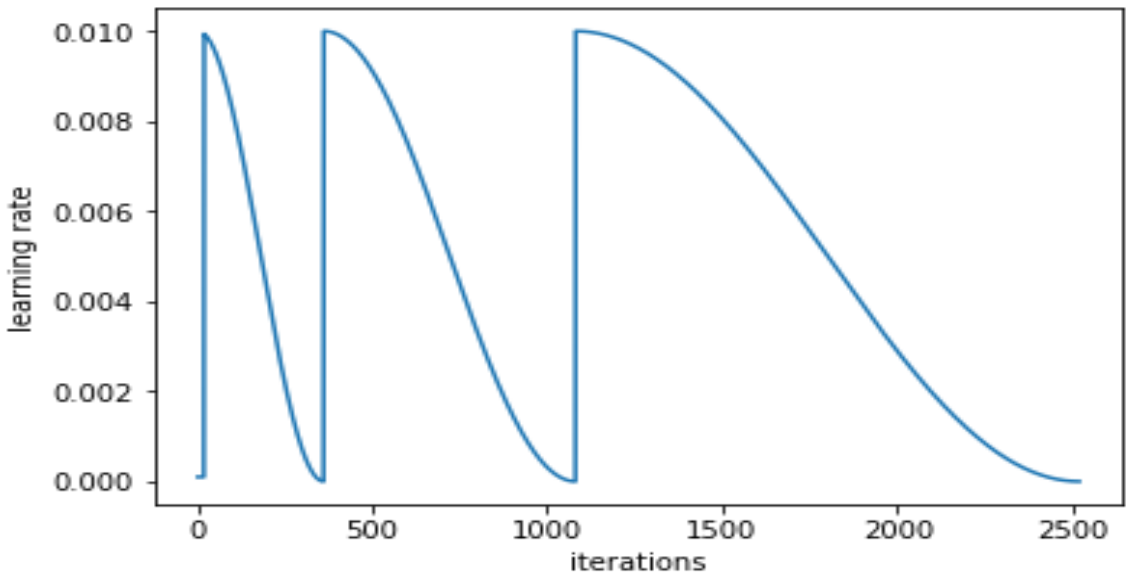


Figure 5-2: Learning Rate for Full Training

D. Fully Trained Network

Once the final layer had been trained, the previous layers in the network were unlocked, and the full network was trained using increasing learning rates. The first third of the network layers used $\frac{10^{-4}}{9}$ as the learning rate. The following two-thirds used $\frac{10^{-4}}{3}$ and 10^{-4}, respectively. Lower learning rate helps prevent the SGD algorithm from moving too far from its pre-trained weights, so the effect of "stepping out of valleys" is not as powerful and the ability of the early layers to capture generalized image features remains. The final training used a cycle multiplier so that the restarts in the SGD with restarts algorithm would occur less frequently. The learning rate is decreased following the cosine function for three cycles consisting of 1, 2, and 3 full epochs, for a total of 6 epochs of training as shown in Fig.5-2.

5.3.4 Experimental Analysis and Results

A. Datasets

The input images for the convolutional neural network are split into a training/validation and testing parts. Table 5-1 illustrates the periods selected for these parts.

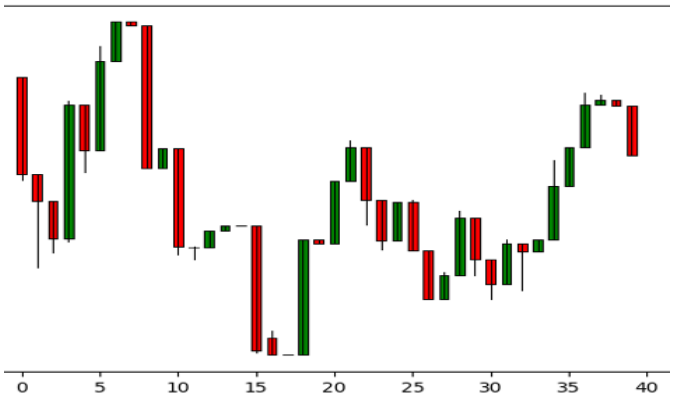
Table 5-1: Periods selection for training/testing datasets

Split	Date Range	5-minute Periods
Training and validation	January 27, 2015, to February 06, 2018	318,528
Testing	February 06, 2018, to March 23, 2018	14,400

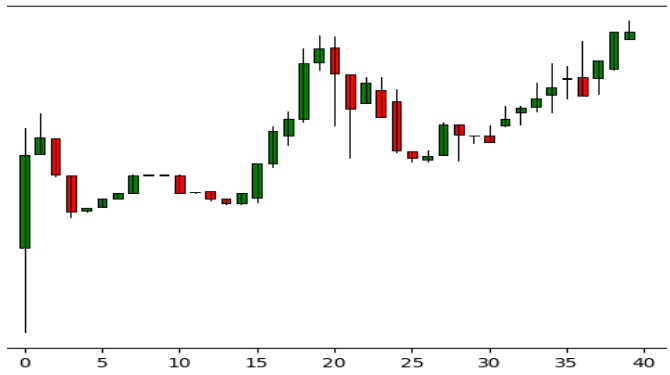
The images used for training the CNN were generated by randomly selecting a number between 1 and $318{,}528 - 41$, then taking the following 40 periods. The number of periods was selected as a simple random sampling.

Further, an open-high-low-close chart (also OHLC) [23] was created for these 40 periods. An OHLC chart is a bar chart that indications the open, high, low, and closing prices for each period. Next, the generated images were cropped to remove extra white space and get rid of the prices along the y-axis. Square images are essential for improving the speed of matrix multiplication in the GPU. This is due to an issue with the CUDA framework for Nvidia GPUs and exists for Tensor Flow 1.7. Examples of these images are illustrated in Fig.5-3.

Each image was then classified as either UP or DOWN by comparing the close prices for the $40^{th}$and 41^{st} periods. The process was repeated 500,000 times. Every 10th image created was placed into a validation set used to measure the log-loss during training. Giving 450,000 images for training and 50,000 for validation. The test images we generate using sensed data. These images were created using a sliding window covering every possible period between February 06, 2018, and March 23, 2018.



(a)



(b)

(c)

Figure 5-3: Three Candlestick Price Charts Spanning 40 5-Minute Periods

This resulted in $14{,}400 - 40 - 1 = 14{,}359$ images for the initial test set (Coinbase data). To further test the model, 50,000 images were generated from the data taken from Poloniex. The period for many of these images overlaps with the data that was used for training (Sept 2017 – December 2017). A set of 3908 images was generated for Apple, Facebook, Google, and Microsoft stocks spanning from January 1st, 2018, to March 23rd, 2018 to checks if training the model can make predictions on stock data.

B. Training the CNN

Images were fed into the network at a resolution of 480x480 to ensure network stability with higher accuracy. For most steps, the batch size was 64. However, this had to be lowered to 16 during the final stage due to memory limitations within the GPU. Training originally took 13 days in total on an intel i7 7700k server with Nvidia 1050Ti GPU (about

a 3x performance boost over and Amazon Web Services P2. xlarge instance).

C. Accuracy of the Proposed CNN Model

The proposed CNN network has shown accuracies of 75.74%, 74.74%, 77.69%, 77.94%, 77.66%, 77.56% for Coinbase, Poloniex, Apple Stock, Facebook, Google Stock, Microsoft, respectively. However, the images in [15] had been classified incorrectly by looking at the second last close price rather than most recent as illustrated in Eq5-1. and Eq5-2.

$$\text{class}_{t=41} = \begin{cases} \text{UP}, & \text{if price}_{t=41} - \text{price}_{t=39} \\ \text{DOWN}, & \text{otherwise} \end{cases} \quad (5\text{-}1)$$

$$\text{class}_{t=41} = \begin{cases} \text{UP}, & \text{if price}_{t=41} - \text{price}_{t=40} \\ \text{DOWN}, & \text{otherwise} \end{cases} \quad (5\text{-}2)$$

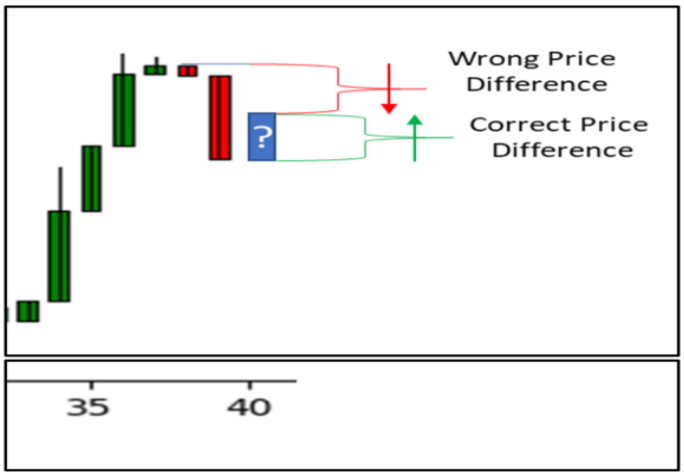


Figure 5-4: Image Classification Bug

A pull request was submitted to the original repository that solves the data generation problem (Fig.5-4). To retrain the model using the correct image classes, the same learning rate selection procedure and pre-training were followed. After training on the correct classes, the model achieved the highest accuracy with, 78.60% on the Coinbase test set.

D. Back Testing Trading Strategy

There are various factors to consider when designing an algorithmic trading bot. The cost of trading (0.1 - 0.25% for most cryptocurrency exchanges), depth of the order-book (there will only be a limited volume to trade at any given price), and speed of APIs all need to be considered when deciding if a strategy will be profitable. The predictive model needs to be transformed into a strategy that can be back-tested for historical performance. A

simple strategy involving investing testing what would happen if an investor put $1000 into Bitcoin on January 1st, 2018, as shown in Fig. 5-5.

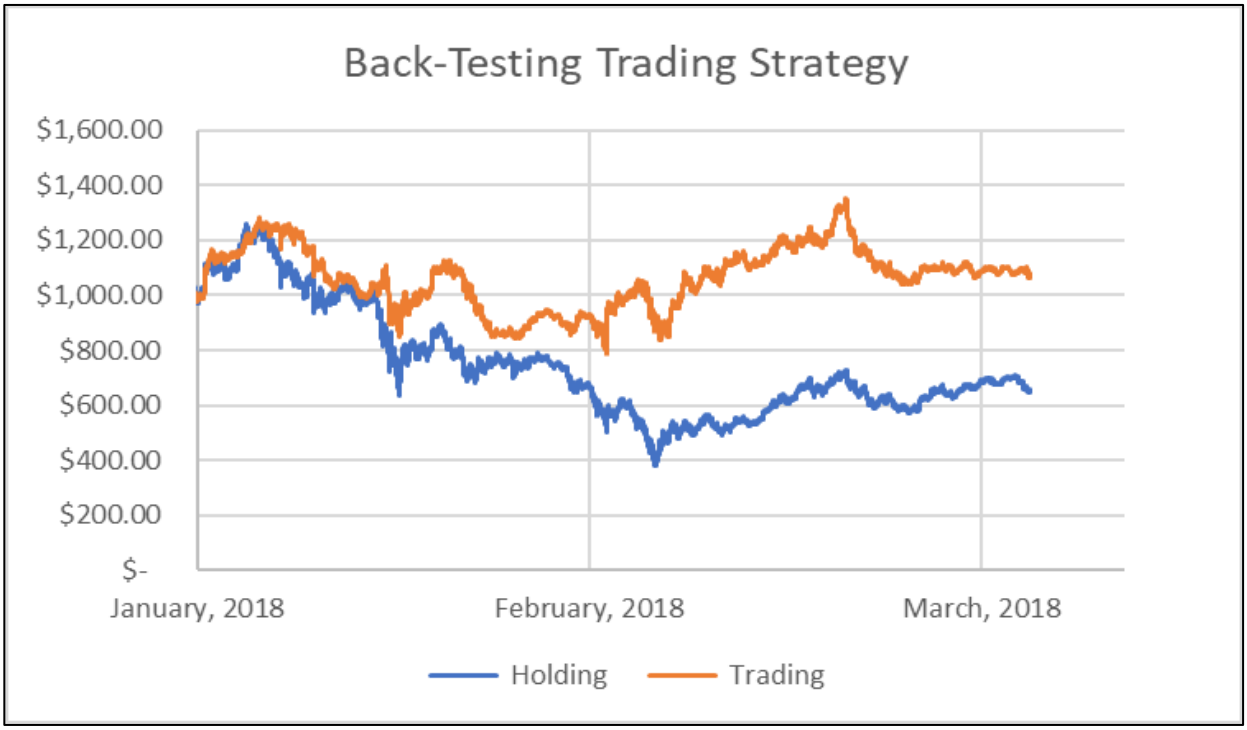


Figure 5-5: Back-Testing Strategy

The simple strategy illustrated above represents buying when the model predicts upward movement and selling when it predicts downward movement. The model-guided strategy has achieved a 6.6% return on the $1000 investment over the past 2.5 months. This might not be considered great by some greedy traders who are looking for 1000+% returns (like those experienced in 2017); however, when compared to the "buy-and-hold" strategy, it can be seen just how profitable this model might be. The past 3 months have been considered a bear market for Bitcoin [20] – [24], and a $1000 investment made on January 1st of this year would have lost 35.19%.

5.3.5 Conclusions and Future Directions

The ability to evaluate market movement for a specific cryptocurrency is critical, given the highly volatile and speculative nature of these assets. In this paper, we developed a prediction model using CNN and visual data charts to better predict the movement in

Bitcoin prices. The hypothesis is that CNNs are able to identify patterns within image data that humans cannot identify. While intuition suggests that information is lost when converting structured numerical data into images, new information may be added to the process. Converting data from a 1-dimensional stream into a 2-dimensional image might help “engineer features” that a CNN can detect visual features that a human would not have thought to create in the 1D data. An accuracy significantly above 70% demonstrates that CNNs can pick up on the patterns within the data, suggesting the validity of the data-chart approach to analyze structured data. Future directions include, moving beyond a simple binary classifier would be the next challenge for this model. Starting with estimating price movement in percentiles, it is possible to extend the CNNs classifier into a full price prediction regression model. Future extensions to the work in this paper include training the whole network on adequately tagged images.

5.3.6 References

The references for this article are detailed in Appendix B.

5.4 The Impact of the Article

This article was published in the "2020 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE)." It has received 5 citations on Google Scholar, and 54 reads and 5 citations on ResearchGate.

5.5 Key Findings of the Article

Converting time series data to an image, also known as "data charting," can be a useful way to visualize and analyze trends and patterns in the data. By converting the data to an image, you can more easily identify trends and patterns that may not be immediately apparent when looking at the data in numerical form. There are a few reasons why data charting might be a useful approach for forecasting the movement of the Bitcoin (BTC) market:

- Visualization: Data charts can provide a visual representation of the data that can be easier to understand and interpret than numerical data alone. This can be particularly useful for identifying trends and patterns in the data.
- Enhanced pattern recognition: Data charts can highlight subtle and potentially undetectable patterns in the data that may not be apparent when looking at the data in numerical form. This can be particularly useful for improving the performance of BTC forecasting models.
- Input to machine learning models: Data charts can be used as input to machine learning models, such as neural networks, which can learn to recognize patterns in the data and make predictions about future values. This can be a powerful approach for BTC forecasting, as machine learning models can be very effective at identifying complex patterns in the data.

Traditional time series models typically require that the time series be stationary, meaning that the statistical properties of the series are constant over time. This can be a limitation if the time series exhibits non-stationary behavior (e.g., trends, seasonality). On the other

hand, Neural network models can often handle non-stationary time series data more effectively. As discussed in Chapter 2, neural network models have demonstrated strong performance in predicting the market movement of Bitcoin. This chapter aims to further enhance the prediction of cryptocurrency prices, with a particular emphasis on Bitcoin, using time-series data charts.

In Article 3, An advanced neural network architecture called ResNet is proposed for the forecasting process while handling data charts. To optimize the performance of the deep learning model, we are implementing a training method called stochastic gradient descent with restarts (SGDR). SGDR is a variant of the popular optimization algorithm stochastic gradient descent (SGD) that involves repeatedly restarting the training process at regular intervals, which can help the model avoid getting stuck in local minima and improve its ability to find the global minimum of the loss function. In addition to using SGDR, a cyclical learning rate (CLR) selection is applied to the training process. CLR involves periodically changing the learning rate of the model during training, which can help the model converge faster and achieve better performance.

To train the CNN-based model, a dataset of images is generated by randomly selecting a starting point in the time series and taking the subsequent 40 periods of data. The number of periods was chosen randomly as a simple sampling method. An open-high-low-close (OHLC) chart for these 40 periods is then created, which is a type of bar chart that shows the open, high, low, and closing prices for each period. After generating the OHLC charts, the images are cropped to remove any extra white space and eliminate the prices along the y-axis. Experimental results have shown that this model is able to achieve an accuracy of 78.6% when applied to the Coinbase test data (https://www.coinbase.com/), which represents a significant improvement over the traditional feature-based time-series data analysis method.

In addition to the above capability of the proposed deep learning-based model on the created BTC data chart, the model has been applied to various stock data sets, and it has demonstrated strong performance in terms of accuracy. When applied to data from Poloniex, the model obtained an accuracy of 74.74%. In addition, it showed accuracies of

77.69%, 77.94%, 77.66%, and 77.56% when applied to data from Apple Stock, Facebook, Google Stock, and Microsoft, respectively. These results suggest that the proposed model is effective at accurately classifying and predicting outcomes in both cryptocurrencies and stock data and may be a useful tool for various applications in the field of financial analysis and investment.

5.6 The Contributions of The Chapter

One of the key advantages of our proposed deep learning model discussed in this chapter is its ability to capture subtle patterns and trends in the data that may not be immediately apparent when looking at the data in numerical form. Furthermore, this chapter has demonstrated that the proposed model can handle non-stationary time series data more effectively than traditional methods, which are often limited to stationary data. This makes it well-suited for short-term and long-term forecasting tasks, where the data may exhibit complex patterns and trends over time. Overall, the results demonstrate the potential of combining data charting and the power of deep learning as a powerful and innovative approach to analyzing time-series data. We believe that this approach has the potential to significantly improve the performance of time-series analysis tasks and open new possibilities for future research in this area in a wide range of applications.

5.7 The Summary of The Chapter

The chapter focuses on enhancing the prediction accuracy of Bitcoin market movements by utilizing data charts as a novel form of time-series data representation. Traditional methods typically rely on numerical features and often require the data to be stationary. In contrast, the chapter proposes a deep learning model that can identify complex and subtle patterns in non-stationary data as well.

A published article cited in the chapter uses Convolutional Neural Networks (CNNs) and presents an advanced neural network architecture called ResNet for this purpose. The training process is optimized using methods like stochastic gradient descent with restarts (SGDR) and cyclical learning rate (CLR), aiming to avoid local minima and accelerate

convergence.

The model is trained on a dataset of images generated from Open-High-Low-Close (OHLC) charts, representing 40 periods of time-series data for Bitcoin. The model achieved a prediction accuracy of 78.6% on Coinbase test data, outperforming traditional time-series methods. Moreover, the model also showed strong performance when applied to various stock data sets.

The chapter emphasizes that this deep learning approach is effective at capturing subtle patterns in both stationary and non-stationary data, making it well-suited for various financial analysis and investment applications. It suggests that this methodology could significantly improve time-series analysis and open new avenues for future research.

Chapter 6 – Predicting the Market Movement in Bitcoin Using Sentiment Analysis

6.1 The Objective of The Chapter

This chapter aims to provide a robust model that can give efficient forecasting of the BTC during market crashes while analyzing unstructured data such as social data (e.g., Twitter). This chapter has empirical evidence that the rationale for developing robust classification models aims at enhancing the forecasting process of the early market of BTC while understanding the impact of social media.

6.2 Published Article 4

Ibrahim, A. (2021, April). **Forecasting the early market movement in Bitcoin using Twitter's sentiment analysis: An ensemble-based prediction model**. In 2021 IEEE International IoT, Electronics and Mechatronics Conference (IEMTRONICS) (pp. 1-5). IEEE, doi: 10.1109/IEMTRONICS52119.2021.9422647.

6.3 The Article Body of Knowledge

The subsequent sections are directly excerpted from the paper "**Forecasting the early market movement in Bitcoin using Twitter's sentiment analysis"**. All credits and rights are attributed to the original author and the source publication.

6.3.1 Introduction

Cryptocurrencies, such as Bitcoin, Ethereum, and Litecoin, are an alternative class of digital assets primarily used as a medium of exchange [1]-[5]. Public key cryptography and blockchain technology are utilized to facilitate decentralized peer-to-peer transactions. Bitcoin, created in 2009, is widely regarded as the world's first cryptocurrency. Following Bitcoin's success, numerous other cryptocurrencies, dubbed 'altcoins' have been

developed. The rise of Bitcoin and altcoins have produced a deluge of data on social media platforms, blogs, forums, and countless other online mediums. There have been quite a few researchers trying to predict Bitcoin prices' behavior based on its emotions on social media platforms, such as Twitter, using various machine learning algorithms [6]-[8]. Researchers have been known to get some significant prediction results. However, very few focus on using ensemble modeling to achieve better prediction results.

XGBoost is an ensemble classifier that provides benefits such as no need for normalized data, scalability to larger data sets, and rule-based behavior that is easier for people to interpret. Thus, this paper aims to propose a Composite Ensemble Prediction Model (CEPM) using the notion of sentiment analysis. The CEPM framework is comprised of five stages, 1) text preprocessing, 2) Sentiment Scoring, 3) individual XGBoost classifications, 4) composite ensemble aggregation, and 5) model validation. In stage 1, various preprocessing steps are performed, including word quantization, text stemming, and stop-word removal. The second stage includes converting tweet text into a sentiment score as a representative of its emotion. Such a task is suited to VADER, a lexicon and rule-based sentiment analysis tool that can deal with the syntax usually used on social media. In the third stage, various instances of the XGBoost classifiers are used. The ensemble modeling is designed to maximize the model performance by utilizing a stacking of ensembles using a majority vote of XGBoost ensembles. Finally, the composite ensemble model is validated using accuracy, recall, precision, and F-scores quality measures. Experimental analysis of Twitter datasets collected during the era of COVID-19 shows that the CEPM model outperforms the individual models. It can be effectively used as an efficient (Bitcoin) BTC predictor to forecast the early market movement of Bitcoin even after the COVID-19 pandemic.

The rest of this paper is organized as follows: Section 2 provided a literature review. In section 3, the text preprocessing is discussed. Vader scoring is presented in section 3. Section 4 presents the adopted classifiers. In section 5, the proposed staking ensemble is introduced. Experimental results and analysis are discussed in section 6. Finally, section 7 concludes the paper and highlights future directions.

6.3.2 Literature Review

Several attempts have been made to use sentiment analysis to predict the early market movement of cryptocurrencies sentiment [9]-[17]. In [9], authors compared the causality of tweet sentiments, tweet volume, and buyers' ratio to sellers on Twitter with the price returns and daily trading volumes of cryptocurrencies. It has been speculated that sentiments expressed on Twitter could help in predicting cryptocurrency price changes. Li et al [10] have attempted to demonstrate this concept by training an Extreme Gradient Boosting Regression tree model (XGBoost) with Twitter sentiments to predict ZClassic price changes. The research in [10] provided the KryptoOracle to predict the Bitcoin price for the next minute using current and historical data from Twitter sentiments and Bitcoin closing prices. XGBoost, a regression tree model, was used because of its performance, speed, and retraining simplicity. In [13], Jain et al. attempted to predict the prices of Bitcoin and Litecoin two hours in advance based on the sentiments expressed in current tweets. They wanted to investigate if social factors could predict the prices of cryptocurrencies. So, they used a Multiple Linear Regression (MLR) model to predict a bi-hourly average price from the number of positive, neutral, and negative tweets accumulated every two hours. Authors in [14] compared the significance of different preprocessing techniques for tweets' sentiment analysis. They used four different machine learning algorithms to classify tweets, and they tested 16 different preprocessing methods. Based on their results, it was recommended to use lemmatization, replacing repeated punctuation, replacing contractions, or removing numbers. The research work in [17] attempted to characterize Twitter users who use controversial terms when mentioning COVID-19 on Twitter and trained various machine learning algorithms for classifying such users. The machine learning algorithms trained on these attributes included Logistic Regression, Random Forest, Support Vector Machine, Stochastic Gradient Descent, Multi-Layer Perceptron, and XGBoost. When trained on the baseline, demographic, and geolocation data, Random Forest had the highest AUC-ROC score out of all algorithms.

6.3.3 Text Data Preprocessing Methods

To categorize a large data set as Twitter, the data must be appropriately cleaned to save computational time and increase the data manipulation's overall accuracy. In heavily text-based datasets, stemming and stop word analysis are crucial in the proper analysis [19]-[22].

A. Text Stemming

Stemming is a pre-processing method utilized in text mining, natural language processing, and information retrieval applications. It is an effective approach to reduce grammatical and word conjunctions to essentially extract the root form or “stem” to improve searching by automatic sorting of word endings at the time of indexing and searching. Since certain words have similar semantic meanings but different word forms, stemming allows for a reduction in the number of distinct terms in a document and increases the number of retrieved documents. The decrease in overall variability of the text, thus shortening the final output processing time for an Information Retrieval System. In stemming, converting a word to its stem assumes each is semantically related, leaving separate words with different meanings. Two main errors occur with stemming: Over-stemming and under-stemming. In over-stemming, words with different stems are stemmed from the wrong root (false positive), and under-stemming is when words that should be stemmed to a specific root are not (false negative). Porter's stemming is an example of a truncating method that removes suffixes or prefixes of a word. It consists of five steps, where within each step, rules are applied until a condition is met. The suffix is removed if the condition is completed and the subsequent step is performed. The result at the end of the 5th step is the resulting stem. The rules follow the syntax: Porter Stemming usually provides a much better output compared to other stemmers, has less stemming error rates, and also the Porter Snowball stemmer framework is independent of the language being used. A drawback of using the Porter stemming algorithm is that the stems produced are not always real words, and the five steps in the algorithm make it a time-consuming process.

B. Stop Word Removal

Stop words in documents occur frequently but are effectively insignificant as they are used to join words in sentences. These words do not contribute to the context, and due to their frequency, they hinder information comprehension. Therefore, they are removed because they increase the amount of text in data, slowing down information retrieval effectiveness in text mining. Stop words include words like "and", "are", "because" etc.

6.3.4 Sentiment Analysis using Vader Scoring

To categorize tweets, the words must be assigned a positive or negative relative to cryptocurrency markets. A predefined value was assigned to the tweet's specific words to predict cryptocurrencies' probability of increasing or decreasing based on tweet sentiment. These words were cross-referenced with programming libraries containing lexicons of words that were assigned positive and negative values. The text used in early market predictors for cryptocurrencies using tweets was weighted positively or negatively based on these predetermined values.

VADER is a lexicon and rule-based sentiment analysis tool that can handle words, abbreviations, slang, emoticons, and emojis commonly found in social media [23]. It is typically much faster than machine learning algorithms as it requires no training [23],[24]. For each body of text, it produces a vector of sentiment scores with negative, neutral, positive, and compound polarities. The negative, neutral, and positive polarities are normalized to be between 0 and 1. The compound polarity can be thought of as an aggregate measure of all the other sentiments, normalized to be between -1 (negative) and 1 (positive). VADER was introduced by C.J. Hutto and Eric Gilbert [24]. They found that it performed better than most other sentiment analysis tools and even surpassed some human judges.

6.3.5 Forecasting Models

Several machine learning algorithms can be used to drill down the data to analyze how Twitter can be an early market indicator for cryptocurrency prices. Historical research indicates that the most commonly used Twitter Sentiment analysis tools include Vector

Support Machines and Naïve Bayes to categorize the data into positive or negative reflections for cryptocurrencies in the market.

A. Support Vector Machines (SVM)

SVM is a supervised machine learning algorithm that can be used for classification. In this algorithm, each data item is plotted as a point in n-dimensional space (where n is the number of features), with the value of each feature being the value of a particular coordinate. For a binary categorization problem, the classification is performed by finding the hyperplane that differentiates the two classes, effectively separating the data using an n-dimensional plane. In cryptocurrency Tweet predictors, the support vectors are positively or negatively valued words in tweets. SVM can be used to clearly and accurately predict an optimal threshold for positive or negative sentiment towards a cryptocurrency given a given tweet. In cases where there is no optimal solution utilizing a simple one-dimensional line and data points have substantial outliers, the data needs to be graphed in a higher dimension. It is possible to create an n-dimensional hyperplane by transforming the data set that utilizes the same maximum distance characteristics as a two-dimensional hyperplane. By using kernel functions, mapping the data in higher dimensions is possible. For SVM, kernel functions can be represented in 3D space. These functions take low-dimensional input space and transform it into a higher-dimensional space, therefore converting a non-separable problem to a separable problem. As the Logistic regressors do not optimize mislabeled data, we use SVM to minimize the classification error rather than solely rely on Naïve Bayes' likelihood. Therefore, the support vector machines model is chosen for the classification of mislabeled data. Using the hyperplane solution and mapping in the 3D plane, misplaced data can be encompassed in the proper classification. For applications in Twitter and cryptocurrencies, any Tweet related to cryptocurrencies is weighted with positive and negative values, and then a hyperplane is placed to separate the data points. Once an initial hyperplane or line is determined and separates the data from each other, the ideal placement is determined. Maximizing the distances between the nearest data point and the hyperplane determines the optimal solution. Once this best-separating hyperplane is found, all data points added to the data set will be classified based on their position relative to the hyperplane.

B. Naïve Bayes Classifier

The Naïve Bayes classification is a simple model to apply to text mining. Naïve Bayes is practical as its assumptions include a feature vector and dependent variable Y. The optimal classification is determined through the maximum likelihood of the given function: While sentiment can have either a positive or negative meaning, for the sake of simplicity in this paper, a simple binary classification is used for Naïve Bayes classification. Thus, for large amounts of data with a short 140-character document such as tweets, conditional-based probability can easily be used. There is little opportunity for varying thoughts in tweets about sentiments. This is based on the feature vector, words that are determined to be a positive or negative sentiment. Specifically, the frequency of these texts is collected for this specified model. In NB, targeted positive and negative words can be thought of as cues that direct each document being classified. Any words that appear multiple times with an insignificant or words that cannot be determined under any class can be removed from the documents to cleanse the data and ensure that probability calculations are more accurate. To account for negation, further manipulation of data with the addition of specific text to tag words with a negated meaning can then be counted as cues towards positive or negative sentiments accurately. The words are randomly grouped to determine the document's sentimental value, and each word's frequency is counted. Regardless of the word's position in a document, the words are placed to decrease frequency. This is based on the assumption that the word's position in the text does not affect how it is depicted in a document. The binary variable of each word is counted to determine the sentiment of the document. In this example, the tweet determines whether it is positive or negative or if the Bitcoin price increases or decreases. The maximum likelihood function is used using the prior class's probability with the likelihood that the document is given the class. Since we assume that the documents are independent of each other and do not affect the class, the maximum likelihood function becomes simpler to solve. Though the independence assumption is usually a constraint to using this model, tweets from individuals are unrelated to each other; thus, the independence assumption favorably works with the model.

6.3.6 The Proposed Composite Ensemble Prediction model (CEPM)

We built a composite of the Extreme Gradient Boosting (XGBoost) using a majority vote over multiple cross-validation iterations. This composite is used to achieve a better overall prediction accuracy than baseline classifiers and individual boosting algorithms. XGBoost is a novel machine-learning algorithm that improves the gradient-boosting decision tree (GBDT) and can be used for both classification and regression problems [18]. XGBoost is a boosting-tree approach that integrates many weak classifiers to form a robust classifier. It uses the CART, classification, and regression tree model.

The CEPM framework is comprised of five stages: 1) text preprocessing, 2) Sentiment Scoring, 3) individual XGBoost classifications, 4) composite ensemble aggregation, and 5) model validation. In the initial stages, various preprocessing steps are performed, including text stemming and stop word removal. The second stage includes converting tweet text into a sentiment score using VADER. The VADER sentiment analysis algorithm was used to assign each tweet a compound sentiment score based on how positive, negative, or neutral their words were. The final sentiment score is factored in the number of Twitter followers, likes, and retweets associated with each tweet. The closing price of Bitcoin, the final sentiment score, and the moving average of the last 100 data points were four input variables for our machine-learning models. In the third stage, various instances of the XGBoost classifiers are used. The ensemble modeling is designed to maximize the model performance by utilizing a stacking of ensembles using a majority vote of XGBoost ensembles. In this paper, a 10-fold cross-validation method was employed. The dataset was divided into ten parts, 9 of which were taken in turn as the training set, one as the test set, and the average value of the ten results was used as the evaluation value of the algorithm performance. Meanwhile, the experiment repeated the above process ten times, and ten evaluation values were obtained for each model, and their mean values and corresponding 95% confidence intervals were counted. The CEPM ensemble model is then validated using various quality measures.

6.3.7 Experimental analysis and results

A. Evaluation Metrics

It was found that a confusion matrix is the most commonly used measure to determine the quality of the methods used in predicting the real-value cryptocurrency trading strategies. The confusion matrix provides a visual performance assessment of a classification algorithm as a matrix, which is then used to determine the quality of the results given the classification problem. For example, a confusion matrix can analyze models for understanding sentiments toward Bitcoin in Tweets. Based on the words associated with the term “Bitcoin,” each tweet is assigned to a negative or positive category. Positive tweets are indicators of upward movements in the Bitcoin price. The most popular metrics used to evaluate the results presented in a confusion matrix include accuracy, precision, recall, and F-score. Each metric gives a value that can communicate whether the model is a good model or not [25]-[29].

Accuracy is computed by determining the percentage of observations that were labeled correctly. This measure has been used as evidence to support the quality of some models used to predict Bitcoin pricing. However, accuracy is not the most reliable metric since accuracy provides misleading results as the classes are not balanced, as is the Bitcoin market. Accuracy is given as a percentage. The closer this value is to 100%, the better the model's predictive ability is. Precision measures the ratio of correct positive inputs. Recall, also known as sensitivity, measures the ratio of the items present in the correctly identified input. These metrics focus on the true positives, making their results more reliable. If Precision is a higher ratio, it represents a robust predictive ability by the model. Lastly, the F-score takes the weighted average of precision and recall, taking both false positives and false negatives. This metric is beneficial in evaluating cases with uneven class distributions [30]-[34].

B. Experimental Datasets

We used the Twitter dataset from [35][36]. Preprocessing steps over time with BTC’s closing prices are computed per minute. As tweets are created much more frequently than once a minute, we aggregated all tweets’ scores into a per-minute. The CEPM ensemble

model is validated using accuracy, recall, precision, and F-scores quality measures. It can be shown from Table 6-1 that the XGBoost ensemble has the highest Precision, recall, and F-score as compared to Logistic regression, SVM, NB, and a single XGBoost. We have assessed the proposed CEPM model's performance using “accuracy” as another quality measure, as shown in Fig.6-1.

Table 6-1: Precision, Recall, F-Score (COVID-19 Tweets)

	Precision	**Recall**	**F-score**
LR	0.6743	0.4532	0.54207141
SVM	0.64843	0.5543	0.59768152
NB	0.665732	0.65421	0.65992071
(XGBoost)	0.78953	0.809532	0.7994059
CEPM	0.8926	0.883474	0.88801355

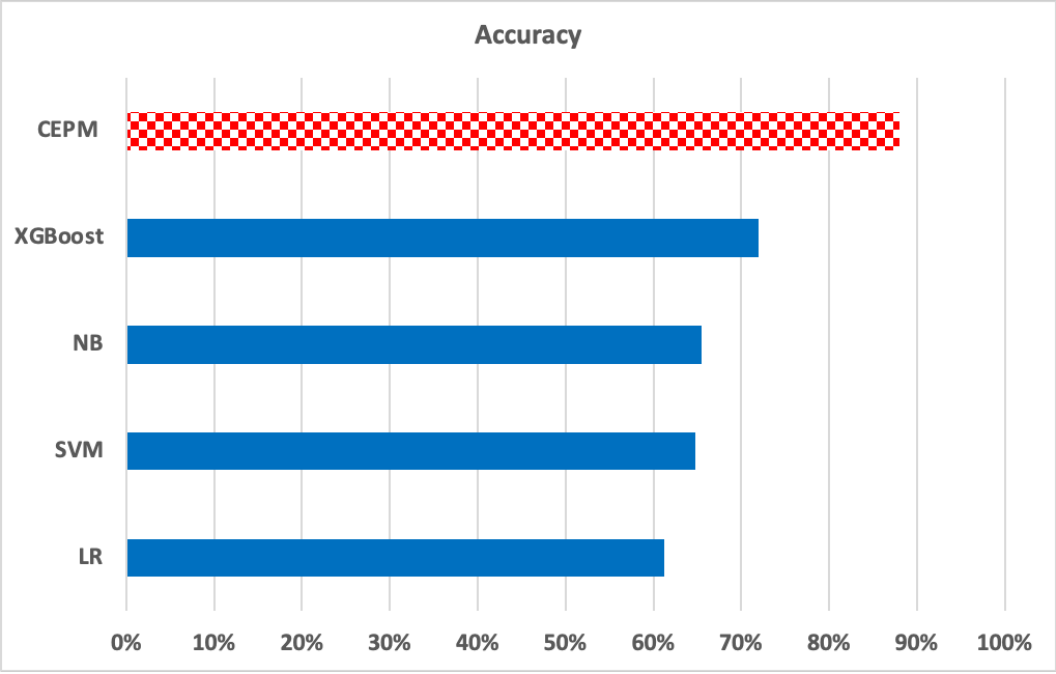


Figure 6-1: Accuracy (COVID-19 Tweets)

It can be shown from Table 6-2 that the CEPM model has achieved an improvement of up to 21%, 16%, 18%, and 22%, in the Precision, Recall, F-score, and accuracy, respectively,

as compared to the LR, SVM, NB, and XGBoost. In Fig.6-2, it can be illustrated that the proposed CEPM takes more iterations measured by the increase in the computational time as compared to other algorithms. Further adjustment to the number of iterations is considered future work to balance the trade-off between accuracy improvement and time overhead.

Table 6-2: % of improvement in Precision, Recall, F-Score, Accuracy (COVID-19 Tweets)

Precision	Recall	F-score	Accuracy
21%	16%	18%	22%

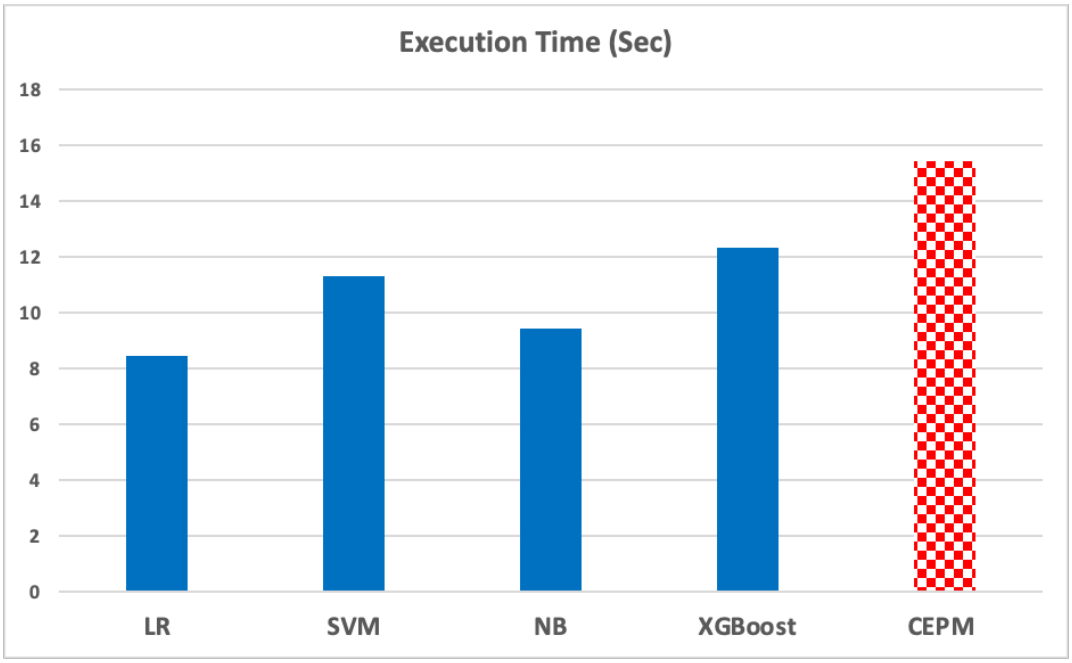


Figure 6-2: Execution Time (COVID-19 Tweets)

6.3.8 Conclusion and Future Directions

In this paper, we have developed a composite aggregate of the well-known XGBoost classifier to better predict the early BTC market movement. Experimental results show that the proposed CEPM outperforms other state-of-the-art techniques using Twitter datasets collected during the Era of COVID-19. The proposed model can be further adopted to forecast the BTC market even after the COVID-19 pandemic to assess

individuals and firms in future investments. Future research directions would include the adjustment of the number of incremental iterations of each XGBoost and the incorporation of various sentiment scoring schemes compared to VADER.

6.3.9 References

The references for this article are detailed in Appendix B.

6.4 The Impact of the Article

This article is published in the "IEEE International IoT, Electronics and Mechatronics Conference (IEMTRONICS) ."On Google Scholar, this article has received 10 citations. In ResearchGate, the article has 82 reads and 9 citations.

6.5 Unleashing Social Media Influence on Bitcoin Forecasting

Posts on social media platforms such as Twitter, Facebook, and Reddit can influence the perceptions and expectations of traders and investors and potentially impact the price of Bitcoin. These sources of data can provide valuable insights into the sentiment, or attitude, of the public towards BTC, as they reflect the collective opinions and emotions of the individuals who produce them. For example, suppose a large number of social media users express positive sentiments about Bitcoin and its future prospects. In that case, this may lead to increased demand for the cryptocurrency and a corresponding price increase. Conversely, social media users expressing negative sentiments or concerns about the market may lead to decreased demand and price decline. In Chapters 2,3 and 4, the primary focus was on handling structured data (numerical features or data charts). In this chapter, the power of social media on impacting the BTC market is broken down into two main processes: a data modeling (i.e., text data) process and a forecasting process. This chapter focused on providing a framework for forecasting the early market movement of BTC using unstructured data. While traditional classification models perform well in short-term forecasting, there is a demand for a robust model that can provide similar or even better results while addressing market-crashing periods such as COVID-19.

6.6 The Methodology Used

XGBoost is a type of ensemble classifier that has several advantages. It does not require data to be normalized, it can handle large datasets, and its decision-making process is based on rules that are easy for humans to understand. Article 4 proposes a Composite Ensemble Prediction Model (CEPM) that utilizes sentiment analysis to make predictions. The CEPM framework consists of five stages: 1) text preprocessing, 2) sentiment scoring,

3) individual XGBoost classifications, 4) composite ensemble aggregation, and 5) model validation. In the first stage, various preprocessing steps are applied to the text, including word quantization, stemming, and stop word removal. The second stage involves converting tweet text into a sentiment score, which represents the emotion conveyed by the text. VADER, a lexicon and rule-based sentiment analysis tool, is used to perform this task, as it is well-suited for dealing with the syntax commonly used on social media. In the third stage, multiple instances of the XGBoost classifier are utilized. The ensemble modeling is designed to improve the model's performance by stacking ensembles and using a majority vote of the XGBoost ensembles. Finally, the composite ensemble model is validated using accuracy, recall, precision, and F-scores quality measures.

6.7 Key Findings of the Article

Experimental analysis of Twitter datasets collected during the COVID-19 pandemic shows that the CEPM model outperforms individual models. It can be effectively used to forecast the early market movement of Bitcoin, even after the pandemic. The CEPM model has improved up to 21%, 16%, 18%, and 22% in Precision, Recall, F-score, and accuracy, respectively, as compared to the Linear Regression, Support Vector Machines, Naive Bayesian, and XGBoost.

6.8 The Contributions of The Chapter

By analyzing social data, it may be possible to identify patterns or trends that can be used to make predictions about the future value of Bitcoin. This is because social data can provide insight into how people think and behave, which can be a useful indicator of market trends. To overcome the limitations of existing classifiers while handling various types of datasets with different configurations and characteristics, such as time-series datasets, in this chapter, a novel ensemble-based classifier is proposed to achieve better forecasting results compared to an individual classifier, especially for unstructured data such as social data.

6.9 The Summary of the Chapter

There have been multiple efforts to utilize sentiment analysis to forecast the initial market behavior of cryptocurrencies through the analysis of tweet sentiment. As discussed in Article 4, various machine learning models have been used to utilize the sentiment of social data to provide a short-term forecasting model for the BTC. However, none of these models have been investigated in market crash periods. In addition, each classifier works on its domain space with its architecture and processes. Researchers have been known to get some significant prediction results. However, few focus on ensemble modeling to achieve better prediction results. This chapter provides a novel ensemble-based model that practices users' sentiment on the Twitter collected dataset during a market crash period (i.e., COVID-19) to provide an efficient early indicator of market movement.

Chapter 7 – Analyzing Bitcoin Trends Using Sentiment Consensus Clustering

7.1 The Objective of The Chapter

To overcome the limitations of labelled data availability and robustness during the market crash period. A generalization framework and algorithm are proposed in this chapter to provide an unsupervised learning process for better forecasting the BTC market during unstable market periods (i.e., market crashes).

7.2 Published Article 5

A. Ibrahim, "Analyzing BTC's Trend During COVID-19 Using A Sentiment Consensus Clustering (SCC)," 2021 IEEE 12th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), 2021, pp. 0460-0465, doi: 10.1109/IEMCON53756.2021.9623182.

7.3 The Article Body of Knowledge

The subsequent sections are directly excerpted from the paper titled "**Analyzing BTC's Trend During COVID-19 Using A Sentiment Consensus Clustering (SCC)"**. All credits and rights are attributed to the original author and the source publication.

7.3.1 Introduction

Bitcoin (BTC) is a digital asset developed in 2009 and is mainly adopted as a medium of exchange. The rapid growth and trade of Bitcoin have resulted in a tremendous amount of social media data, such as tweets. [1]-[5]. Clustering Analysis had a significant impact on segmenting BTC movements into good, neutral, and negative mood states. However, because each clustering method operates in its own domain space, there is no best clustering technique for text data like tweets. [6]-[8]. Identifying the true clusters, clustering scalability, vulnerability to noise, dealing with distributed data, and manipulating

data of varied configurations are all challenges in data clustering. We use the concept of consensus clustering in this research to cluster the attitudes collected by lexical sentiment analysis successfully. The recommended attitude The SCC framework has five key steps: 1) preprocessing, 2) sentiment, 3) clustering, 4) aggregation, and 5) validation. We start by stemming the text and removing the unnecessary words. After that, VADER scoring is used to turn the text into a sentiment score. Various clustering algorithms are then used in the next stage. By employing an agreement technique, the Consensus clustering model is designed to maximize segmentation performance. Finally, the constructed consensus model is validated using a variety of quality indicators, including the separation index (SI), Sum of Squared Errors, and Mean Squared Errors [9]. The Consensus model outperforms the various clustering algorithms in experiments using COVID-19 Twitter datasets. Compared to the KM, BM, and PAM, the SCC has improved by up to 24%, 26%, and 21% in the SI, MSE, and SSE, respectively. Using the SCC has a computational overhead of only 5%. In addition, by utilizing actual BTC prices for trend prediction, the SCC has achieved a remarkable accuracy of up to 36%. The SCC can be further adopted as an outstanding predictor to forecast the behavior of the Bitcoin market post-COVID-19 pandemic.

The remaining of the paper is: the second section included a review of the literature. Text preparation is described in section 3. In section 4, Vader's scoring is shown. Clustering approaches are presented in Section 5. The proposed Consensus clustering approach is introduced in section 6. In section 6, the findings and analysis of the experiments are discussed. Section 7 brings the manuscript to a close and discusses future research.

7.3.2 Literature Review

Many studies have been presented to forecast the market mechanics of cryptocurrencies using Twitter's analysis [9]-[17]. Researchers in [9] evaluated tweet volume and moods and buyers' to sellers' ratios on Twitter to cryptocurrency price returns and daily trading volumes. Li et al. [10] showed that sentiments expressed on Twitter might be used to predict price movements using Twitter sentiments; they trained an XGBoost for this purpose. The KryptoOracle estimated the Bitcoin price for the next 1 minute using historical data, Twitter sentiments, and the closing prices [10]. The authors of [13]

projected Bitcoin and Litecoin prices 2 hours ahead of time based on tweet sentiments. They utilized MLP to predict a bi-hourly average price. The importance of several preprocessing strategies for sentiment analysis of tweets was compared in [14]. They classified tweets using four machine learning algorithms and 16 preprocessing approaches. The studies in [17] and [18] aimed to categorize Twitter users who utilize problematic terms when addressing COVID-19 on Twitter and trained several machine learning algorithms to do so. LR, RF, SVM, MLP, and XGBoost were among the machine learning methods trained on these properties. Random Forest has the greatest AUC-ROC score. A data mining method is presented in [19] for detecting groups of similar Twitter messages made on a specific occasion. The authors launched the Louvain algorithm and its modified variant on Twitter datasets in [20] to boost the performance and shorten the execution time. [21] presents a comparative analysis of various clustering techniques on Twitter datasets. The research published in [22] constructed an algorithm that merged the DBSCAN with a consensus matrix on a Twitter corpus. Then, they utilized cluster analysis to uncover subjects that the tweets described. They used k-means and Non-Negative Matrix Factorization to cluster the tweets (NMF). Both algorithms produced similar findings; however, the NMF was faster and generated results that were easier to grasp.

7.3.3 Text Preprocessing

To improve general accuracy, the text data should be cleansed. The procedures of stemming word and removing stop word are critical in text analysis for comprehensively text-based datasets [22]-[23].

A. Stemming of Text

Word stemming can be defined as a technique for reducing word and grammatical conjunctions to retrieve the root shape or defined as "stem" to enhance search results. Stemming reduces the amount of different terms in a given document while increasing the total documents returned. Transforming a word to its root presupposes that all words are semantically connected, resulting in independent words with different meanings. Porter's stemming eliminates a word's suffixes or prefixes regardless of the language used. The

Porter stemming method has the disadvantage that the roots it produces are not necessarily actual or true words, and it is computationally expensive.

B. Removal of Stop Word

Stop words are common in texts; however, they are largely meaningless because they connect words in sentences. These words add nothing to the context, and their recurrence makes understanding information difficult. As a result, they've been deleted because they add to the amount of text in data, decreasing the effectiveness of the text mining process. Stop words include words like "and," "are," and others.

7.3.4 Vader scoring

VADER is a sentiment analysis method that uses a lexical and rule-based approach to address common social media terms, acronyms, slang, emoticons, and emojis [23]. It is frequently of high speed compared to traditional methods using machine learning modeling [23][24] because it does not require any training. It generates a vector of sentiment scores with negative, neutral, positive, and compound polarities for each body of text. All polarities, negative, neutral, and positive, are standardized to a number between 0 and 1. The compound polarity is the sum of all other emotions on a scale of -1 (negative) to 1 (positive) [25].

7.3.5 Clustering Approaches

Several clustering algorithms are used to partition the sentiments to analyze the early market indicators for cryptocurrency prices.

A. K-means

For many practical applications, KM is regarded as an excellent clustering approach [21]. The algorithm is an iterative procedure that requires a priori knowledge of the number of clusters k. The first partitioning is constructed at random; the centroids are assigned to random locations in the space region. The algorithm partitions the data into k disjoint partitions marked by the corresponding centroids based on an objective function criterion. Data elements are assigned to the nearest centroid. The distance criterion is the most often used objective function criterion. The cosine correlation and Euclidian distance are

often used distance (or similarity) measurements. The procedure converges when partitioning does not change. KM is appealing because of its convergence quality, as well as its simplicity. KM is susceptible to noise and cannot handle datasets of varying forms. It is skewed toward spherical shape datasets.

B. Bisecting K-means

The bisecting KM technique [23] is a variation of the KM approach. Bisecting k-means splits the data set into two groups using k-means. One of them is then chosen and further split in half. This procedure is iteratively repeated until the required number of clusters, k, has been obtained. There are several methods for deciding which cluster to break (e.g., homogeneity criteria). For example, at each phase, we can choose (1) the cluster with the largest size, (2) the cluster with the lowest homogeneity, or (3) a criterion that takes both size and homogeneity into account. Iteratively applying a divisive bisecting clustering technique to the dataset can be grouped into any number of groups. A flat or hierarchical division can be generated using the BKM approach. The clusters are arranged in a hierarchical binary taxonomy. The bisecting division approach is highly interesting in many scenarios, such as document retrieval/indexing challenges. A "refinement" is often required to re-cluster the findings because some information is left behind with no way to re-cluster it at each level [20].

C. Partitioning Around Medoids

The PAM technique [26] selects a medoid for each cluster at each cycle. Medoids are created for each group by finding an element mi within the group that minimizes the objective function, which is the sum of all cluster object distances to the cluster medoid. The advantage of PAM is that it is more forgiving of noisy data and anomalies. PAM works well with small datasets but not so well with large ones. CLARA [29] processes one or more random samples from a big data source using PAM. Ng and Han [15] propose CLARANS as an extension of PAM. For disk-resident datasets, CLARA and CLARANS are inefficient because they require many scans of the entire dataset. A successful sample solution of clustering does not always mean that the entire dataset will cluster successfully.

7.3.6 The Sentiment Consensus Clustering (SCC)

The sentiment Consensus clustering (SCC) framework consists of five key steps:

Step 1: We used the raw data without preprocessing and saw a considerable decline in performance; as a result, we performed text stemming and removal of stop words.

Step 2: The text is then converted into a sentiment score using VADER scoring. In this paper, we use the negative, neutral, and positive sentiment scores to represent each tweet in the Consensus clustering. Such that each tweet is represented by a vector v in the 3-dimensional space. The input matrix M to the Consensus clustering model is of size (nx3), where n is the total number of tweets in the collected corpus. At this stage, the Proximity Matrix, PM, using cosine similarity (ranges from 0-1), is calculated offline. The size of the PM matrix is nxn, as the PM is symmetric, we only store the upper diagonal elements such that only n(n-1)/2 elements are stored for calculations.

$$PM(i,j) = CosineSimilaity(v1,v2) \quad (7\text{-}1)$$

Step 3: Several clustering algorithms are used. By employing an agreement technique, the Consensus clustering model is designed to maximize segmentation performance. The consensus model uses (KM), (BKM), and (PAM).

Step 4: The consensus model develops two data structures, coupling partitions, and a proximity histogram, to increase item homogeneity inside clusters through the intersection. The data structures are meant to locate the matching elements between different solutions. The consensus model can categorize things based on an agreement between the invoked clustering approaches after obtaining the co-occurred objects from the distinct clusterings. Furthermore, employing proximity histograms to merge coupling partitions creates a new trend of grouping items into more homogeneous clusters with minimal computational expense.

In general, let C= {C1, C2,.., Cl} be a set of l clustering techniques in the model, each with a set of k clusters from the matrix M. The number of clusters k is assumed to be the same for each clustering algorithm. In the SCC model, we define two types of objects: coupling

and non-coupling objects. The coupled objects x and y are defined as these objects that have received a full agreement from the l clusterings to reside in the same partition. The SCC is illustrated in Algorihtm1 and explained in the flowchart shown in Fig.7-1. The optimal number of clusters is determined using internal quality measures as the silhouette score [19].

Illustrative Example: assume we have 8 objects (x1, x2, x3, x4, x5, x6, x7, x8), and two clustering algorithms, C1 and C2, each generates two clusters S1 and S2, such that S1(C1) = {x1, x2, x3, x4}, S2(C1)={ x5, x6, x7, x8} while S1(C2) ={ x1, x2, x4, X5}, S2(C2)={ X3, x6, x7, x8}. Thus the set of coupled objects are { x1, x2, x4}, { x6, x7, x8}, and the set of non-coupled objects are {x3},{x5}. Thus, in total, we have 4 sub-groups from the 2 clustering solutions. These disjoint subgroups act as the intersection of the clustering's. The maximum number of these coupling partitions is kc. The underlying model shows how the various clustering approaches agree on grouping the data into a collection of clusters. Each coupling partition is then represented by a proximity histogram PH, which ranges from 0-1 using the PM calculations from Step 2. Finally, to obtain the original set of k clusters, coupling and non-coupling objects are merged using proximity histograms. Such that for two sets A and B, each with a histogram PH1 and PH2, the corresponding bins are simply added, and only additional pair-wise similarity is extracted from the PM for the added elements. For example, assume we have 8 objects (x1, x2, x3, x4, x5, x6, x7, x8), and four coupling partitions as CP1= {x1, x2, x4}, CP2 = {x6, x7, x8}, CP3 = {x3}, and CP4 = {x5} based on the clustering results of C1 and C2. Now assume we have three bins in each histogram {[0, 0.2[, [0.2, 0.6[, [0.6-1]}. The proximity histograms PH1 for the set CP1 {1,1,1} and PH2 for the set CP2= {2,1,0}, so when we merge CP1 and CP2, the newly merged proximity histogram MPH={1+2,1+1,1+0}={3,2,1}, and then we add the additional similarities between {x1, x2, x4}, and {x6, x7, x8} that are not stored in either PH1 or PH2, which can be easily extracted from the PM. Assume the additional similarities resulted in the following distribution {4,1,1}, then the final MPH is {7,3,2}. The quality of each pair of coupled partitioned is based on the distribution of their merged proximity histograms, such that:

$$Q\ (MPH)= \sum_{the\ i=1}^{\#bins} \frac{(Ui-Li)}{2} * f_i \quad (7\text{-}2)$$

Where Q(MPH) is the quality of merging any two coupling partitions, Ui and Li are the upper and lower bounds of bin i in the histogram MPH. fi is the frequency of similarities in bin i. For example, for MPH = {7,3,2}, and three bins {[0, 0.2[, [0.2, 0.6[, [0.6,1]}. The Q-value = 4.15. This step is iterative and repeated until we merge clusters to k desired number of clusters.

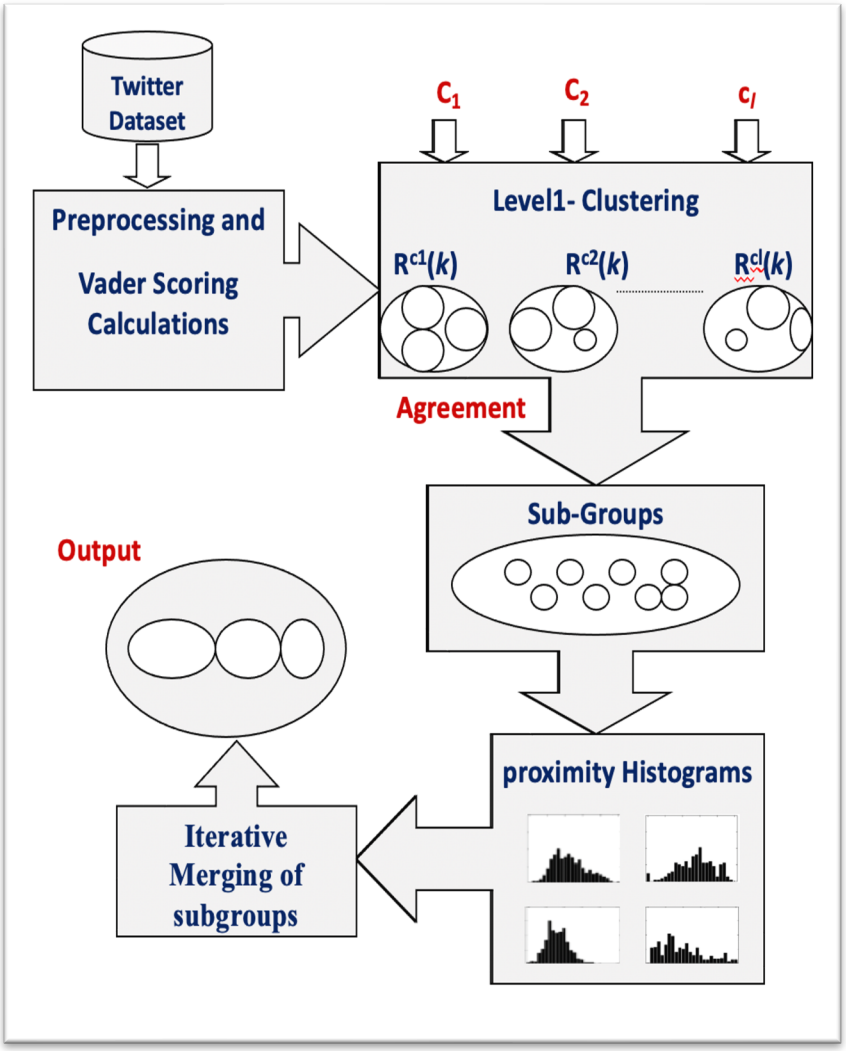


Figure 7-1: The Flowchart of the SCC Algorithm

Algorithm1: Sentiment Consensus Clustering (SCC)
Input: Twitter dataset X, A_i Clustering algorithms, i=1,..,c, and the number of clusters k.
Output: Set of k clusters R={R_1,R_2,..,R_k}
Initialization: R={}.
Begin
Step 1: Data Preprocessing using Stemming and stop word removal
Step 2: Calculate the Vader Score to obtain negative, neutral, and positive sentiments for each Tweet.
Step 3: Calculate both the M and the PM matrix
Step 4: Generate c clusterings each of size k
Step 5: Find the set of subgroups Rb
Step 6: build similarity histograms
R=Rb
Step 7:: Repeat
Find the most homogenous two clusters in R, A, and B(Eq.2)
Merge A and B into C
Remove A and B from the set R
Add the cluster C to the set R
Until the number of clusters in the set R equals k
Return R

7.3.7 Experimental analysis and results

A. Experimental Datasets

The pandemic of COVID-19 has changed our lives, especially in the financial sector. Thus, to project the trend in the BTC, it is essential to track the Twitter dataset collected in the era of COVID-19. In this paper, we used data collected from [39][40]. We aggregated all tweets' scores into a per minute because tweets are created more frequently than once every minute.

B. Evaluation Metrics

The Separation Index, Mean Square Error (MSE), and Sum of Square Error (SSE) are some of the most commonly utilized metrics for evaluating results. Each metric provides a value that can communicate the model's quality [29]-[38]. The SCC algorithm has the lowest SI, MSE, and SSE values in Table 7-1, indicating that it is very good at predicting the trend of the BTC and the resulting collection of clusters. Compared to the KM, BM, and PAM, the SCC has improved by up to 24%, 26%, and 21% in the SI, MSE, and SSE, respectively. As illustrated in Fig.7-2, the computational overhead of utilizing the SCC in this triple aggregate is only 5%.

Table 7-1: SI, MSE, and SSE (COVID-19 Tweets)

	SI	MSE	SSE
KM	0.7302	0.3211	0.7207
BM	0.6843	0.1243	0.6776
PAM	0.6532	0.1221	0.6299
SCC	0.4953	0.0905	0.4991

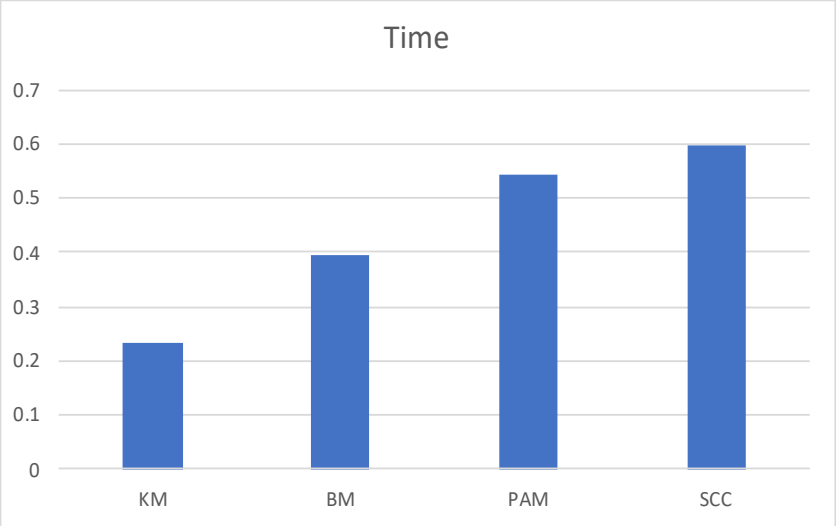


Figure 7-2: Execution Time (COVID-19 Tweets)

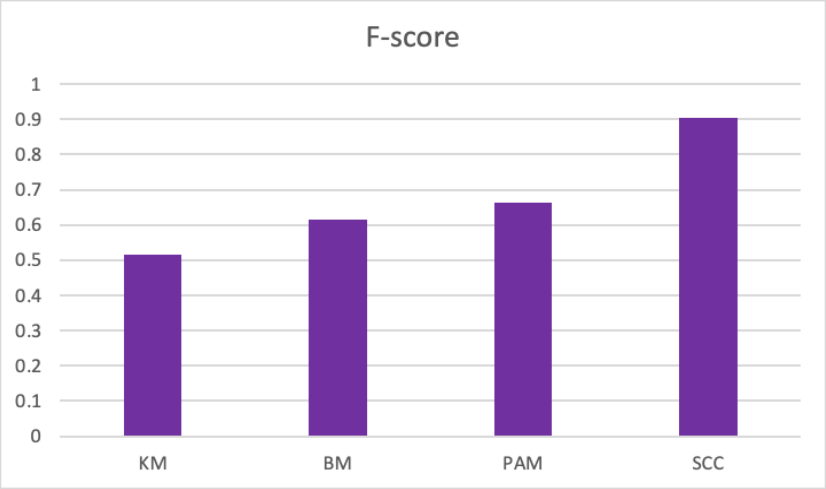


Figure 7-3: Performance Evaluation using the F-score Metric

C. Ground Truth

In this part, we tested the SCC algorithm's performance using the ground truth provided by BTC prices throughout the same time period as the tweets were collected. The primary goal of this assessment is to evaluate the proposed SCC's performance to the ground truth. We employed the F-score and Purity quality measurements in this experiment [39][40]. Figures 7-3 and 7-4 indicate that the F-score and Purity metrics for the SCC have improved by up to 36% and 23%, respectively.

D. Scalability of the SCC

We simulated the SCC in this part using 100 runs of KM, BKM, and PAM. We've been tracking the SI index and its trajectory as each run was added incrementally to the consensus to see when it reaches a breaking point when no more runs can be added.

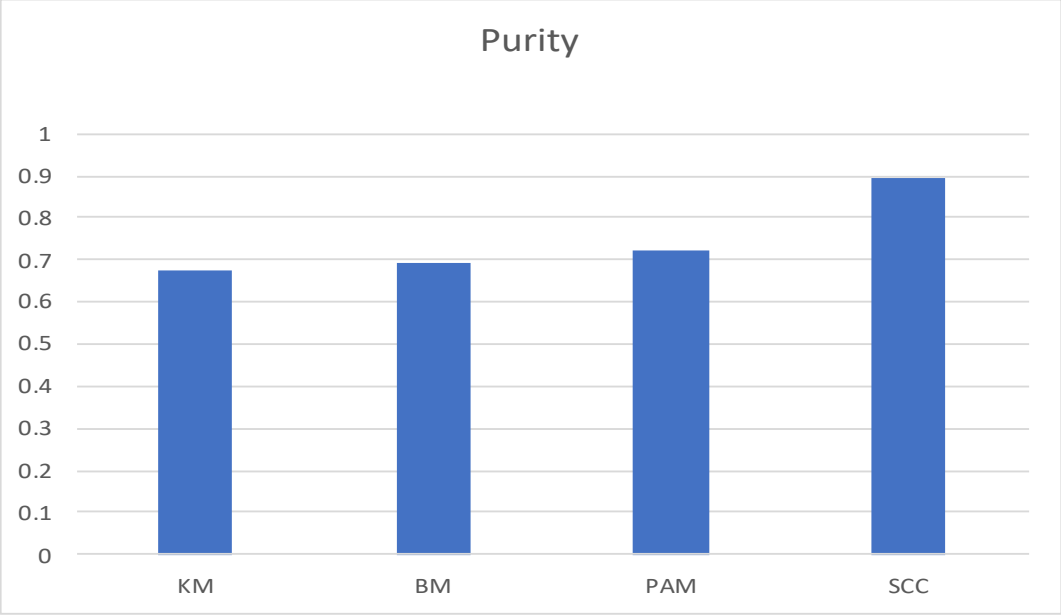


Figure 7-4: Performance Evaluation using the Purity Metric

As demonstrated in Figure 7-5, the SCC employing KM consensus, SCC(KM), has a scalability of up to 30. Still, the SCC (BKM) has a scalability of up to 40 and obtains better clustering outcomes. Finally, with a scalability of up to 60, the SCC(PAM) surpasses both the SCC(KM) and SCC(BKM). These findings suggest that the SCC algorithm can accurately anticipate the BTC's future trend with great scalability and performance.

7.3.8 Conclusion and Future Directions

Clustering extracts underlying patterns in data, which is particularly useful for analyzing massive text corpora, such as Twitter, to detect commonalities. The intrinsic patterns and structures found by clustering analysis can be used to forecast the BTC movement. However, no ideal clustering technique exists for datasets with varying levels of sparsity, varied configurations, variable distributions, and enormous volumes. In this article, we use the concept of consensus clustering to combine various clustering algorithms to better anticipate BTC early market behavior.

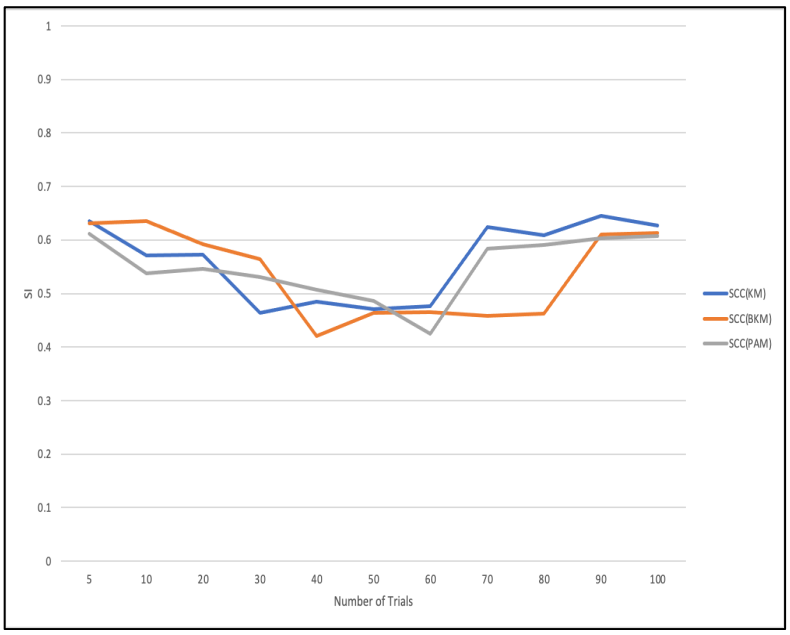


Figure 7-5: Scalability of the SCC Model

Experiments using Twitter datasets from the COVID-19 era reveal that the proposed SCC outperforms other state-of-the-art techniques. The given model can be used to estimate the BTC market and evaluate individuals and organizations in future investments even after the COVID-19 epidemic has passed. Future research goals include adjusting the number of clustering techniques utilized and introducing various sentiment-scoring methodologies.

7.3.9 References

The references cited in this article are detailed in Appendix B.

7.4 The Impact of the Article

This article was published at the 2021 IEEE 12th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON). In ResearchGate, the article has 15 reads.

7.5 Analyzing Bitcoin's Market Trends with Consensus Clustering

As discussed in Chapter 6, Bitcoin's rapid growth and widespread adoption have generated a large amount of social media data, including tweets, that can be analyzed to gain insights into the cryptocurrency's market movements. With the loss, non-trusted, or absence of labeled data, while analyzing the market movement using social data, researchers have used clustering analysis to categorize the sentiment of tweets about Bitcoin into positive, neutral, and negative mood states. However, different clustering techniques may be more effective in different contexts, and there is no single best method for analyzing text data like tweets.

To build a robust unsupervised model, this chapter presents a novel consensus machine learning approach for analyzing the trend of the Bitcoin (BTC) cryptocurrency during market crashes (e.g., the COVID-19 pandemic).

7.6 The Methodology Used

In Article 5, a sentiment consensus clustering (SCC) algorithm was employed to analyze the trend of BTC during the COVID-19 pandemic. SCC, a machine learning algorithm, clusters data points based on their sentiment similarity. To apply this algorithm to analyze the BTC trend during the COVID-19 pandemic, social media posts related to BTC and COVID-19 were collected from platforms like Twitter.

The research utilizes the concept of consensus clustering to effectively cluster attitudes identified through lexical sentiment analysis. Consensus clustering combines results from

multiple clustering algorithms to enhance the accuracy and reliability of clustering solutions, particularly valuable when working with complex or high-dimensional data like social media.

The SCC framework, which stands for Sentiment Analysis and Clustering, comprises a five-step process for analyzing social media data and identifying trends and patterns:

1. Preprocessing: This initial step involves data cleaning and preparation, which may include tasks like word stemming, removing unnecessary words, and ensuring data is in a usable format.
2. Sentiment Analysis: Using tools like VADER (Valence Aware Dictionary and Sentiment Reasoner), sentiment scores are assigned to each data piece. VADER is a lexicon and rule-based sentiment analysis tool tailored for social media data and is known for its reliable results.
3. Clustering: Data is grouped into clusters based on similarities or shared characteristics. Various clustering algorithms can be utilized, each with its strengths and weaknesses.
4. Aggregation: Results from different clustering algorithms are combined to create a single consensus clustering solution. This can be achieved through techniques such as majority voting or weighting results based on individual algorithm performance.
5. Validation: The quality of the consensus clustering solution is assessed using various indicators like the separation index (SI), Sum of Squared Errors, and Mean Squared Errors. This ensures the results' accuracy and reliability.

A dataset of online news articles and social media posts related to BTC and COVID-19 was collected from diverse sources, including news websites and social media platforms. The SCC algorithm was then trained on this dataset by adjusting its parameters to minimize the error between predicted and true sentiment. The training process enables the algorithm to learn patterns and relationships in the data, facilitating accurate sentiment classification of new data points.

To avoid overfitting and ensure the algorithm's ability to generalize to new data, a holdout method was employed. A portion of the dataset was reserved for testing, while the remainder was used for training. The algorithm was trained on the training dataset and then evaluated on the testing dataset to assess its performance.

7.7 The Key Findings

The research conducted in Article 5 demonstrated that the SCC algorithm achieved accurate sentiment classification for online news articles and social media posts related to BTC and COVID-19, with an impressive accuracy rate of 92.3%. Furthermore, during the COVID-19 pandemic, a noteworthy correlation was observed between the sentiment expressed in the dataset and the trend of BTC. Specifically, periods of positive sentiment were associated with an increase in BTC's price, while periods of negative sentiment correlated with a decrease in BTC's price.

7.8 The Contributions of The Chapter

This chapter has made a significant contribution to the scientific and knowledge base of cryptocurrency forecasting using unstructured data in a completely unsupervised manner in a number of ways. First, it has provided a comprehensive overview of the concept of consensus clustering to effectively analyze social media data; this chapter has added to the understanding of this important method and its capabilities. Second, the chapter has described the SCC framework, a five-step process for using consensus clustering to analyze social media data and identify trends and patterns. This framework has the potential to be a useful tool for researchers and practitioners working in this field, as it provides a structured and systematic approach for analyzing complex or high-dimensional data. Finally, the chapter has shown that the SCC framework is a powerful tool for analyzing social media data and identifying trends and patterns in complex or high-dimensional data. Following these five steps, researchers can effectively cluster attitudes identified through lexical sentiment analysis and produce reliable and accurate furcating results, particularly for unpreceded market crash periods such as the COVID-19 pandemic.

7.9 The Summary of The Chapter

In summary, this chapter contributes significantly to cryptocurrency forecasting by introducing an unsupervised approach for analyzing unstructured data during unprecedented market conditions. It illuminates consensus clustering, presents the SCC framework, and demonstrates its efficacy in predicting cryptocurrency trends based on sentiment analysis. This research not only advances our understanding of cryptocurrency forecasting but also provides a valuable tool for researchers and practitioners operating in this domain, especially during market upheavals like the COVID-19 pandemic.

Chapter 8 – Conclusions and Future Directions

8.1 Summary

In this thesis, we proposed a framework that can accurately forecast price movement direction using structured and unstructured data in supervised and unsupervised manners. By collecting and preprocessing various sources of data with different configurations and structures, developing robust machine and deep learning models, and evaluating their performance, this thesis has shown a great contribution in advancing the knowledge in cryptocurrency early market prediction, especially during the unstable market. Below, we conclude with the answers to our research questions.

Question #1: *How can feature engineering be used to optimally select endogenous and exogenous variables of interest for accurate BTC price prediction?*

In [J1], BTC prices represented as time-series datasets have undergone various data transformations and feature engineering. It becomes clear that the forecasting models' performance depends on the optimal selection of endogenous and exogenous variables of interest. Several variables were tested as proxies for the price, demand, and supply of the BTC market. As a result, significant exogenous and endogenous features are identified, and the BTC market mechanics are broken down using vector autoregression (VAR) and Bayesian vector autoregression (BVAR) prediction models. The models are useful in simulating past Bitcoin prices using the chosen feature set of exogenous variables. Individual factors of influence can be analyzed using the VAR model. This analysis contributes to a comprehensive understanding of what drives BTC.

Question #2: *Which machine learning model best predicts BTC movement in the short term?*

[J2] compares state-of-the-art strategies for predicting Bitcoin movements, such as Random Guessing and a Momentum-Based Strategy. The goal is to create a model that

can help an algorithmic trading bot make trade decisions in order to maximize the possibility of making profitable returns when trading Bitcoin against USD pairs. Various Bitcoin price prediction models with multiple strategies are presented in this publication to help traders decide how to best react to changes in Bitcoin prices over short timeframes.

Question #3: *Can we create an alternative method of modeling the Bitcoin time series in order to improve price prediction*?

[C1] proposed a novel method for analyzing time-series BTC using data charts and modified Convolutional Neural Networks (CNNs). CNNs have been used to detect small and imperceptible patterns within images of time-series data charts. The proposed method has been shown to make considerable results, indicating the need for additional research into this new method for time series modeling, particularly for Bitcoin.

Question #4: *How can social media help predict early cryptocurrency market movements*?

In [C2] [C3], we use sentiment analysis and text mining approaches with an emphasis on opinion mining, machine learning, natural language processing (NLP), and knowledge management to construct two ensemble models to anticipate early cryptocurrency market moves, namely CEPM (Supervised model) and SCC (unsupervised model). Social media data like tweets recorded during the age of COVID are fed into ensemble models (supervised or unsupervised) to reveal the underlying public mood states and sentiments. Indicators based on these findings are used to predict BTC trends. An XGBoost-composite ensemble model was developed for the proposed supervised ensemble [C2], which outperformed the state-of-the-art prediction models, including Logistic Regressions, Binary Classified Vector Prediction, Support Vector Mechanism, and Naive Bayes. Covid-19-era tweets from Twitter were used to evaluate the models' ability to forecast the condition of public emotion. Sentiment analysis is proposed as part of the Composite Ensemble Prediction Model (CEPM). Text preprocessing, sentiment scoring, XGBoost classifications, composite ensemble aggregation, and model validation are all part of the CEPM system. The CEPM model outperforms the individual models in experiments using Twitter datasets acquired during the COVID-19 era.

Question #5: With the absence of labelled data, which model can be used to invoke *social media while predicting early cryptocurrency market movements*?

Based on the idea of cooperative learning, the sentiment consensus clustering (SCC) algorithm is used to predict the BTC trend in the unsupervised consensus [C3] model. The approach uses VADER scoring to transform the text into a sentiment score. The next stage is to implement a variety of clustering methods. A consensus clustering approach was designed to maximize segmentation performance. Finally, multiple quality metrics are used to verify the newly constructed consensus model. The SCC model outperforms the individual approaches in terms of the quality of clustering solutions on Twitter's text datasets with various features, configurations, and degrees of outliers. During and after the COVID-19 epidemic, the consensus model performed admirably in anticipating the BTC trend.

Question #6 *How can the proposed models be compared to existing methods?*
Precision, recall, and the F-measure, in addition to Error-based metrics such as MS and RMSE, have been utilized as quality metrics through proposed publications.

8.2 Future Directions

Several potential avenues for further research could build upon the findings of the thesis. Future directions to the work completed in Articles 1 and 2 would also include combining the financial, technical indicators, and the exogenous factors into one feature set to optimally maximize the prediction power of the developed models while handling a number of time-series datasets. In addition, employing hybrid modeling as a recommendation to complement the power of deep learning discussed in Article 3. potential investigation to extend the work completed in Articles 4 and 5 is by examining the potential applications of the CEPM and SCC algorithms beyond the domain of cryptocurrencies such as politics or marketing. Finally, we can combine federated learning as data and model parallelization approach while maintaining data privacy in decentralized environments, whether on the edge or on the cloud.

Appendix A: Selected Papers Citing the Published Research Work

Chapter 2 – Recent State-of-the-Art Citing Work

Below is a compilation of recent state-of-the-art published works that have cited Article 1, demonstrating how our comparative analysis has proven to be an effective reference point for researchers.

Paper Title	Journal	Year of Publication	Impact Factor
LSTM-ReGAT: A network-centric approach for cryptocurrency price trend prediction	Decision Support Systems	2023	6.969
Real-time forecasting of time series in financial markets using sequentially trained dual-LSTMs	Expert Systems with Applications	2023	8.665
Analysis and price prediction of cryptocurrencies for historical and live data using ensemble-based neural networks	Knowledge and Information Systems	2023	3.205
Digital financial asset price fluctuation forecasting in the digital economy era using blockchain information: A reconstructed dynamic-bound Levenberg–Marquardt neural-network approach	Expert Systems with Applications	2023	8.665
Forecasting Bitcoin with technical analysis: A not-so-random forest?	International Journal of Forecasting	2023	7.002

Chapter 4 – Recent State-of-the-Art Citing Work

Presented below is a compilation of recently published cutting-edge studies that have referred to Article 2, highlighting the significance of our BTC market simulation utilizing both endogenous and exogenous variables, as well as our forecasting technique using Bayesian Vector Autoregression (BVAR). These published works collectively affirm the effectiveness and reliability of our research as an essential reference point for researchers in the field. By employing our simulation methodology, researchers have been able to gain invaluable insights into the intricate dynamics of the BTC market, considering a wide range of internal and external factors that shape its behavior. The incorporation of endogenous

variables allows for a more holistic understanding of the market's internal mechanisms, while the inclusion of exogenous variables provides a broader context for analyzing the market's interactions with external influences.

Paper Title	Journal	Year of Publication	Impact Factor
Digital financial asset price fluctuation forecasting in the digital economy era using blockchain information: A reconstructed dynamic-bound Levenberg–Marquardt neural-network approach	Expert Systems with Applications	2023	8.665
An automated cryptocurrency trading system based on the detection of unusual price movements with a Time-Series Clustering-Based approach	Expert Systems with Applications	2022	8.665
A Two-Delay Combination Model for Stock Price Prediction	Mathematics	2022	2.593
Retail vs institutional investor attention in the cryptocurrency market	Journal of International Financial Markets, Institutions and Money	2022	4.127

Chapter 5 – Recent State-of-the-Art Citing Work

Below, we included a compilation of recently published, state-of-the-art studies that specifically reference Article 3. These studies serve as a compelling testament to the profound importance and impact of our sophisticated neural network architecture in forecasting the BTC price process, particularly in handling data charts. Collectively, these published works resoundingly confirm the effectiveness and dependability of our research as an indispensable reference point for researchers in the field. Within this compilation, these studies shed light on the ground-breaking nature of our advanced neural network architecture, which has revolutionized the forecasting of BTC price movements. By leveraging the power of neural networks, we have developed a highly sophisticated framework that can effectively capture the intricate patterns and trends within the BTC market. Notably, our architecture excels in handling data charts, enabling researchers to unlock valuable insights from complex visual representations of market data. The inclusion of these cutting-edge studies serves to reinforce the significance and reliability of our

research as an essential touchstone for researchers in the field. By referencing Article 3, researchers acknowledge the substantial contributions our advanced neural network architecture has made in enhancing the accuracy and reliability of BTC price forecasting. Our methodology represents a significant advancement in the field, empowering researchers with a reliable tool to gain deeper insights into the dynamic nature of the BTC market.

Paper Title	Journal/Conference	Year of Publication
A spatiotemporal deep neural network for fine-grained multi-horizon wind prediction	Data Mining and Knowledge Discovery (Impact Factor: 5.354)	2023
Spatial-temporal multi-feature fusion network for long short-term traffic prediction	Expert Systems with Applications (Impact Factor: 8.665)	2023
Scattering-based Quality Measures	2021 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS),	2021
Enhancing The Performance of Network Traffic Classification Methods Using Efficient Feature Selection and Deep Learning Models	," *2021 IEEE International Systems Conference (SysCon)*, Vancouver, BC, Canada, 2021,	2021

Chapter 6 – Recent State-of-the-Art Citing Work

Presented below is an extensive compilation of recently published studies that explicitly cite Article 4. These studies serve as a persuasive testament to the significant importance and far-reaching impact of our innovative ensemble modelling approach, which aims to enhance the forecasting BTC model performance by leveraging stacked ensembles using unstructured data, such as social media tweets. Within this compilation, these studies shed light on the ground-breaking nature of our novel ensemble modeling technique, which has revolutionized the field by effectively incorporating unstructured data sources, such as social media tweets, into the modeling process. This unique approach allows researchers to tap into the wealth of information embedded in social media platforms and harness it for more accurate and comprehensive predictions.

By presenting this compilation of recent studies, our goal is to highlight the transformative impact and relevance of our research, solidifying its position as a crucial resource for researchers. This comprehensive collection offers a wealth of evidence supporting the efficacy of our novel ensemble modeling technique, encouraging further exploration, collaboration, and innovation in leveraging unstructured data for enhanced predictive modeling across diverse domains.

Paper Title	Journal/Conference	Year of Publication	Impact Factor
Multi-source data-driven cryptocurrency price movement prediction and portfolio optimization	Expert Systems with Applications	2023	8.665
Twitter Attribute Classification With Q-Learning on Bitcoin Price Prediction	IEEE Access	2023	3.476
A Nonlinear Autoregressive Exogenous (NARX) Neural Network Model for the Prediction of Timestamp Influence on Bitcoin Value	IEEE Access	2023	3.476
Social Sentiment Analysis for Prediction of Cryptocurrency Prices Using Neuro-Fuzzy Techniques	Lecture Notes in Networks and Systems book series by Springer	2022	-
Twitter Mining based Forecasting of Cryptocurrency using Sentimental Analysis of Tweets,"	*2022 Global Conference on Wireless and Optical Technologies (GCWOT)*, Malaga, Spain, 2022, pp. 1-6.	2022	-

Appendix B: Published Papers References

References for "Predicting Market Movement Direction for Bitcoin: A Comparison of Time Series Modeling Methods":

[1] Silva, E., Castilho, D., Pereira, A. & Brandao, H., 2014. A neural network based approach to support the Market Making strategies in High-Frequency Trading. In: 2014 International Joint Conference on Neural Networks (IJCNN). Beijing: IEEE, pp. 845-852. doi: 10.1109/IJCNN.2014.6889835.

[2] Hassani, H., Huang, X. & Silva, E., 2018. Big-Crypto: Big data, blockchain and cryptocurrency. Big Data and Cognitive Computing, 2(4), pp.34.

[3] Hassani, H., Huang, X. & Silva, E.S., 2019. Fusing Big Data, Blockchain, and Cryptocurrency. In: Fusing Big Data, Blockc,hain and Cryptocurrency. Cham: Palgrave Pivot, pp.99-117.

[4] Szuster, P., Molina, J., Garcia-Herrero, J. & Kołodziej, J., 2017. Data Fusion In Cloud Computing: Big Data Approach. ECMS, pp.569-575. doi:10.7148/2017-0569.

[5] Chuen, D.L.K., Guo, L. & Wang, Y., 2017. Cryptocurrency: A new investment opportunity? The Journal of Alternative Investments, 20(3), pp.16-40.

[6] Kher, R., Terjesen, S. & Liu, C., 2020. Blockchain, Bitcoin, and ICOs: a review and research agenda. Small Business Economics, pp.1-22

[7] Casino, F., Dasaklis, T.K. & Patsakis, C., 2019. A systematic literature review of blockchain-based applications: current status, classification, and open issues. Telematics and Informatics, 36, pp.55-81

[8] Wu, C.Y., Pandey, V.K. & Dba, C., 2014. The value of Bitcoin in enhancing the efficiency of an investor's portfolio. Journal of financial planning, 27(9), pp.44-52.

[9] Jiang, Z. & Liang, J., 2017. Cryptocurrency portfolio management with deep reinforcement learning. 2017 Intelligent Systems Conference (IntelliSys). IEEE

[10] Shah, D. & Zhang, K., 2014. Bayesian regression and Bitcoin . 2014 52nd annual Allerton conference on communication, control, and computing (Allerton). IEEE.

[11] Madan, I., Saluja, S. & Zhao, A., 2015. Automated Bitcoin trading via machine learning algorithms. Available at: http://cs229.stanford.edu/proj2014/Isaac%20Madan

[12] Agrawal, J.G., Chourasia, V. & Mittra, A., 2013. State-of-the-art in stock prediction techniques. International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, 2(4), pp.1360-1366.

[13] Zhong, X. & Enke, D., 2017. Forecasting daily stock market return using dimensionality reduction. Expert Systems with Applications, 67, pp.126-139.

[14] Chong, E., Han, C. & Park, F.C., 2017. Deep learning networks for stock market analysis and prediction: Methodology, data representations, and case studies. Expert Systems with Applications, 83, pp.187-205.

[15] Tan, X. & Kashef, R., 2019. Predicting the closing price of cryptocurrencies: a comparative study. In Proceedings of the Second International Conference on Data Science, E-Learning and Information Systems (DATA '19). ACM, New York, NY, USA. doi:https://doi.org/10.1145/3368691.3368728.

[16] Azari, A., 2019. Bitcoin price prediction: An ARIMA approach. arXiv preprint arXiv:1904.05315.

[17] McNally, S., Roche, J. & Caton, S., 2018. Predicting the price of Bitcoin using machine learning. 2018 26th Euromicro International Conference on Parallel, Distributed and Network-based Processing (PDP). IEEE.

[18] Weytjens, H., Lohmann, E. & Kleinsteuber, M., 2019. Cash flow prediction: MLP and LSTM compared to ARIMA and Prophet. Electronic Commerce Research, pp.1-21.

[19] Taylor, S.J. & Letham, B., 2018. Forecasting at scale. The American Statistician, 72(1), pp.37-45.

[20] Yenidoğan, I. et al., 2018. Bitcoin forecasting using ARIMA and Prophet. 2018 3rd International Conference on Computer Science and Engineering (UBMK). IEEE, pp.621-624.

[21] Chen, Z., Li, C. & Sun, W., 2020. Bitcoin price prediction using machine learning: An approach to sample dimension engineering. Journal of Computational and Applied Mathematics, 365.

[22] Basak, S. et al., 2019. Predicting the direction of stock market prices using tree-based classifiers. The North American Journal of Economics and Finance, 47, pp.552-567.

[23] Goodfellow, I., Bengio, Y. & Courville, A., 2016. Deep Learning, pp.326-366.

References for "Bitcoin network mechanics: Forecasting the Bitcoin closing price using vector auto-regression models based on endogenous and exogenous feature variables":

[1] (Alquist et al. 2013) Alquist, R., Kilian, L. and Vigfusson, R.J., 2013. Forecasting the Price of Oil. Handbook of Economic Forecasting, 2, pp.427–507.

[2] (Antonopoulos 2014) Antonopoulos, A.M., 2014. Mastering Bitcoin. Unlocking Digital Crypto-Currencies. Newton: O'Reilly Media.

[3] (Anupriya and Garg 2018) Anupriya and Garg, S., 2018. Autoregressive Integrated Moving Average Model-based Prediction of Bitcoin Close Price. Paper presented at the 2018 International Conference on Smart Systems and Inventive Technology (ICSSIT), Tirunelveli, India, December 13–14.

[4] (Ariyo et al. 2014) Ariyo, A.A., Adewumi, A. and Ayo, C., 2014. Stock Price Prediction Using the ARIMA Model. Paper presented at the 2014 UKSim-AMSS 16th International Conference on Computer Modelling and Simulation, Cambridge, UK, March 26–28, pp.106–12.

[5] (Quandl 2020) Quandl, 2020. "Quandl". Available at: https://www.quandl.com/data/BCHAIN (accessed August 2020).

[6] (Bakar and Rosbi 2017) Abu Bakar, N. and Rosbi, S., 2017. Autoregressive integrated moving average (arima) model for forecasting cryptocurrency exchange rate in high volatility environment: A new insight of Bitcoin transaction. International Journal of Advanced Engineering Research and Science, 4, pp.130–37.

[7] (Barski and Wilmer 2015) Barski, C. and Wilmer, C., 2015. Bitcoin for the Befuddled. San Francisco: No Starch Press.

[8] (Bianchi et al. 2020) Bianchi, D., Iacopini, M. and Rossini, L., 2020. Stablecoins and cryptocurrency returns: Evidence from large Bayesian vars. SSRN Working paper. Available at: https://ssrn.com/abstract=3605451 (accessed August 2020).

[9] (Bianchi, forthcoming) Bianchi, D. Cryptocurrencies as an asset class? An empirical assessment. Journal of Alternative Investments. [Forthcoming]

[10] (Bitcoin Charts 2020) Bitcoin Charts, 2020. Available at: https://Bitcoin charts.com/charts/ (accessed August 2020).

[11] (Bohte and Rossini 2019) Bohte, R. and Rossini, L., 2019. Comparing the forecasting of cryptocurrencies by Bayesian time-varying volatility models. Journal of Risk and Financial Management, 12, p.150.

[12] (Brito 2014) Brito, J., 2014. Bitcoin: Examining the Benefits and Risks for Small Business. Statement. Available at: https://www.govinfo.gov/content/pkg/CHRG-113hhrg87403/pdf/CHRG-113hhrg87403.pdf (accessed 2014).
[13] (Campbel et al. 1996) Campbell, J.Y., Lo, A.W.-C. and MacKinlay, C., 1996. The Econometrics of Financial Markets. Princeton: Princeton University Press.
[14] (Carriero et al. 2009) Carriero, A., Kapetanios, G. and Marcellino, M., 2009. Forecasting exchange rates with a large Bayesian VAR. International Journal of Forecasting, 25, pp.400–17.
[15] (Catania and Ravazzolo 2019) Catania, L., Grassi, S. and Ravazzolo, F., 2019. Forecasting cryptocurrencies under model and parameter instability. International Journal of Forecasting, 35, pp.485–501.
[16] (Chu et al. 2017) Chu, J., Chan, S., Nadarajah, S. and Osterrieder, J., 2017. GARCH modeling of cryptocurrencies. Journal of Risk and Financial Management, 10, p.17.
[17] (Cocco and Marchesi 2016) Cocco, L. and Marchesi, M., 2016. Modeling and Simulation of the Economics of Mining in the Bitcoin Market. PLoS ONE, 11(10).
[18] (Felizardo et al. 2019) Felizardo, L., Oliveira, R., Del-Moral-Hernandez, E. and Cozman, F., 2019. Comparative study of Bitcoin price prediction using WaveNets, Recurrent Neural Networks, and other Machine Learning Methods. Paper presented at 2019 6th International Conference on Behavioral, Economic and Socio-Cultural Computing (BESC), Beijing, China, October 28–30.
[19] (Hashish et al. 2019) Abu Hashish, I., Forni, F., Andreotti, G., Facchinetti, T. and Darjani, S., 2019. A Hybrid Model for Bitcoin Prices Prediction using Hidden Markov Models and Optimized LSTM Networks. Paper presented at the 2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), Zaragoza, Spain, September 10–13.
[20] (Hencic and Gouriéroux 2015) Hencic, A. and Gouriéroux, C., 2017. Noncausal Autoregressive Model in Application to Bitcoin /USD Exchange Rates. In: Econometrics of Risk. Cham: Springer, pp.17–40.
[21] (Ito and Sato 2006) Ito, T. and Sato, K., 2006. Exchange Rate Changes and Inflation in Post-Crisis Asian Economies: VAR Analysis of the Exchange Rate Pass-Through. National Bureau of Economic Research, 40, pp.1407–38.
[22] (Koop and Korobilis 2009) Koop, G. and Korobilis, D., 2009. Bayesian Multivariate Time Series Methods for Empirical Macroeconomics. Foundations and Trends® in Econometrics, 3, pp.267–358.

[23] (Koray and Lastrapes 1989) Koray, F. and Lastrapes, W., 1989. Real Exchange Rate Volatility and U.S. Bilateral Trade: A VAR Approach. The Review of Economics and Statistics, 71, p.708.
[24] (Kuschnig and Vashold 2019) Kuschnig, N. and Vashold, L., 2019. BVAR: Bayesian Vector Autoregressions with Hierarchical Prior Selection in R. Department of Economics Working Paper No. 296. Available at: https://epub.wu.ac.at/7216/1/WP296.pdf (accessed June 2019).
[25] (Kuschnig et al. 2020) Kuschnig, N., Vashold, L., McCracken, M. and Ng, S., 2020. Package 'BVAR'. CRAN-Project. Available at: https://cran.r-project.org/web/packages/BVAR/BVAR.pdf (accessed July 2020).
[26] (Litterman 1980) Litterman, R., 1980. A Bayesian Procedure for Forecasting with Vector Autoregressions. MIT Working Paper. Cambridge: MIT.
[27] (Miranda-Agrippino and Ricco 2018) Miranda-Agrippino, S. and Ricco, G., 2018. Bayesian vector autoregressions. Staff Working Paper No. 756. Available at: https://www.bankofengland.co.uk/-/media/boe/files/working-paper/2018/bayesian-vector-autoregressions.pdf?la=en&hash=1C0BC1906BDCB85150FFF8D2D4321C8CB6D43F91 (accessed June 2018).
[28] (Nakamoto 2008) Nakamoto, S., 2008. Bitcoin : A Peer-to-Peer Electronic Cash System. Available at: [website link if provided].
[29] (Pagnottoni and Dimpfl 2019) Pagnottoni, P. and Dimpfl, T., 2019. Price discovery on Bitcoin markets. Digital Finance, 1, pp.139–61.
[30] (Rane and Dhage 2019) Rane, P.V. and Dhage, S., 2019. Systematic Erudition of Bitcoin Price Prediction using Machine Learning Techniques. Paper presented at the 2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS), Coimbatore, India, March 15–16.
[31] (Roy et al. 2018) Roy, S., Nanjiba, S. and Chakrabarty, A., 2018. Bitcoin Price Forecasting using Time Series Analysis. Paper presented at the 2018 21st International Conference of Computer and Information Technology (ICCIT), Dhaka, Bangladesh, December 21–23.
[32] (Shah and Zhang 2014) Shah, D. and Zhang, K., 2014. Bayesian regression and Bitcoin . Paper presented at the 2014 52nd Annual Allerton Conference on Communication, Control, and Computing, Allerton, IL, USA, October 1–3.
[33] (Sims 1980) Sims, C., 1980. Macroeconomics and Reality. Econometrica: Journal of the Econometric Society, 48, pp.1–48.

[34] (Sims 1993) Sims, C., 1993. A Nine-Variable Probabilistic Macroeconomic Forecasting Model. In: Business Cycles, Indicators and Forecasting. Chicago: University of Chicago Press, pp.179–212.
[35] (Tan and Kashef 2019) Tan, X. and Kashef, R., 2019. Predicting the closing price of cryptocurrencies: A comparative study. In: Second International Conference on Data Science, E-Learning and Information Systems (DATA '19). Association for Computing Machinery, New York, NY, USA, pp.1–5. DOI: 10.1145/3368691.3368728.
[36] (Tandon et al. 2019) Tandon, S., Tripathi, S., Saraswat, P. and Dabas, C., 2019. Bitcoin Price Forecasting using LSTM and 10-Fold Cross validation. Paper presented at the 2019 International Conference on Signal Processing and Communication (ICSC), NOIDA, India, March 7–9.
[37] (Tobin and Kashef 2020) Tobin, T. and Kashef, R., 2020. Efficient Prediction of Gold Prices Using Hybrid Deep Learning. In: Image Analysis and Recognition. ICIAR 2020. Lecture Notes in Computer Science. Edited by A. Campilho, F. Karray and Z. Wang. Cham: Springer, vol. 12132. DOI: 10.1007/978-3-030-50516-5_11.
[38] (United States Securities and Exchange Commission 2017) United States Securities and Exchange Commission, 2017. Annual Report Archive. Available at: http://www.annualreports.com/HostedData/AnnualReportArchive/m/AMEX_MGT_2017.pdf.
[39] (Wang et al. 2017) Wang, S., Ye, S. and Li, X., 2017. The impact of real effective exchange rate volatility on economic growth in the process of renminbi internationalization: An empirical study based on VAR model. Paper presented at the 2017 4th International Conference on Industrial Economics System and Industrial Security Engineering (IEIS), Kyoto, Japan, July 24–27.
[40] (Wu et al. 2018) Wu, C.-H., Ma, Y.-F., Lu, R.-S. and Lu, Y.-F., 2018. A New Forecasting Framework for Bitcoin Price with LSTM. Paper presented at the 2018 IEEE International Conference on Data Mining Workshops (ICDMW), Singapore, November 17–20.

References for "Predicting the Demand in Bitcoin Using Data Charts: A Convolutional Neural Networks Prediction Model":

[1] Srivastava, S., 2017. Deep Learning in Finance. Towards Data Science.

[2] Razavian, S., Azizpour, H., Sullivan, J. and Carlsson, S., 2014. CNN Features Off-the-Shelf: An Astounding Baseline for Recognition. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops.

[3] Abdel-Hamid, O., Deng, L. and Yu, D., 2013. Exploring Convolutional Neural Network Structures and Optimization Techniques for Speech Recognition. In: INTERSPEECH, Lyon.

[4] Simonyan, K. and Zisserman, A., 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. In: ICLR.

[5] Lin, M., Chen, Q. and Yan, S., 2014. Network In Network. CORR Journal.

[6] Fischer, M.M., 2006. Computational Neural Networks — Tools for Spatial Data Analysis. Spatial Analysis and GeoComputation, pp. 79-102.

[7] Silva, E., Castilho, D. and Pereira, A., 2014. A neural network-based approach to support the market making strategies in high-frequency trading. In: IJCNN, 2014 International Joint Conference, pp. 845-852.

[8] Wang, Z. and Oates, T., 2015. Encoding Time Series as Images for Visual Inspection and Classification Using Tiled Convolutional Neural Networks. Trajectory-Based Behavior Analytics: Papers from the 2015 AAAI Workshop.

[9] He, K., Zhang, X., Ren, S. and Sun, J., 2016. Deep Residual Learning for Image Recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, pp. 770-778.

[10] Loshchilov, I. and Hutter, F., 2017. SGDR: Stochastic Gradient Descent with Warm Restarts. In: ICLR 2017, 5th International Conference on Learning Representations.

[11] Krizhevsky, A., Sutskever, I. and Hinton, G., 2012. ImageNet Classification with Deep Convolutional Neural Networks. Journal of Neural Information Processing Systems.

[12] Smith, L.N., 2017. Cyclical Learning Rates for Training Neural Networks. In: IEEE Conference on Applications of Computer Vision (WACV), pp. 464-472.

[13] Russakosky, O. and Deng, J., 2015. ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision (IJCV), 115(3).

[14] Tiong, L.C., Ngo, D.C. and Lee, Y., 2014. Stock Price Prediction Model using Candlestick Pattern Feature.

[15] Rémy, P., 2019. When Bitcoin meets Artificial Intelligence. Available at: https://github.com/philipperemy/deep-learning-Bitcoin .
[16] Tan, X. and Kashef, R., 2019. Predicting the closing price of cryptocurrencies: a comparative study. In: Proceedings of the Second International Conference on Data Science, E-Learning and Information Systems (DATA '19). Association for Computing Machinery, New York, NY, USA. DOI: https://doi.org/10.1145/3368691.3368728.
[17] Shah, D., 2014. Bayesian regression and Bitcoin . In: Fifty-second Annual Allerton Conference, USA.
[18] Taylor, S. and Letham, B., 2017. Prophet: forecasting at scale. Facebook research.
[19] Hayashi, H., 2017. Is Prophet Really Better than ARIMA for Forecasting Time Series Data? Exploratory.
[20] Yermack, D., 2015. Is Bitcoin a Real Currency? An Economic Appraisal. Available on: [website link if provided].
[21] Jiang, Z. and L., J., 2017. Cryptocurrency Portfolio Management with Deep Reinforcement Learning. Intellisys Conference, London, UK.
[22] PyTorch, n.d. Available at: https://pytorch.org/.
[23] Zhong, X. and Enke, D., 2017. Forecasting daily stock market return using dimensionality reduction. Expert Systems with Applications, pp. 126-139.
[24] Madan, I. and S., S., 2017. Automated Bitcoin Trading via Machine Learning Algorithms. Semantic Scholar.

References for "**Forecasting the early market movement in Bitcoin using Twitter's sentiment analysis**":

[1] Tan, X. & Kashef, R., 2019. Predicting the closing price of cryptocurrencies: a comparative study. In: Proceedings of the Second International Conference on Data Science, E-Learning and Information Systems, December 2019, pp. 1-5.

[2] Ibrahim, A., Kashef, R., Li, M., Valencia, E. & Huang, E., 2020. Bitcoin Network Mechanics: Forecasting the BTC Closing Price Using Vector Auto-Regression Models Based on Endogenous and Exogenous Feature Variables. Journal of Risk and Financial Management, 13(9), pp.189.

[3] Ibrahim, A., Kashef, R. & Corrigan, L. Predicting market movement direction for Bitcoin : A comparison of time series modeling methods. Computers & Electrical Engineering, 89, p.106905.

[4] Tobin, T. & Kashef, R., 2020. Efficient Prediction of Gold Prices Using Hybrid Deep Learning. In: International Conference on Image Analysis and Recognition, June 2020, pp. 118-129. Springer, Cham.

[5] Ibrahim, A.F., Corrigan, L. & Kashef, R., 2020. Predicting the Demand in Bitcoin Using Data Charts: A Convolutional Neural Networks Prediction Model. In: 2020 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE), London, ON, Canada, 2020, pp. 1-4. doi: 10.1109/CCECE47787.2020.9255711.

[6] Kashef, R. & Kamel, M.S., 2010. Cooperative clustering. Pattern Recognition, 43(6), pp.2315-2329.

[7] Kashef, R., 2008. Cooperative clustering model and its applications.

[8] Kashef, R., 2020. A boosted SVM classifier trained by incremental learning and decremental unlearning approach. Expert Systems with Applications. doi:10.1016/j.eswa.2020.114154.

[9] Kraaijeveld, O. & Smedt, J.D., 2020. The predictive power of public Twitter sentiment for forecasting cryptocurrency prices. Journal of International Financial Markets, Institutions, and Money, p.101188, Mar. 2020.

[10] Li, T.R. et al., 2019. Sentiment-Based Prediction of Alternative Cryptocurrency Price Fluctuations Using Gradient Boosting Tree Model. Frontiers in Physics, 7, Oct. 2019.

[11] Mohapatra, S., Ahmed, N. & Alencar, P., 2020. KryptoOracle: A Real-Time Cryptocurrency Price Prediction Platform Using Twitter Sentiments. arXiv:2003.04967 [cs.CL], Feb. 2020.

[12] Kaplan, C., Aslan, C. & Bulbul, A., 2018. Cryptocurrency Word-of-Mouth Analysis via Twitter. ResearchGate. [Online]. Available:

https://www.researchgate.net/publication/327988035_Cryptocurrency_Word-of-Mouth_Analysis_viaTwitter.
[13] Jain, A. et al., 2018. Forecasting Price of Cryptocurrencies Using Tweets Sentiment Analysis. In: 2018 Eleventh International Conference on Contemporary Computing (IC3), Noida, 2018, pp. 1-7. doi: 10.1109/IC3.2018.8530659.
[14] Symeonidis, S., Effrosynidis, D. & Arampatzis, A., 2018. A comparative evaluation of pre-processing techniques and their interactions for twitter sentiment analysis. Expert Systems with Applications, 110, pp.298-310, Nov. 2018.
[15] Sailunaz, K. & Alhajj, R., 2019. Emotion and sentiment analysis from Twitter text. Journal of Computational Science, 36, Sep. 2019. doi: 10.1016/j.jocs.2019.05.009.
[16] Rosen, A., 2017. Tweeting Made Easier. Twitter. [Online]. Available: https://blog.twitter.com/en_us/topics/product/2017/tweetingmadeeasier.html. [Accessed: 24-Jul-2020].
[17] Lyu, H. et al., 2020. Sense and Sensibility: Characterizing Social Media Users Regarding the Use of Controversial Terms for COVID-19. IEEE Transactions on Big Data. doi: 10.1109/TBDATA.2020.2996401.
[18] Wang, L. et al., 2020. Prediction of Type 2 Diabetes Risk and Its Effect Evaluation Based on the XGBoost Model. In: Healthcare, 8(3), p.247. Multidisciplinary Digital Publishing Institute, September 2020.
[19] Kashef, R. & Kamel, M.S., 2009. Enhanced bisecting k-means clustering using intermediate cooperation. Pattern Recognition, 42(11), pp.2557-2569.
[20] Kashef, R. & Kamel, M., 2006. Distributed cooperative hard-fuzzy document clustering. In: Proceedings of the Annual Scientific Conference of the LORNET Research Network, November 2006.
[21] Kashef, R. & Kamel, M.S., 2007. Hard-fuzzy clustering: a cooperative approach. In: 2007 IEEE International Conference on Systems, Man and Cybernetics. pp.425-430. IEEE, October 2007.
[22] Yeh, T.Y. & Kashef, R., 2020. Trust-Based Collaborative Filtering Recommendation Systems on the Blockchain. Advances in Internet of Things, 10(4), pp.37-56.
[23] Hutto, C.J., 2014. VADER-Sentiment-Analysis. GitHub. [Online].
Available: https://github.com/cjhutto/vaderSentiment. [Accessed: 24-Jul-2020].

[24] Hutto, C.J. & Gilbert, E., 2014. VADER: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text. Presented at the Eighth International AAAI Conference on Weblogs and Social Media, May 2014. [Online].
Available: https://www.aaai.org/ocs/index.php/ICWSM/ICWSM14/paper/view/8109.
[25] Hass, G., Simon, P. & Kashef, R., 2020. Business Applications for Current Developments in Big Data Clustering: An Overview. In: 2020 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), Singapore, Singapore, 2020. doi: 10.1109/IEEM45057.2020.9309941.
[26] Close, L. & Kashef, R., 2020. Combining Artificial Immune System and Clustering Analysis: A Stock Market Anomaly Detection Model. Journal of Intelligent Learning Systems and Applications, 12, pp.83-108. doi: 10.4236/jilsa.2020.124005.
[27] Fayyaz, Z. et al., 2020. Recommendation Systems: Algorithms, Challenges, Metrics, and Business Opportunities. Applied Sciences, 10(21), p.7748.
[28] Kashef, R.F., 2018. Ensemble-Based Anomaly Detetction using Cooperative Learning. In: KDD 2017 Workshop on Anomaly Detection in Finance, January 2018, pp. 43-55. PMLR.
[29] Ebrahimian, M. & Kashef, R., 2020. Efficient Detection of Shilling's Attacks in Collaborative Filtering Recommendation Systems Using Deep Learning Models. In: 2020 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), Singapore, Singapore, 2020. doi: 10.1109/IEEM45057.2020.9309965.
[30] Ebrahimian, M. & Kashef, R., 2020. Detecting Shilling Attacks Using Hybrid Deep Learning Models. Symmetry, 12(11). doi:10.3390/sym12111805.
[31] Kashef, R., 2020. Enhancing the Role of Large-Scale Recommendation Systems in the IoT Context. IEEE Access, 8, pp.178248-178257.
[32] Nawara, D. & Kashef, R., 2020. IoT-based Recommendation Systems–An Overview. In: 2020 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), September 2020, pp. 1-7. IEEE.
[33] Kashef, R. & Niranjan, A., 2017. Handling Large-Scale Data Using Two-Tier Hierarchical Super-Peer P2P Network. In: Proceedings of the International Conference on Big Data and Internet of Thing, December 2017, pp. 52-56.
[34] Li, M., Kashef, R. & Ibrahim, A., 2020. Multi-Level Clustering-Based Outlier's Detection (MCOD) Using Self-Organizing Maps. Big Data and Cognitive Computing, 4(4). doi:10.3390/bdcc4040024.

[35] Pano, T. & Kashef, R., 2020. A Corpus of BTC Tweets in the Era of COVID-19. In: 2020 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), September 2020, pp. 1-4. IEEE.
[36] Pano, T. & Kashef, R., 2020. A Complete VADER-Based Sentiment Analysis of Bitcoin (BTC) Tweets during the Era of COVID-19. Big Data and Cognitive Computing, 4(4), p.33.

References for "Analyzing BTC's Trend During COVID-19 Using A Sentiment Consensus Clustering (SCC)":

[1] Tan, X. & Kashef, R., 2019. Predicting the closing price of cryptocurrencies: a comparative study. In: Proceedings of the Second International Conference on Data Science, E-Learning and Information Systems, December 2019, pp. 1-5.

[2] Ibrahim, A. et al., 2020. Bitcoin Network Mechanics: Forecasting the BTC Closing Price Using Vector Auto-Regression Models Based on Endogenous and Exogenous Feature Variables. Journal of Risk and Financial Management, 13(9), pp.189.

[3] Ibrahim, A., Kashef, R. & Corrigan, L. Predicting market movement direction for Bitcoin : A comparison of time series modeling methods. Computers & Electrical Engineering, 89, p.106905.

[4] Tobin, T. & Kashef, R., 2020. Efficient Prediction of Gold Prices Using Hybrid Deep Learning. In: International Conference on Image Analysis and Recognition, June 2020, pp. 118-129. Springer, Cham.

[5] Ibrahim, A.F., Corrigan, L. & Kashef, R., 2020. Predicting the Demand in Bitcoin Using Data Charts: A Convolutional Neural Networks Prediction Model. In: 2020 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE), London, ON, Canada, 2020, pp. 1-4. doi: 10.1109/CCECE47787.2020.9255711.

[6] Kashef, R. & Kamel, M.S., 2010. Cooperative clustering. Pattern Recognition, 43(6), pp.2315-2329.

[7] Kashef, R., 2008. Cooperative clustering model and its applications.

[8] Kashef, R., 2020. A boosted SVM classifier trained by incremental learning and decremental unlearning approach. Expert Systems with Applications. doi:10.1016/j.eswa.2020.114154.

[9] Kraaijeveld, O. & Smedt, J.D., 2020. The predictive power of public Twitter sentiment for forecasting cryptocurrency prices. Journal of International Financial Markets, Institutions, and Money, p.101188, Mar. 2020.

[10] Li, T.R. et al., 2019. Sentiment-Based Prediction of Alternative Cryptocurrency Price Fluctuations Using Gradient Boosting Tree Model. Frontiers in Physics, 7, Oct. 2019.

[11] Mohapatra, S., Ahmed, N. & Alencar, P., 2020. KryptoOracle: A Real-Time Cryptocurrency Price Prediction Platform Using Twitter Sentiments. arXiv:2003.04967 [cs.CL], Feb. 2020.

[12] Kaplan, C., Aslan, C. & Bulbul, A., 2018. Cryptocurrency Word-of-Mouth Analysis via Twitter. ResearchGate. [Online]. Available:

https://www.researchgate.net/publication/327988035_Cryptocurrency_Word-of-Mouth_Analysis_viaTwitter.
[13] Jain, A. et al., 2018. Forecasting Price of Cryptocurrencies Using Tweets Sentiment Analysis. In: 2018 Eleventh International Conference on Contemporary Computing (IC3), Noida, 2018, pp. 1-7. doi: 10.1109/IC3.2018.8530659.
[14] Symeonidis, S., Effrosynidis, D. & Arampatzis, A., 2018. A comparative evaluation of pre-processing techniques and their interactions for twitter sentiment analysis. Expert Systems with Applications, 110, pp.298-310, Nov. 2018.
[15] Sailunaz, K. & Alhajj, R., 2019. Emotion and sentiment analysis from Twitter text. Journal of Computational Science, 36, Sep. 2019. doi: 10.1016/j.jocs.2019.05.009.
[16] Rosen, A., 2017. Tweeting Made Easier. Twitter. [Online]. Available: https://blog.twitter.com/en_us/topics/product/2017/tweetingmadeeasier. [Accessed: 24-Jul-2020].
[17] Lyu, H. et al., 2020. Sense and Sensibility: Characterizing Social Media Users Regarding the Use of Controversial Terms for COVID-19. IEEE Transactions on Big Data. doi: 10.1109/TBDATA.2020.2996401.
[18] Wang, L. et al., 2020. Prediction of Type 2 Diabetes Risk and Its Effect Evaluation Based on the XGBoost Model. In: Healthcare, 8(3), p.247. Multidisciplinary Digital Publishing Institute, September 2020.
[19] Baralis, E. et al., 2013. Analysis of Twitter Data Using a Multiple-level Clustering Strategy. In: Model and Data Engineering. MEDI 2013. Lecture Notes in Computer Science, vol 8216. Springer, Berlin, Heidelberg. doi:10.1007/978-3-642-41366-7_2.
[20] Kim, Y.H. et al., 2013. Two Applications of Clustering Techniques to Twitter: Community Detection and Issue Extraction. Discrete Dynamics in Nature and Society. doi:10.1155/2013/903765.
[21] Alnajran, N. et al., 2017. Cluster Analysis of Twitter Data: A Review of Algorithms. In: Proceedings of the 9th International Conference on Agents and Artificial Intelligence - Volume 1: ICAART. pp.239-249. doi: 10.5220/0006202802390249.
[22] Godfrey, D. et al., 2014. A case study in text mining: Interpreting twitter data from world cup tweets. arXiv preprint arXiv:1408.5427.
[23] Kashef, R. & Kamel, M.S., 2009. Enhanced bisecting k-means clustering using intermediate cooperation. Pattern Recognition, 42(11), pp.2557-2569.

[24] Kashef, R. & Kamel, M., 2006. Distributed Consensus hard-fuzzy document clustering. In: Proceedings of the Annual Scientific Conference of the LORNET Research Network, November 2006.
[25] Kashef, R. & Kamel, M.S., 2007. Hard-fuzzy clustering: a Consensus approach. In: 2007 IEEE International Conference on Systems, Man and Cybernetics. pp.425-430. IEEE, October 2007.
[26] Yeh, T.Y. & Kashef, R., 2020. Trust-Based Collaborative Filtering Recommendation Systems on the Blockchain. Advances in Internet of Things, 10(4), pp.37-56.
[27] Hutto, C.J., 2014. VADER-Sentiment-Analysis. GitHub. [Online]. Available: https://github.com/cjhutto/vaderSentiment. [Accessed: 24-Jul-2020].
[28] Hutto, C.J. & Gilbert, E., 2014. VADER: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text. Presented at the Eighth International AAAI Conference on Weblogs and Social Media, May 2014. [Online]. Available: https://www.aaai.org/ocs/index.php/ICWSM/ICWSM14/paper/view/8109.
[29] Hass, G., Simon, P. & Kashef, R., 2020. Business Applications for Current Developments in Big Data Clustering: An Overview. In: 2020 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), Singapore, Singapore, 2020. doi: 10.1109/IEEM45057.2020.9309941.
[30] Close, L. & Kashef, R., 2020. Combining Artificial Immune System and Clustering Analysis: A Stock Market Anomaly Detection Model. Journal of Intelligent Learning Systems and Applications, 12, pp.83-108. doi: 10.4236/jilsa.2020.124005.
[31] Fayyaz, Z. et al., 2020. Recommendation Systems: Algorithms, Challenges, Metrics, and Business Opportunities. Applied Sciences, 10(21), p.7748.
[32] Kashef, R.F., 2018. Ensemble-Based Anomaly Detetction using Consensus Learning. In: KDD 2017 Workshop on Anomaly Detection in Finance, January 2018, pp. 43-55. PMLR.
[33] Ebrahimian, M. & Kashef, R., 2020. Efficient Detection of Shilling's Attacks in Collaborative Filtering Recommendation Systems Using Deep Learning Models. In: 2020 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), Singapore, Singapore, 2020. doi: 10.1109/IEEM45057.2020.9309965.
[34] Ebrahimian, M. & Kashef, R., 2020. Detecting Shilling Attacks Using Hybrid Deep Learning Models. Symmetry, 12(11). doi:10.3390/sym12111805.
[35] Kashef, R., 2020. Enhancing the Role of Large-Scale Recommendation Systems in the IoT Context. IEEE Access, 8, pp.178248-178257.

[36] Nawara, D. & Kashef, R., 2020. IoT-based Recommendation Systems–An Overview. In: 2020 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), September 2020, pp. 1-7. IEEE.
[37] Kashef, R. & Niranjan, A., 2017. Handling Large-Scale Data Using Two-Tier Hierarchical Super-Peer P2P Network. In: Proceedings of the International Conference on Big Data and Internet of Thing, December 2017, pp. 52-56.
[38] Li, M., Kashef, R. & Ibrahim, A., 2020. Multi-Level Clustering-Based Outlier's Detection (MCOD) Using Self-Organizing Maps. Big Data and Cognitive Computing, 4(4). doi:10.3390/bdcc4040024.
[39] Pano, T. & Kashef, R., 2020. A Corpus of BTC Tweets in the Era of COVID-19. In: 2020 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), September 2020, pp. 1-4. IEEE.
[40] Pano, T. & Kashef, R., 2020. A Complete VADER-Based Sentiment Analysis of Bitcoin (BTC) Tweets during the Era of COVID-19. Big Data and Cognitive Computing, 4(4), p.33.