Cryptocurrency Trading Program Using Machine Learning

by

Dante Knight, Jack Gerulskis, Sean Dandeneau

A Major Qualifying Project

Submitted to the Faculty

of the

WORCESTER POLYTECHNIC INSTITUTE

In partial fulfillment of the requirements for the

Degree of Bachelor of Science

in Computer Science

by

May 2021

APPROVED:

Smith, Therese

Abstract

Cryptocurrency is a digital asset that has been historically volatile. This volatility allows traders to capitalize on short term price movement. More specifically, cryptocurrency is vulnerable to "epidemic-like price bubbles" from social media factors compared to traditional assets (Phillips, 2018)[11]. Social media's influence on the price of cryptocurrency gives traders a unique opportunity for predicting price movements. Traditionally, traders have used technical analysis to predict the best opportunities to buy and sell, but sentiment analysis of posts on social media can help improve their accuracy. While humans are capable of manually conducting technical analysis, it is near impossible for them to understand the trends and consensus of an asset from posts on social media. A computer can conduct both technical and sentiment analysis more efficiently and use them as indicators to make accurate predictions on future price movements. The goal of this project is to create an automated trading program that uses technical and sentiment analysis as inputs for a machine learning model which can predict profitable opportunities to buy and sell cryptocurrency.

Contents

List of Figures

Chapter 1

Introduction

Machine learning trading algorithms have had success in the stock markets for some time now (Chang, Lui, 2009)[2]. More recently, they also have outperformed traditional buy-and-hold strategies for cryptocurrencies (Jiang, Liang, 2017)[10]. The successes of machine learning models trading cryptocurrency has inspired more research in the field. Prior research in the field has shown sentiment analysis from social media sites is a useful indicator for machine learning models. (Phillips, 2018)[11]. Other researchers have successfully created multiple profitable models using technical analysis. This report intends to build on the previous research by finding new ways to conduct meaningful sentiment analysis and also by creating an ensemble model to increase accuracy and effectiveness.

Different artificial neural networks have shown to outperform traditional statistical models in 72% of cases, but the highly volatile nature of the cryptocurrency market makes it difficult for a single model to be effective in every instance (Chang et al. 2017)[3]. In the other 28% of situations, statistical models outperform machine learning (Chang et al. 2017)[3]. The other challenge with creating models that uses sentiment analysis is accurately converting posts on social media to data the models can understand. A few of the challenges with this conversion is understanding slang, sarcasm, and irony. Selecting an optimal model depending on the volatility of the market and properly analysing sentiment will be critical to creating an effective trading program.

As mentioned, prior research on trading cryptocurrency using machine learning has already been done. Hegazy and Mumford created a model using a supervised learning strategy that had a 57% accuracy in predicting price fluctuations (Hegazy, Mumford, 2016). Jiang and Liang used deep reinforcement learning to increase their initial investment by 1000% (Jiang, Liang, 2017)[10]. Shah and Zhang achieved a 200% return on their inital investment in 2 months by using Bayesian regression(Shah, Zhang, 2014)[13]. Fischer and others created a profitable model using an arbitrage approach (Fischer et al, 2019)[4]. Stenqvist and Lonno used deep learning algorithms to analyze 2.27 million tweets to predict Bitcoin price fluctuations with a 79% accuracy (Stenqvist, Lonno, 2017)[14].

In order to differentiate ourselves from the work that has already been done, we have identified two gaps in the research that we will attempt to fill. First, we will implement an ensemble learning model that can operate accurately regardless of the volatility. This will improve the reliability of the system because the outputs of the models in the ensemble system will reduce the effect of bad decisions from varying volatility. It is important to note that the best model in the group might still outperform the ensemble in the short term, but the ensemble will outperform the individual models in the long term (Fumera, Roli, 2005)[5]. The second gap is using the impressions metric when converting posts from social media to machine readable data. This should enable the model to differentiate between posts that have lots of interactions with people compared to tweets that nobody interacts with. This is something that we didn't see in our prior research and will help identify posts that have a stronger influence. By implementing these solutions to on top of successful models that have already been created, we should be able to enhance their performance.

The goal of this project is to create an automated trading program that uses

technical and sentiment analysis as inputs for a machine learning model which can predict profitable opportunities to buy and sell cryptocurrency. Our objectives to achieve this goal were to collect data from Twitter, conduct accurate sentiment analysis, create profitable individual models, and lastly to create an ensemble model.

This report begins with an overview of cryptocurrency trading strategies and machine learning trading. We then discuss sentiment analysis and the key techniques involved with properly performing this analysis. Following the background, our report details how we gathered our data and created our ensemble model. We then finish by discussing the results and implications of our study. This research benefits cryptocurrency traders and prior studies, who can use our research to improve their trading decisions.

Chapter 2

Background

2.0.1 Technical Analysis Trading Strategies

The most prevelant analysis for cryptocurrency trading is Technical Analysis. Technical Analysis is, "the study, practice, and analysis of chart patterns, indicators and oscillators, and the candlesticks themselves that make up price charts of assets (https://primexbt.com/blog/cryptocurrency-trading-strategies/)". This form of analysis will identify factors such as Relative Strength Index (RSI) and moving averages that can give a trader an edge in the market. Understanding these strategies is critical before using them as indicators for our model.

RSI is the most common technical cryptocurrency trading strategy. RSI identifies when an asset is being over or under valued and bought in the market. It watches the price movement of a tradable asset set to a scale of 0-100, with two triggers at an upper and lower bound. As a baseline, these triggers are set at 70 and 30. When the stock reaches or goes above 70, RSI tells us that this stock is being overbought and overvalued, and history tells us this stock will revert back towards a lower price. On the other hand, when a stock drops below 30, it is being oversold and undervalued with a high likelihood of trending back to the mean of 50. An example of RSI is depicted in the figure below.

The moving average trading strategy identifies the moving average of the stock and, like RSI, has two triggers to identify when to buy or sell. Shown in the graph below, when the price crosses the moving average line it tells us when to buy or short

Figure 2.1: RSI

the stock. Moving averages are help identify trends by comparing the averages from a long and short time frame. Time frames for moving averages commonly follow the Fibonacci series. For example a short term moving average may be 5 days and that would be compared with a longer moving average like 13 days.

Figure 2.2: Moving Average

2.0.2 Machine Learning Trading Strategies

To help remove the human element out of trading, we look towards machine learning strategies. These strategies remove human emotion and bias to make a calculated decision. There are numerous different machine learning trading methods, so we will just cover the top three.

The first is graph neural networks. Graph neural networks (GNNs) are a deep learning discipline that focuses on models that operate efficiently on graph data structures (https://www.coindesk.com/five-machine-learning-methods-crypto-traders-should-know-about). Though this is a relatively new area of deep learning, it is widely used for numerous applications in companies like Uber, Google, Microsoft and DeepMind. GNNs use "a graph as input representing the flows in and out of exchanges and infer relevant knowledge relevant to its impact on price (https://www.coindesk.com/five-machine-learning-methods-crypto-traders-should-know-about)." GNNs open the potential for new quant methods based on blockchain datasets.

The second machine learning strategy is generative models. Generative models look at historical data to develop their own synthetic data that closer mirrors the distribution of a training data set. When we combine the real dataset with the synthetic one that was generated, we then have a large enough dataset to train a complex deep learning model. One technique used in generative models is generative adversarial neural networks (GANs). Although not directly related to cryptocurrency trading, GANs have proven very successful in image classification. The ability to memorize previous trends and identify them happening in real time separate machine learning trading strategies from technical strategies.

Semi-supervised learning is the last machine learning trading strategy we will talk

about. Semi-supervised learning focuses on creating models that learning with small labeled data sets and a large amount of unlabeled data. "Semi-supervised learning is analogous to a teacher presenting a few concepts to a group of students and leaves the other concepts to homework and self-study (https://www.coindesk.com/five-machine-learning-methods-crypto-traders-should-know-about)." A semi-supervised learning model will identify important features in the labeled data and use that information to incorporate the unlabeled data into the training.

Machine learning trading strategies get much more complex that technical strategies but can be much more accurate with far less risk. All of these models will be able to automatically calculate and interpret technical indicators mentioned in the previous section to make predictions.

2.0.3 Comparing Trading Strategies

As discussed above, there are two groups of trading strategies that we will be looking at: human designed and machine learned. The first group is the type of strategy that can be understood well because they are the creation of a human who designed it with a certain theory in mind. A typical human designed trading strategy will issue orders in cases like when a certain price point is reached or a technical indicator crosses some value or some combination of one or more of both. One can imagine a strategy which uses RSI, a common technical indicator which indicates positive and negative momentum on a price series from 0-100 with 100 being the highest positive momentum and 0 being the highest negative momentum. A common way to use RSI is to have two thresholds near the extremes of the spectrum like 30 and 70 or 18 and 86. When RSI crosses the chosen threshold in this strategy, at the higher threshold it would sell the asset and at the lower it would buy. This is an attempt to buy when the price is moving most down and

sell when price is moving most up. A more complex example of this can be seen in the table below which has columns for buy and sell signals which are determined by the functions shown in each row of the two columns. The strategy described can be understood and rationalized by the people who implement it either through trading manually or by automatically.

Rule Type	Strategy Type	Buy rule: ($Signal_{t+1} = 1$ when)	Sell rule: ($Signal_{t+1} = 0$ or -1 when)	Parameterization settings	Total rules
F	TS	$L_t > (1+x) \underset{0<k<t}{\text{argmax}} \{L_k \mid L_k < L_{k-1}\}$	$L_t < (1-x) \underset{0<k<t}{\text{argmax}} \{L_k \mid L_k \rangle L_{k-1}\}$	$x \in [1\%: 1\%: 50\%]$;	50
TRB	TS	$L_t > \underset{1\leq k<n_1}{\text{argmax}} \{L_{t-k}\}$	$L_t < \underset{1\leq k<n_1}{\text{argmin}} \{L_{t-k}\}$	$n_1, n_2 \in [2: 3: 143]$;	2304
MA	TS	$MA_t(n_1) > MA_t(n_2)$	$MA_t(n_1) \leq MA_t(n_2)$	$n_1, n_2 \in [2: 3: 143]$; $n_1 < n_2$;	1128
MACD	TS	$MACDp_t(n_1,n_2) > x$	$MACDp_t(n_1,n_2) \leq x$	$n_1, n_2 \in [2: 3: 143]$; $n_1 < n_2$; $x \in [-16\%: 2\%: 16\%]$; $x \neq 0$	18,048
	TR	$MACD_t(n_1,n_2) > S_t(n_3)$	$MACD_t(n_1,n_2) \leq S_t(n_3)$	$n_1, n_2 \in [2: 3: 143]$; $n_1 < n_2$; $n_3 \in [2: 3: 14]$;	5640
	TR	$MACD_t(n_1,n_2) > D_t(n_3)$	$MACD_t(n_1,n_2) \leq D_t(n_3)$	$n_1, n_2 \in [2: 3: 143]$; $n_1 < n_2$;	2256

Figure 2.3: Human Designed Strategy

The same cannot be said for trading strategies that have been learned using machine learning. While it may be possible to guess how a model with a simple architecture works, as the complexity increases, the guess becomes more rough and less reflective of the actual model. This isn't to say that machine learning isn't

desirable for a trading strategy because it certainly is. Only that one's understanding of the connection between the input and the output becomes less accurate as the model becomes more complex. The example in the table below is set up similar to the human designed strategy above but the buy and sell signal columns are missing because the nice functions that we had before can't be defined as easily now (Anghel, 2020)[1]. The equivalent to the buy and sell signals in this example would be the last dense layer in each of the networks that outputs only one value at the end which would be interpreted as buy sell or hold. In the example before, if there was a buy signal, we would be able to point to the RSI being below the threshold and say with certainty that that is what caused the buy signal unlike with the machine learning example.

2.0.4 Comparing Tradable Assets

The correlation between price of bitcoin and the many other cryptocurrencies has been studied in the past and has increased since bitcoin's all time high price near the end of 2017. This is an important consideration because it has implications on the prospect of diversifying an all cryptocurrency portfolio, which could be a strategy we consider. The findings tell us that all of the cryptocurrencies prices are correlated which makes diversification less effective at spreading out risk over multiple independent investments (Ferreira and Pereira, 2019)[**ferreira˙contagion˙nodate**]. This is because when the assets are correlated, investing in two different assets instead of one isn't going to shield you from the price fluctuation of one because a similar fluctuation will most likely be occurring in the other asset. An example can be seen below showing the last year of prices of bitcoin and ripple (XRP).

The figures have green and red boxes signifying upwards and downwards price trends respectively. The figure showing the price of XRP has additional grey vertical

RNN	keras	Layer (type)	Output Shape	Param #	
		simple_rnn (SimpleRNN)	*(None, None, 4)*	*36*	
		simple_rnn_1 (SimpleRNN)	*(None, 4)*	*36*	
		dense_5 (Dense)	*(None, 1)*	*5*	
WDNN	keras	Layer (type)	Output Shape	Param #	Connected to
		input_1 (InputLayer)	*[(None, 34)]*	*0*	
		dropout (Dropout)	*(None, 34)*	*0*	*input_1[0][0]*
		dense (Dense)	*(None, 32)*	*1120*	*dropout[0][0]*
		dropout_1 (Dropout)	*(None, 32)*	*0*	*dense[0][0]*
		dense_1 (Dense)	*(None, 32)*	*1056*	*dropout_1[0][0]*
		dropout_2 (Dropout)	*(None, 32)*	*0*	*dense_1[0][0]*
		dense_2 (Dense)	*(None, 32)*	*1056*	*dropout_2[0][0]*
		dropout_3 (Dropout)	*(None, 32)*	*0*	*dense_2[0][0]*
		dense_3 (Dense)	*(None, 32)*	*1056*	*dropout_3[0][0]*
		concatenate (Concatenate)	*(None, 66)*	*0*	*input_1[0][0]*
					dense_3[0][0]
		dense_4 (Dense)	*(None, 1)*	*67*	*concatenate[0][0]*

Figure 2.4: Machine Learning Strategy

lines which represent the starts and ends of the green and red boxes for bitcoin. It can be seen by the eye that these assets have some serious price correlation but that being said not everything is the same. Another thing to note is that certain cryptocurrencies seem to have higher volatility meaning that they are prone to fluctuations of greater percentages than bitcoin. This can be seen in the figures above. The percentage changes for the boxes in the XRP graph are generally greater than those of bitcoin. This is an important consideration because it can help us to

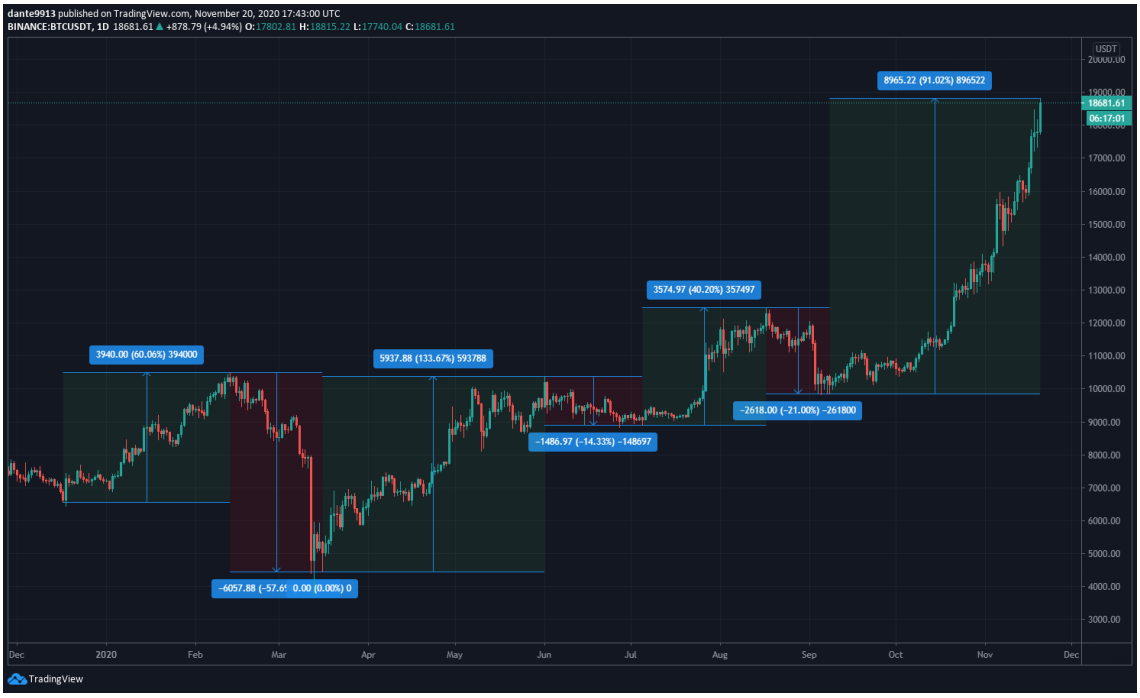

Figure 2.5: Bitcoin Price Graph

gauge how much risk a strategy is taking on when it invests in each cryptocurrency and knowing that allows the strategy to regulate its own risk. This can most likely be attributed to the lower market cap of XRP in comparison to Bitcoin. In general as market cap increases, price fluctuations become lower in percentage change.

2.0.5 Cryptocurrency Markets vs Traditional Markets

A common example of a traditional market would be US equities. In this market, brokers use to make their money by charging large fees. Recently, brokers like Robinhood have popularized the concept of $0 commission trades. Now most big firms offer the same deals to investors. The dropping fees forced brokers to find more ways to generate revenue. Brokers often sell orders that don't specify a specific exchange to execute the trade on to market makers. The market makers are given the opportunity to trade with individual investors the lack the advantages high

Figure 2.6: XRP Price Graph

frequency trading firms have. This creates an issue for individual traders trying to profitable trade intra-daily because most of the time they will be trading with market makers or the high frequency trading firms on the exchanges. Market makers aren't that interested in institutional investors because they have a better advantage than individuals. These markets are highly efficient, have low spreads in the orders, and news propagates instantly.

In cryptocurrency markets, the biggest difference is that there are no longer brokers. Instead of brokers managing orders, the investors handle the orders management themselves in a peer to peer fashion. The settlement is then handled without intervention from the National Securities Clearing Corporation and is handled by the exchange. If the investor is trading on a decentralized exchange the trade is then conducted smart contracts. Unlike centralized exchanges where buy and sell orders are paired by an order book, decentralized exchanges operate by matching the in-

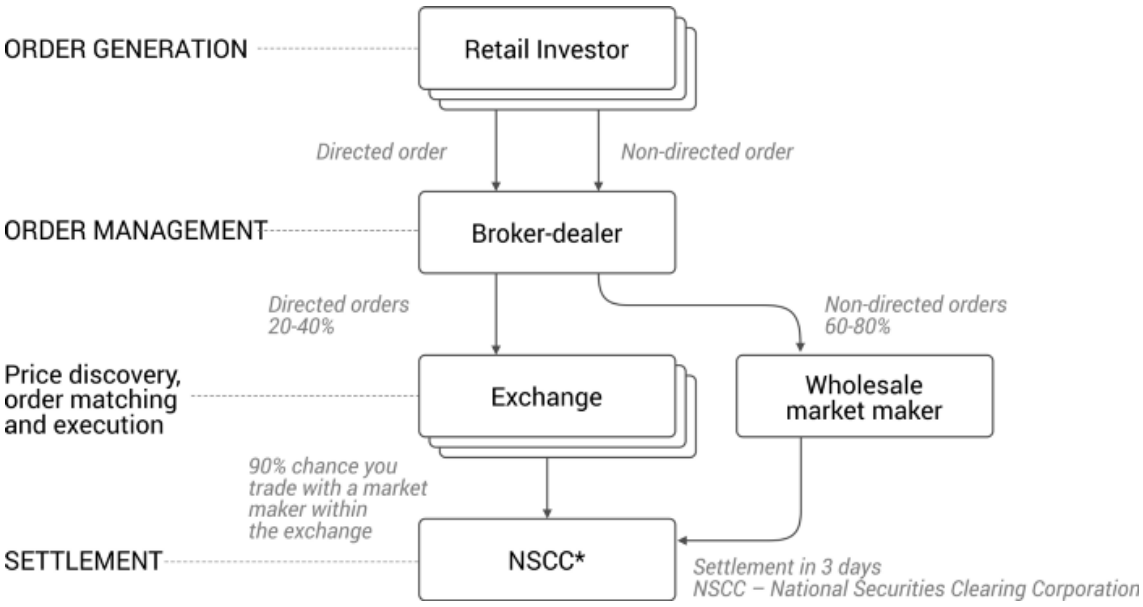

Figure 2.7: US Equities Order Flow

dividuals that want to execute the buy and sell orders and allow them to exchange their assets. By removing the middleman between the investor and the exchanges, the high frequency trading firms loose their critical advantage between directed and non directed orders. Decentralized exchanges make it even harder for high frequency traders because they operate using smart contracts which lowers transaction speeds and have higher fees. This is constantly changing as the technology improves but for now those factors and the lower liquidity result in less high frequency trading being done on decentralized exchanges. Contrary to traditional markets, the markets are more inefficient have wider spreads and slower news propagation.

Another common difference the two markets volatility. As seen in Figure 2.9, the 5 day moving for Bitcoin's variance is high. (Correlate this to twitter activity?)

2.0.6 Trading Algorithms

Machine learning trading algorithms have been shown to have success trading in traditional stock markets for some time now (Chang, Lui, 2009)[2]. Even with

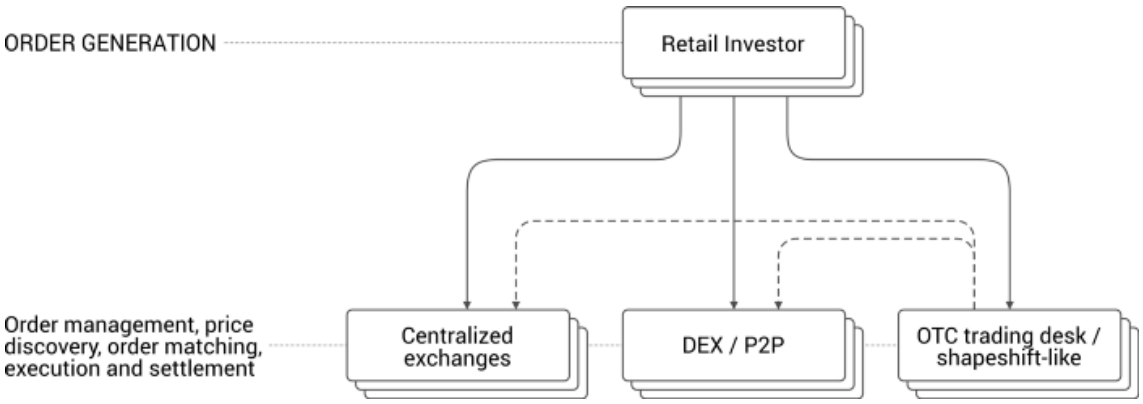

Figure 2.8: Cryptocurrency Order Flow

the recent emergence of various cryptocurrencies, machine learning has also shown to outperform traditional buy and hold strategies over a one year period with 12 cryptocurrencies (Jiang, Liang, 2017)[10]. One of the biggest differences cryptocurrencies compared with traditional stock markets is the short term volatility. This volatility has been partially attributed to online factors. The strongest correlation between an online factor and the price of a cryptocurrency was the polarization of opinions on Twitter about the digital asset (Phillips, 2018)[11]. The polarization of opinions on Twitter often indicated that positive price action for the cryptocurrency in the medium-term while other factors like security breaches had a greater impact on the price in the short term (Phillips, 2018)[11]. In the medium term, opinions seem to have the greatest impact on price action. In the short term, facts, or news, have the greatest impact on the price action. Ideally, an algorithm using these factors after completing a sentiment analysis would have an advantage over algorithms that do not have any understanding of public opinion.

"The relationships link online activity increases to price falls (the converse is not observed). It is not surprising that occasionally discussion is associated with price falls, as negative events (e.g. blockchain bugs, and exchange hacks) are newsworthy in the community." (Phillips, 2018)[11] A long term indicator for online activity that

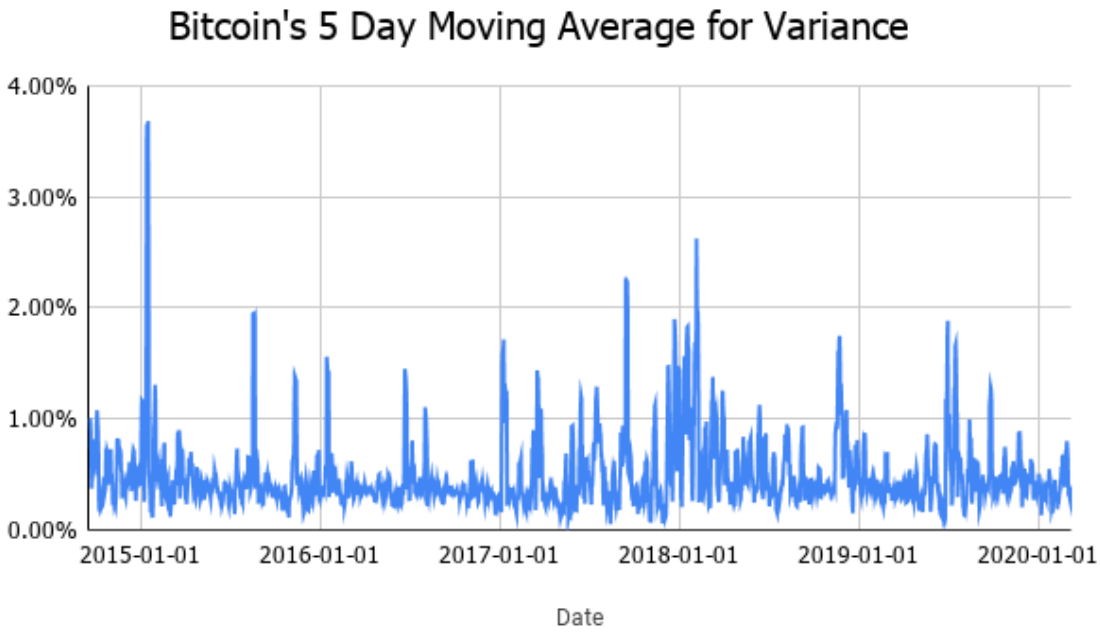

Figure 2.9: Bitcoin's 5 day moving average from 2015-2020.

has proven to be positively correlated with price movement is activity on Reddit. [Aside: this article proposes that this long term factor is because as there is more technological progress, there is more discussion and news related to it in these forums. Additionally the author suggests a future project looks at the github pages because the project are all open source. Could be a cool new factor] There are also periods in the medium to long time range that create bubbles from a positive feedback loop where an increase in discussion online about a cryptocurrency results to more usage and higher prices which in turn creates more discussion about the cryptocurrency (Phillips, 2018)[11]. This often creates a bubble because the price is exceeding the actual value of the token from the positive feedback loop.

Between 7-10 days an amount of positive user replies were successful in predicting price fluctuations (Kim et al, 2016)[7]. This report found that smaller communities had less accuracy in predicting price fluctuations. Since the time the report was

published, 2016, there are many more active communities with many daily active users.

Figure 2.1 shows the relation between price movement of the cryptocurrency PinkCoin and number of tweets mentioning PinkCoin. PinkCoin has a small market cap so the sample size could be considered small, but from the graph it is clear that a huge spike in the number of tweets about the cryptocurrency preceded jumps in price which was commonly followed by a dramatic drop. This is probably due to the bubble like behavior described by Phillips.

Our analysis reveals that increases in opinion polarization and exchange volume precede rising Bitcoin prices, and that emotional valence precedes opinion polarization and rising exchange volumes (Phillips, 2018)[11]. Emotional valence refers to the negativity or positivity of a comment. Opinion polarizations refers to conflicting viewpoints about a topic.

A group researching cryptocurrency trading using machine learning got positive results from a model that they trained to trade by letting it control trading decisions on a sliding scale from -1 to 1. -1 represents completely short which means opening a short trade worth 100% of the accounts value. This means that if the price drops 10%, the trade will be worth 1% of the starting value. 1 means the same but for the long side, a price increase of 1% means the trade value increases by 1%. By using a scheme like this, the model is able to learn how to risk different amounts in different scenarios instead of either relying on a traditional method of risk calculation or by simply using only the cases -1 or 1. This group trained their model with a reward system that calculated reward considering the length of a trade. For example it would be a better idea to take two trades that give 1.5% in 1 day each rather than take one trade that returns 2% in two days. By dividing their reward by time they effectively have given the model a way to interpret opportunity cost. This means

that the model will be incentivized to choose time effective trades instead of purely basing decisions on predicted profit or loss (Koker and Koutmos, 2020)[8]. In a similar fashion, another group made a trading model that was different than all the others so far. They used a portfolio management system instead of just trading one asset pair. The model would output percentages for each asset to be traded including cash. This way the model would output allocation specifications and the group simulated what the account value would be after executing trades to reach the models allocation numbers. Like the other group, they also had a time element to their reward but in this case they also divided the profit by the initial investment as well to give ROI over time (Laura et al, 2018)[9].

Machine learning algorithms face a new set of challenges with cryptocurrency that are not present in traditional exchanges. Machine learning algorithms trading on traditional exchanges have access to many macroeconomic variables that have been strong indicators of price movement (Koker and Koutmos, 2020)[8]. However, cryptocurrency does not have strong macroeconomic variables.

As an alternative to machine learning, technical analysis has also proven to be easier to implement and still effective (Anghel, 2020)[1]. In some scenarios, certain inputs for machine learning algorithms actually underperform traditional technical analysis (Anghel, 2020)[1]. Many current reports on effective machine learning algorithms are victims of data dredging (Anghel, 2020)[1]. Data dredging is misusing data analysis to find patterns and present results as statistically significant.

2.0.7 Sentiment Analysis

Sentiment Analysis has proven to be a useful indicator when determining price fluctuations of an asset in the past. One of the earlier studies to do this was a study that tried to predict cryptocurrency price fluctuations based on user com-

ments and replies. They collected data from multiple online sources where users can post text and placed them into one of five categories: Very negative, negative, neutral, positive, very positive (Kim et al, 2016)[7]. They were able to predict fluctuations with 79% accuracy using a 7 day lag. Another study using deep learning algorithms achieved also a 79% accuracy in predicting price fluctuations of Bitcoin by conducting similar sentiment analysis on over 2 million tweets (Stenqvist, Lonno, 2017)[14]. They found the sentiment analysis to be most effective when splitting the data into 30 minute time blocks. There was also varying level of success by pairing the sentiment analysis to price data from up to 90 minutes later. The study definitively concluded that sentiment analysis of relevant tweets is a useful price predictor. A later study achieved 90% accuracy in predicting price fluctuations by conducting sentiment analysis on popular social media platforms about a specific asset (Colianni et al, 2018)[12].

As the cryptocurrency ecosystem becomes more popular, the studies are finding more and more success with less lag. Additionally, the volume of data to be analyzed has increased. Polling for tweets about any cryptocurrency will return a collection of recent tweets about the topic. The mechanisms for collecting the data and analyzing it have changed, but currently one of the most prevalent techniques is using Twitter's APIs along with a sentiment analyzing technique called VADER (Hutto, 2015)[6]. VADER, or Valence Aware Dictionary and sEntiment Reasoner, is a rules based module that can accurately interpret sentiment for social media type content. It is capable of interpreting slang, emojis, and hashtags. It understands negations such as "not good" and "wasn't that good". It properly analyzes punctuations and all caps. It understands intensity modifiers like "very" and "kind of". A high level overview of this process is as follows.

VADER is easy to use and returns a few metrics about the sentence it analyzed.

It scores the positivity, neutrality, and negativity. It also has a metric called compound that is the sum of the other three metrics. The compound score ranges from -1 to 1 with -1 being very negative and 1 being very positive.

Chapter 3

Methodology

3.0.1 Collect Historical Sentiment Data

Before we can perform sentiment analysis we need content to analyze. We chose Twitter as our source because we can collect a large number of data points within a small period. We set up a script to run for 6 months and gather tweets every 60 seconds. The script was written in python and utilizes the Twitter API. Our script maximizes the API's limits. As a result, our script queried 16,665 tweets a day, 694 an hour, and 11 a minute. By limiting our query to 11 tweets every minute we ensured a more even distribution of data.

The endpoint (`https://api.twitter.com/2/tweets/search/recent`) allows for the request to specify relevant search terms to be returned. Also, the request can include the tweet's timestamp, likes, retweets, replies, language, and quotes. The query also gets the followers of the original poster. Lastly, all the tweets collected were in English.

3.0.2 Collect Historical Price Data

In order to train our trading models we first need to figure out what types of data are useful for predicting trading returns. Some of this work was already done by researching past attempts at trading using machine learning. Since it's almost impossible to tell if a type of data will be useful for predicting trading returns, we

will most likely include some data into our set that we don't end up using for our final model. The types of data that we will consider include: price data, technical indicators and user text entries about cryptocurrency scraped from various sources. Using just price data and technical indicators would be a traditional way of training a model like ours but we hope to add some utility with the user text data being how we can gauge market sentiment.

To get our price data, we downloaded the historical candle data as a csv for multiple different crypto trading pairs in USD. This gives us a series of candles with their times. One candle consists of 4 prices: the open, high, low, and close price, where the open price is the first price that a trade is made at during that candle period whereas the close price is the last price traded at during the candle period. This data could serve as input data to the model as well as be used to calculate returns.

Technical indicators are certain values derived from things like price and volume data. We will use existing technical indicators as well as possibly creating new ones to use for our model. Since a technical indicator is calculated by using existing price info, it wouldn't be too hard to calculate custom indicators if we ended up using them. The advantages to using technical indicators over price data is that since many technical indicators oscillate from one region of values to another, it is easier to map the indicator output to a range of 0 to 1 with proper distribution which makes training models easier.

3.0.3 Train Child Models

The next step is to train our models. We want to train our models with different sets and combinations of data so we can learn what data is more able to be used for our problem. It might be the case that certain trading pairs and timeframes

might do better with models that use different input data. For example if you used a daily time frame trading strategy to trade on the 5 minute timeframe, you might not do well. It turns out that different data is sometimes better suited for a specific timeframe or trading pair.

The child models can be any assortment of traditional trading strategies to a trained neural network. The idea is that we find methods of trading that work well either in general or under certain circumstances. For example, traders consider the 200 day moving average of an asset to be a line in the sand between being "bearish" or "bullish". Bearish and bullish are terms used to describe if price is generally trending down or up respectively. Taking this in mind, we can figure that strategies that take more longs than shorts above the 200 day moving average would in general perform better than a similar strategy which took more shorts than longs.

We will be using the programming language Python because it has many modules that make data collection and manipulation as well as machine learning much easier and faster to do. We hope that using tools like Keras and TensorFlow will help us to quickly create machine learning architectures without having to worry about implementing them ourselves.

If possible it would be extremely useful to be able to make a trading pair agnostic model. This would allow us to train the model using data from multiple trading pairs instead of just keeping it to one pair to a model. This would help because it would allow us to use the same model and strategy for all or most of the trading pairs we are looking at while vastly increasing the amount of test data available.

3.0.4 Train Parent Model Using Child Models

Once the lower models are trained, we can now take the output of those models and feed it to a new model along with their respective effectivenesses. This new

model should then be able to interpret the input and come up with a trading decision. This step may not be feasible or even practical depending on the type of output we get from our previous models and whether or not we are able to develop asset agnostic models.

3.0.5 Stream Sentiment Data

The model will be making decisions in real time, therefor the historical tweets API previously used is not sufficient. Instead the model will be directly interacting with Twitter's streaming endpoint that opens a connection and delivers content that matches a specific rule set as their tweeted. After specifying a timeframe, the script gathers and analyzes the tweets. It finally finds the average normalized, weighted composite sentiment and variance at the end of the time frame. This will become a single input data point for the machine learning algorithm to use.

As the project goes on tailoring the rule set will be an important part of optimizing our model. The current configuration retrieves all tweets containing "Eth", "Ethereum", "ETH", or "$ETH". There's two rules that use those keywords. The first returns tweets that have been annotated by Twitter's algorithm as content that is referring to politics. The second just returns any tweet with the keyword. Tweets gathered that match the first rule will be given a slightly higher weight when calculating overall sentiment because they are generally more significant tweets.

3.0.6 Automate Trading using Parent Model

We will most likely employ the simple method of having the trading states of completely short and long mapped from -1 to 1 respectively. The benefit of doing it like this would be that our model doesn't have to output any sort of price, it only needs to output a number from -1 to 1 to signal to our trading algorithm to market

sell or buy depending on the change in output between the two steps. For example if the model starts at an output of 1 but drops to 0 we know that we need to exit the long position by selling the same amount used to open the trade at market price.

These trades would all be executed on margin so that we don't need to worry about actually buying and selling the assets in question. The way this works is that if we had 1000 dollars on an exchange and we want to short BTC, if we weren't using margin we wouldn't be able to enter the trade because in order to enter a short trade, you must sell the asset which at the time was not in the account. By using margin, the trader is able to borrow the asset of the exchange to sell based on the amount of capital in their account. At some exchanges, the amount lent can be up to 100 times the capital in the account. Obviously using leverage like that is dangerous so we will only use margin to borrow a value that matches the account value simply for the utility of being able to both short and long with just cash.

The exchange we will be using to trade is Kraken. We will be using this exchange because it is one of the highest volume exchanges that services American citizens. There are many other exchanges that might have been better but since they aren't legally allowed to work with American citizens, we would be risking being kicked off the platform as well as possible loss of control of funds. For these reasons, it is not worth it to use an exchange like Binance, even though it would be better in every way. The one other clear option would be Coinbase Pro because it has higher volume than Kraken. As can be seen from the pie chart in Figure 4.1 below, Kraken and Coinbase Pro make up less than five percent of the volume of the 8 popular global exchanges included in the chart.

This is bad because it means that the exchange is more likely to be manipulated as the order books are thinner meaning that price can be moved the same amount for much less on low volume exchanges. Despite being a low volume exchange when

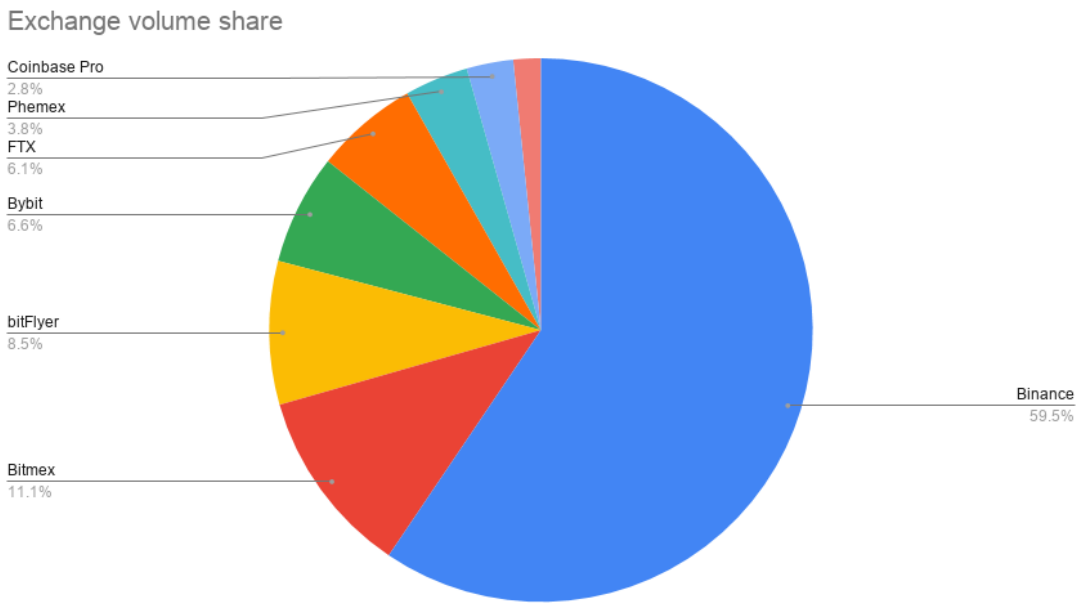

Figure 3.1: Share of 8 popular exchanges by volume

compared to global exchanges, Coinbase Pro and Kraken both take up a majority share of the volume from exchanges that are US friendly, which can be seen in Figure 4.2. The problem with Coinbase in particular is that its trading fees are higher than Kraken's for nearly all volumes of trading except after $50 million in 30 day trading volume as shown in Figure 4.3 below. Since we will almost certainly be trading under $50 million volume each month, Kraken is the superior choice in terms of fees. This is important especially for strategies which trade for small movements as well as strategies which make a higher number of trades than normal.

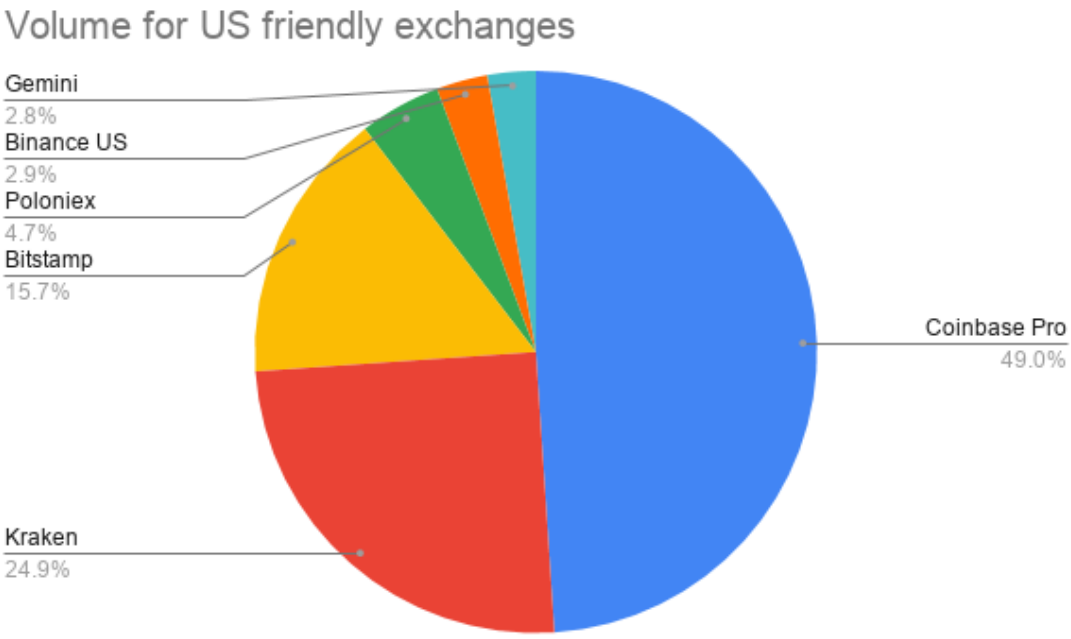

Figure 3.2: Share of 8 popular exchanges by volume

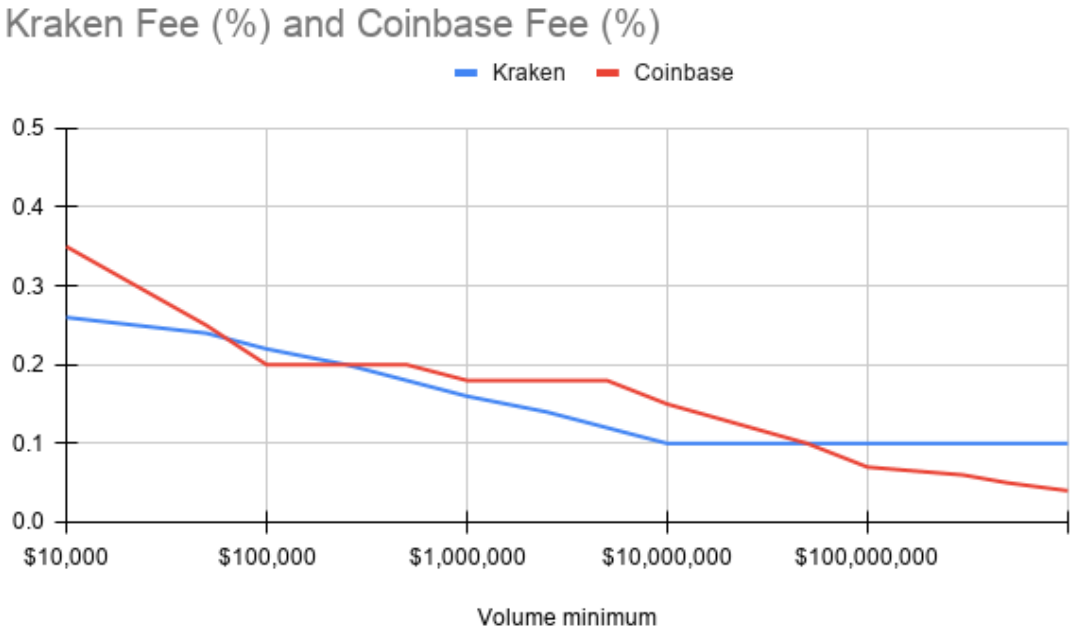

Figure 3.3: Taker fees for Kraken and Coinbase at different monthly volumes

Chapter 4

Results and Conclusions

At the conclusion of our development we were able to create a model that could trade profitably. However, getting to that point took a lot of refinement. One of the biggest challenges was first getting our sentiment analysis to have a meaningful correlation with Ethereum price movement. From the start in October to the end of the collection in January, we were able to collect around 500,000 tweets that included the keyword "Ethereum", "Eth", "$Eth", or "Eth". Our background explored grouping the data into various time frames and delays. One of the most common time frames was 2 hours with a 30 min delay. Our data unfortunately had too many gaps at that interval. Our data collection was done on a embedded system that was not connected to a monitor, so when unforeseen events like WiFi interruption or power outages occurred, it took human intervention to reboot the collection script. Trying to fill in the remaining data was difficult and dropping rows with missing data was not an option. It was critical for the model to be able to see all of the price data to accurately understand the trends from our technical indicators. Our best option was to modify the time frames. The smallest feasible time frame was 13 hours but we opted for a 24 hour time frame because it had the strongest correlation with our data. Even though the market is open 24 hours, the tweets we collected mostly originated from the US. The majority of trades from the original posters of the tweets are conducted between 7am to 8pm. This reasoning is most likely why this time frame was the most useful for our model. Our algorithm

was then refined to look at the previous days sentiment and technical analysis and make a decision at 7am EST to either hold a position, buy, or sell an active position. While the algorithm may miss out on intraday opportunities, it was much more reliable at the 24 hour time frame. We implemented a 7% stop loss to prevent too much damage from incorrect decisions.

Our best model was a Multilayer Perceptron but we also included the three other models discussed in the background into the ensemble. It achieved an r-squared value of 0.37 during testing and continually improved during the testing phase as more data from twitter was gathered in real time. Although the model ultimately was making a classification decision as to if the price would increase or decrease over the next time frame, regression was more useful. The algorithm predicted the % increase or decrease for the next day and if it was within a 90% confidence interval it would make a decision, otherwise it would ignore or close the current position if it was not confident enough.

Figure 4.1 was the final result of our model but to further illustrate the issues we had with trying to fit the model with a 2 hour time a figure of the fit using that time frame is in figure 4.2. The model had a r-squared of -0.12. Obviously, this was not sufficient to trade profitably.

Regardless of the time frames, we were able to successfully extract and analyze data, create meaningful technical indicators, and finally make a profitable automated trading model. The model will benefit from additional testing over a longer period to understand its sustainability. The model is currently still running and additional testing is being done.

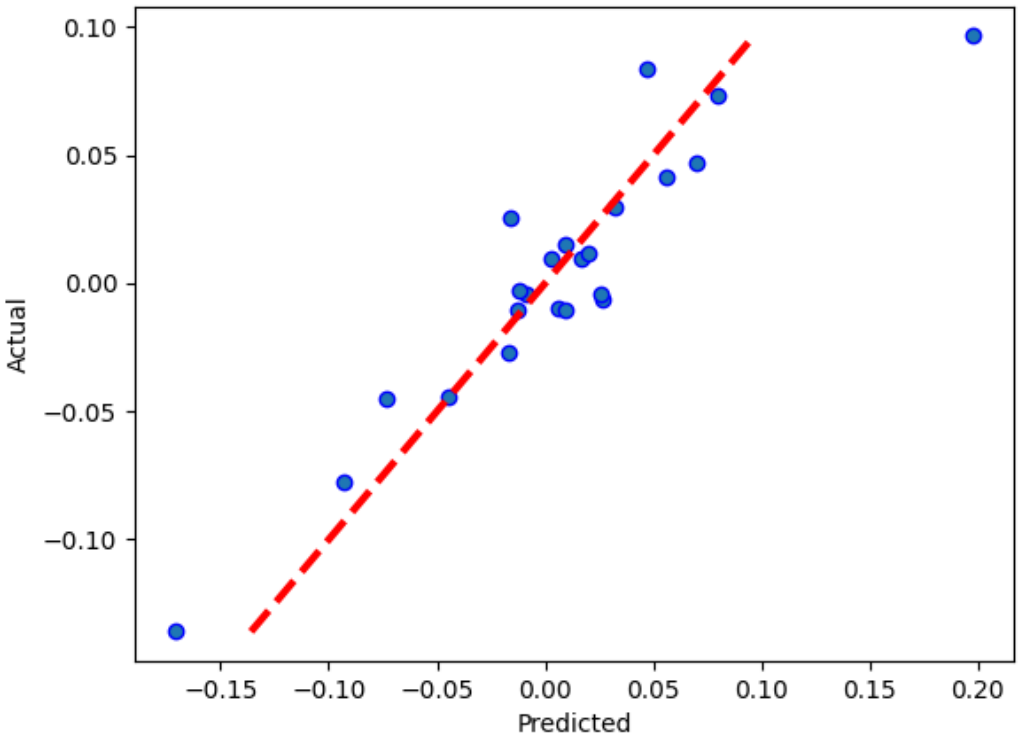

Figure 4.1: Fit of MultiLayer perceptron during testing, 24 hour time frame

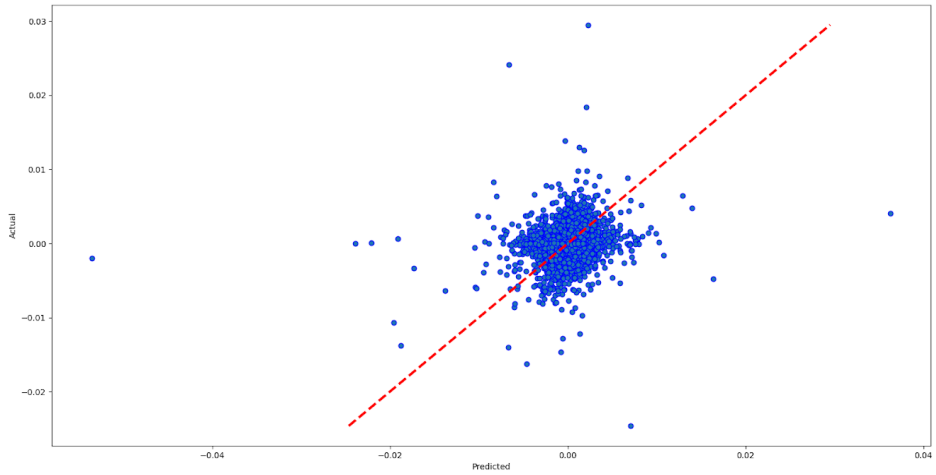

Figure 4.2: Fit of MultiLayer perceptron during testing, 2 hour time frame

Chapter 5

Limitations and Future Work

Like every other project, we were limited on both the scope and time available for our project. We initially thought we would be able to trade based on our model live for longer but due to time limitations we haven't been able to trade live for long enough to be fully confident in its success yet. This is something we will continue to do in the future and is certainly the next step for this project.

One improvement to make could be further developing the algorithm for taking the output from the model and turning that into trading decisions that consistently profit. This work is the final bridge between the model and making profitable trades but hopefully that will become easier as our model improves over time.

Another consideration is making sure that the system that makes live trades has a notification system of some kind to make sure that any technical problems like lost power or internet are solved as soon as possible. We could also switch the hosting of our live trading to a hosting service instead, which comes with its own problems, but it could be worth it. This test will give further proof that our system works even with real world problems like human error or technical problems. We focused on getting our model to successfully trade ETH mostly due to time concerns.

A cryptocurrency we were looking at, XRP, had its trading ability shut down on all but a few small exchanges in the US, which made it nearly impossible to trade live, making it not worth it to pursue. In the future, XRP trading should be available again in the US and we can trade and train our model on it and other

cryptocurrencies.

Bibliography

[1] Dan-Gabriel Anghel. "A reality check on trading rule performance in the cryptocurrency market: Machine learning vs. technical analysis". In: *Finance Research Letters* (June 20, 2020), p. 101655. ISSN: 1544-6123. DOI: 10.1016/j.frl.2020.101655. URL: http://www.sciencedirect.com/science/article/pii/S1544612320304414 (visited on 09/14/2020).

[2] P Chang et al. *An Ensemble of Neural Networks for Stock Trading Decision Making*. Berlin: Springer Berlin Heidelberg, 2009, pp. 1–10.

[3] Yung-Ho Chang, Chia-Ching Jong, and Sin-Chong Wang. "Size, trading volume, and the profitability of technical trading". In: *International Journal of Managerial Finance* Vol.13.4 (2017), pp. 475–494.

[4] Thomas Fischer, Christopher Krauss, and Alexander Deinert. "Statistical Arbitrage in Cryptocurrency Markets". In: *Journal of risk and financial management* 12.1 (Feb. 13, 2019), p. 31.

[5] G Fumera and F Roli. "A theoretical and experimental analysis of linear combiners for multiple classifier systems". In: *IEEE transactions on pattern analysis and machine intelligence* 27.6 (June 2005), pp. 942–956.

[6] CJ Hutto. *VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text*. Jan. 2015. URL: https://www.researchgate.net/publication/275828927_VADER_A_Parsimonious_Rule-based_Model_for_Sentiment_Analysis_of_Social_Media_Text (visited on 09/14/2020).

[7] Young Bin Kim et al. "Predicting Fluctuations in Cryptocurrency Transactions Based on User Comments and Replies". In: *PLOS ONE* 11.8 (Aug. 17,

2016), e0161197. ISSN: 1932-6203. DOI: 10.1371/journal.pone.0161197. URL: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0161197 (visited on 09/14/2020).

[8] Thomas Koker and Dimitrios Koutmos. "Cryptocurrency Trading Using Machine Learning". In: 13.8 (2020). DOI: DOI:10.3390/jrfm13080178.

[9] Alessandretti Laura et al. "Anticipating cryptocurrency prices using machine learning". In: *Alessandretti, Laura; ElBahrawy, Abeer; Aiello, Luca Maria; Baronchelli, Andrea* (Nov. 9, 2018). DOI: DOI:10.1155/2018/8983590.

[10] J Liang and Z Jiang. *Cryptocurrency portfolio management with deep reinforcement learning*. London, Sept. 2017, pp. 905–913.

[11] RC Phillips and D Gorse. *Cryptocurrency price drivers: Wavelet coherence analysis revisited*. 2018.

[12] Otabek Sattarov et al. "Recommending Cryptocurrency Trading Points with Deep Reinforcement Learning Approach". In: *Applied Sciences* 10.4 (Jan. 2020), p. 1506. DOI: 10.3390/app10041506. URL: https://www.mdpi.com/2076-3417/10/4/1506 (visited on 09/25/2020).

[13] Devavrat Shah and Kang Zhang. "Bayesian regression and Bitcoin". In: (Oct. 6, 2014). URL: https://arxiv.org/abs/1410.1231.

[14] Evita Stenqvist and Jacob Lönnö. *Predicting Bitcoin price fluctuation with Twitter sentiment analysis*. 2017. URL: http://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-209191 (visited on 09/25/2020).

Appendices