Forecasting Cryptocurrency Price Movements with Deep Learning and Sentiment Analysis

EC331 Research in Applied Economics

University of Warwick

Student ID: 1610710

Word count: 4987

Abstract

The amount of available cryptocurrency news data is constantly growing, posing the question of whether this can be exploited to create profitable automated cryptocurrency trading systems. This research explores the viability of using deep learning models to forecast changes in cryptocurrency prices using a vast data set of news articles and social media posts. A robust sentiment analysis model is used to extract sentiment scores, converting the news data into a quantitative data set. Three different models are built with corresponding trading strategies to test their profitability using a large 7-month test period of unseen data. All three models perform very well, especially during volatile periods, with the best model achieving an overall return on trades of 287.9%, as well as 91.3% profitable trades during the test period. One of the key findings is that models incorporating news data greatly outperform those that do not, suggesting that the Efficient Markets Hypothesis may not be applicable to cryptocurrency trading. Additionally, incorporating technical analysis indicators into the data set seems to make a small impact on the profitability of the models, bringing into question their widespread popularity.

Acknowledgements

I would like to thank my supervisor, Jonathan Cave for his insightful comments, suggestions and feedback. I also wish to acknowledge the Economics Department at the University of Warwick, coinmarketcap.com for hosting cryptocurrency price data, CryptoCompare for hosting the news articles and pushshift.io for providing an API used to scrape the Reddit data.

TensorFlow and Keras were used to build and train all the models for this research. Additionally, Numpy and Pandas were used to perform most of the data pre-processing in Python.

Table of Contents

Chapter 1: The Scope and Methods of Study

1.1 Introduction

Cryptocurrencies and their underlying technologies - such as Blockchain in the case of Bitcoin and Smart Contracts in the case of Ethereum - are regarded as being in a similar state to the early Internet, where predicting which coins will prevail is seemingly almost impossible.

One of the main focuses of this research is to assess the impact of news data on the profitability of trading strategies that predict short-term cryptocurrency price movements. It is very common to observe a dramatic increase in the price of a coin soon after the company behind it announces a partnership with another major company or lose a substantial amount of its value after receiving bad publicity. Al-Khazali et al (2018) find that the volatility of Bitcoin's price is greatly affected by both positive and negative news data, whereas the volatility of a more traditional asset such as gold is mostly unaffected by news.

For the remainder of this document, the term 'news data' will be used to describe news article headlines, previews and bodies, as well as Reddit submissions and comments.

1.2 Background

The literature surrounding the subject of financial time series forecasting presents various techniques that can be classified into four major categories:

- Fundamental Analysis
- Technical Analysis (TA)
- Linear Modelling
- Machine learning (ML)

This research will focus almost entirely on ML methods, specifically, a type of Recurrent Neural Network (RNN) model called Long Short-Term Memory (LSTM). This type of network is widely used in time series forecasting due to its ability to 'memorise' patterns over long time sequences.

In total, three types of model will be implemented:

- (1) Buy/Sell: predicting 1 for buy signals and -1 for sell signals

- (2) Buy/Sell/Hold: predicting 1 for buy signals, 0 for hold signals and -1 for sell signals
- (3) The best out of (1) and (2) with technical analysis indicators added to the data set

1.3 Research Hypotheses

The null hypothesis (H_0) is that profitable machine-learning-based trading strategies utilising sentiment analysis[1] on news data cannot be created, after accounting for reasonable trading costs. The corresponding alternative hypothesis (H_1) is that such trading strategies can be created. In addition to these, an additional hypothesis (H_2) can be formed stating that incorporating technical analysis methods does not improve the profitability of the models being tested.

Malkiel (2003) states that markets are very efficient at absorbing news information and that using models based on technical and fundamental analysis is equivalent to randomly selecting a portfolio of stocks. This theory is called the Efficient Market Hypothesis (EMH), which proposes that price returns are independent of past price performance and that the best forecasting element is the current price. Furthermore, the Random Walk Hypothesis (RWH) states that stock market prices evolve according to a random walk, and thus have no predictability, suggesting that H_0 cannot be rejected. However, Darrat et al (2000) implement a ML model that successfully makes predictions for various Chinese stocks over a twelve-month test period - contradicting the RWH.

Finding evidence to reject H_0 in the context of cryptocurrencies could motivate further research using similar modelling techniques with other financial instruments such as stocks. Additionally, it may encourage governments to regulate cryptocurrency trading further, since observing predictable price patterns using news data could be evidence of market manipulation performed by ‘pump and dump’ groups on social media platforms such as Discord and Reddit. Li et al (2018) explore the effects of such groups and find that they can be very disruptive to the prices of new coins with low market capitalisations.

To test these hypotheses, multiple trading strategies will be built for each model type outlined above using the same time period of unseen trading data for a specific group of cryptocurrencies.

[1] Sentiment analysis is a subset of Natural Language Processing (NLP) that measures the opinion, attitude and/or emotion of text and categorises it into either ‘positive’ or ‘negative’ with an assigned sentiment score between -1 and 1.

Chapter 2: Data Collection and Manipulation

2.1 Data Collection

To conduct the research, 80,532 news articles are collected from CryptoCompare for 11 popular cryptocurrencies. In addition, 1,980,723 Reddit submissions and 21,795,734 Reddit comments are also collected discussing 43 different coins. The time period used to train, validate and test the models starts on 01/01/2017 and ends on 24/03/2019. The year 2017 was picked as the starting date since this was when cryptocurrencies began to obtain significant public attention, leading to a vast number of articles being written discussing their future potential.

Daily price data for over 250 of the largest cryptocurrencies, sorted by market capitalisation, is collected from coinmarketcap.com - containing data on Open, High, Low, Close, Volume and Market Capitalisation.

To accurately compare the performance of each model type, a subset of the data is chosen containing sufficient price and news data. The eight coins that satisfy these constraints are: Bitcoin (BTC), Ethereum (ETH), Ripple (XRP), Litecoin(LTC), Monero(XMR), Zcash (ZEC), Ethereum-Classic (ETC) and Dash (DASH).

2.2 Data Pre-Processing

There are various ways to perform sentiment analysis, however, the method used employs a sentiment lexicon that maps certain words and phrases to sentiment scores to generate a continuous score between -1 and 1, where -1 is very negative, 0 is neutral and 1 is very positive.

Since creating a good lexicon is a very complicated process, a well-established sentiment analysis tool called VADER is used as it performs better than a lot of its counterparts, as found by Hutto et al (2014). One of the drawbacks of using a system like VADER is that sarcastic comments in text are typically misclassified, as the words used by these phrases carry negative connotations when examined separately. Zhang et al (2016) show that using a RNN model to perform sentiment analysis can overcome this. However, allocating time and computational resources to building such a model would not have been viable given the scope of the research. Some examples of sentiment scores assigned to various test news headlines can be found in Table 5 in the Appendix.

One of the challenges of the news data pre-processing is devising a method that takes results collected at irregular time intervals and returns an evenly-spaced time series. Since daily price data is used, the mean of the sentiment scores for each day is taken. One drawback of using this method is that every post is weighed equally, even though certain posts would have reached a wider audience and thus may have had a larger impact on prices. Since the data collected did not have any popularity metric (such as 'up-votes' in the case of the Reddit data), a weighting factor could not be incorporated in the data aggregation process. This is an aspect of the research that could be improved upon if an appropriate data set were found.

Each prediction target is either 1 or -1, denoting whether the close price increases or decreases, respectively, after a given time period. Three prediction periods are tested: 1, 3 and 5 days into the future, with the possibility of adding more if the model's performance increases as the prediction period length increases.

The time series data is separated into sliding windows with sequence length n, which are shifted forwards by one day at a time. One of the three prediction targets is then assigned to the end of each window. To determine a good value for n, different values are tested between 50 and 250, in increments of 25, using a baseline model. The results of this process can be found in Table 7 in the Appendix and suggest that $n = 100$ should be used.

Many research papers and articles using deep learning models to forecast financial time series data omit one key pre-processing characteristic - they overlook the fact that the model will not be capable of predicting values outside the range of values it receives as input, as pointed out by Schultze-Kraft (2019). This statement is further backed up by Huang et al (2004). One solution is to use the percentage change of the time series values instead of the raw values. This way, the model is more likely to be able to predict large price changes, if the data contains some examples of these.

Another greatly overlooked feature is data balancing. For example, if most training targets are of a certain class, the model can artificially achieve an accuracy greater than 50% by only predicting that class. To counteract this, the training targets are balanced so that there are equally as many buy and sell signals, forcing the model to have to learn the underlying trends in the data to increase training accuracy. A drawback of balancing is that some samples are dropped, decreasing the number of examples the model can use for training.

Finally, the pre-processed data is split into training (50%), validation (20%) and testing (30%) sets. The purpose of these will be explained in Section 3.2.

Chapter 3: Buy/Sell Model

3.1 Buy/Sell Model Overview

The RNN model can be built using a variety of layers, where each serves a distinct purpose. The main one used is called `CuDNNLSTM`, which is a very optimised LSTM layer using Nvidia's CuDNN framework. This layer is a good choice, as it is frequently used for time series forecasting with great performance and accuracy, as demonstrated by Lei et al (2017).

Since trends in cryptocurrency prices change over time, another layer called `Dropout` is added after the `CuDNNLSTM` layer. Dropout is a regularisation technique that selects neurons in the network that are ignored during different steps in the training process. This prevents the network from over-fitting, where the noise in the training data is learned instead of the general trend. In the context of time series forecasting, not utilising dropout layers could cause the model's performance to decrease when a change in a market trend is encountered. For example, if most of the training set is dominated by a period of exponential growth - as was the case for cryptocurrencies throughout most of 2017 - the addition of dropout layers could assist the model with identifying changes in market trends. A dropout probability of 20% is used for each `Dropout` layer. Additionally, a `BacthNormalization`[2] layer is added after each `Dropout` layer, since Ioffe et al (2015) claim that it decreases training time for RNNs.

The model is comprised of three sets of `CuDNNLSTM`, `Dropout`, and `BatchNormalisation` layers, for a total of nine initial layers. Since the model is set to perform classification on two classes, a fully connected[3] `Dense` layer is added at the end of the model structure with two output units. A summary of the model can be found in Table 8 in the Appendix.

[2]Batch normalization is sometimes added after a layer to prevent the distribution of the outputs from changing between layers.

[3]Every neuron in a fully connected layer receives an input from every node in the previous layer.

3.2 Buy/Sell Model Training

Throughout the training process, two main metrics are monitored: accuracy and loss. Accuracy measures the percentage of targets that are correctly classified and loss measures how close the predicted probabilities for each class are to the ground truth probabilities, where lower values indicate more accurate predictions. The loss function used to train the models is called categorical cross-entropy, defined as:

$$-\frac{1}{N}\sum_{i=1}^{N}\log(p(x_i))q(x_i),$$

where N is the number of training samples, $p(x_i)$ is the probability (or confidence value) returned by the model for prediction x_i and $q(x_i)$ is the actual probability for prediction x_i (which is either 0 or 1).

To prevent over-fitting, the validation accuracy and loss are monitored throughout the training process. This is crucial, since only monitoring the training accuracy and loss gives an unrealistic representation of how well the model preforms on unseen data. Figure 1 shows an example where the validation loss begins to increase after 50 epochs. Therefore, the weights used for the final model should be the ones minimising the validation loss.

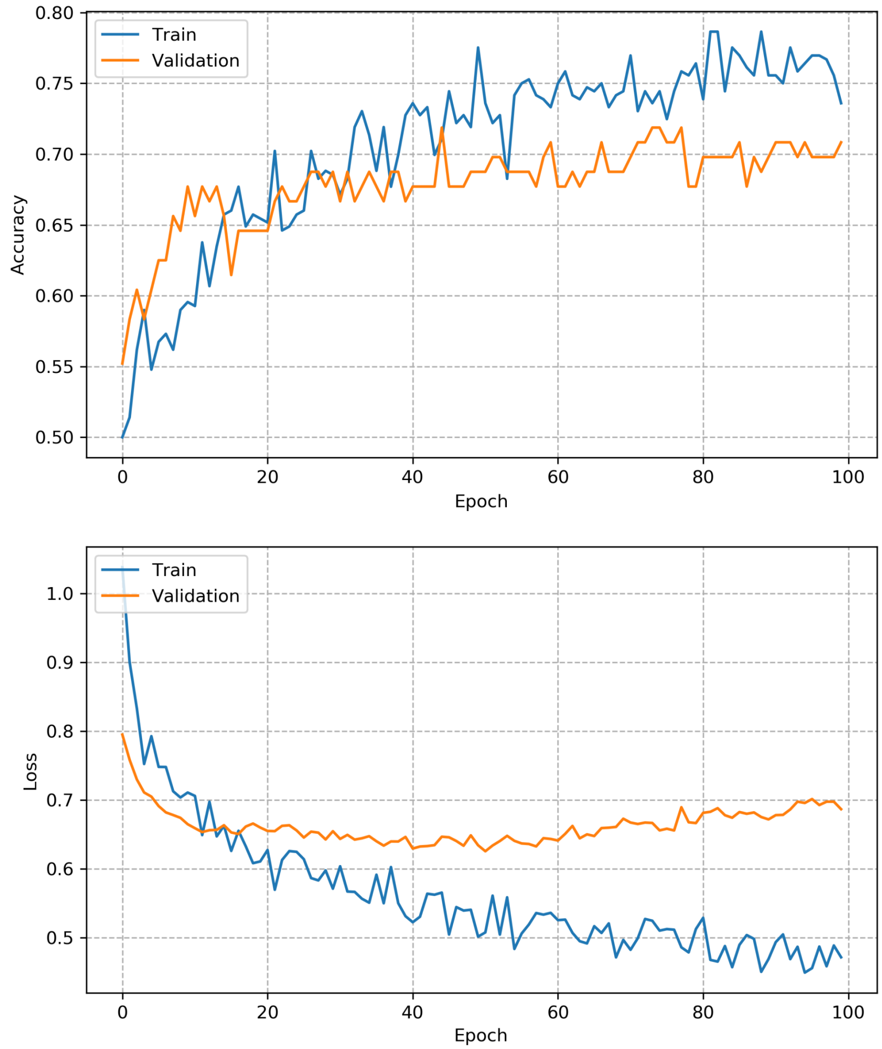

Figure 1: Training and validation loss and accuracy

It is worth mentioning that models trained with Keras do not output feature importances, i.e., how much impact each variable has on the prediction target. Some other ML models such as Random Forest provide this summary, however, they are not likely to perform as well as an LSTM model on the data set being tested. This should not be an issue, since RNN models have been shown to ignore features that have little predictive value, as demonstrated by Putchala et al (2017).

3.3 Buy/Sell Model Validation Results

To compare the performance of traditional model such as OLS; Analysis of Variance (ANOVA) tests and F-tests can be used. Since deep learning models are used for this research, such tests cannot be performed. Instead, the metrics used to compare the relative performances of the models are the classification accuracy and categorical cross-entropy loss.

For most ML problems, it is common to test the model on the whole test set in a single batch. Since trends in cryptocurrency prices are likely to change throughout the 7-month test period, a method called walk-forward testing is used. This works by predicting the first value in the test set using the training set, then moving the correct value into the training set and repeating the process until the test set is empty. Since it is a reasonable assumption that models will be re-trained each day with updated data, this testing method should produce a more realistic use case. Figure 2 depicts the walk-forward testing process.

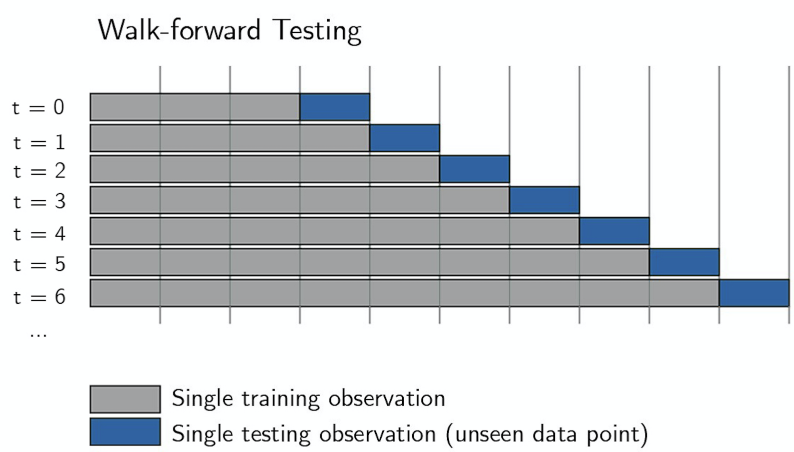

Figure 2: Walk-forward testing

Table 1 shows that the models incorporating sentiment data perform better than their counterparts with no news (*). In fact, the models with news data have higher accuracies by 4.5 percentage points on average compared to those without news data.

Table 1: Buy/Sell Model - Validation Results

Model	Accuracy	Categorical Cross-Entropy
1 Day	53.6%	0.7562
3 Day	63.1%	0.5300
5 Day	62.4%	0.5754
1 Day (*)	53.2%	0.6940
3 Day (*)	55.0%	0.5628
5 Day (*)	57.3%	0.5963

The difference in accuracies between both types of model is small, but observable. However, the testing method does not reflect how useful each model would be in a real-world scenario.

3.4 Buy/Sell Model Trading Strategy Overview and Results

To create a more realistic testing environment for the models, trading strategies are built with the following rules:

- Generate a buy signal if the prediction target is 1, and a sell signal if it is -1
- Ignore the signal if the confidence value is more than two standard deviations below the mean confidence value, i.e., if it is abnormally low
- If two consecutive trading signals are the same, the position is kept open and only reversed when a different signal is encountered

The trading strategies are intended to be used on platforms which allow users to short sell assets such as Contract for Differences (CFDs).

For each of the six models in Table 1, a trading strategy is built using the eight cryptocurrencies outlined in Section 2.1. Each strategy begins with an initial account balance of $10,000, and every trade is sized at $1,000.

The main metric used to compare trading strategies is the Return on Assets (ROA) multiplied by the Equity Curve Correlation (ECC)[4]. This not only quantifies how profitable each strategy is, but also how volatile the equity curve is. Having a relatively straight equity curve with few drawdown periods should reduce the strategy's overall risk profile.

The confidence threshold resulting in the best performance metric could be selected using a genetic optimisation algorithm, however, this would lead to over-fitting. This is because the threshold found would be the best value for the test period and not necessarily for future predictions.

In addition, a buy-and-hold strategy is also created as a benchmark - a method employed by Cesari et al (2003), Fung et al (2004), as well as Metghalchi et al (2008). The only rule for this strategy is that a buy signal is made at the beginning of the test set and is closed at the end of the set. The initial trade is also sized at $1,000 to be directly comparable to the other strategies.

Figure 3 shows the trading signals for Bitcoin using the model with three-day predictions and news data. The blue triangles represent buy signals and the red triangles, sell signals. The strategy generates a correct sell signal before the relatively volatile period starting in mid-November and ending in mid-December, as well as a corresponding buy signal near the bottom of the downward trend. This may suggest that the model performs well when large price movements occur. Conversely, the period between mid-October and mid-November is an example of many trades reversing their positions very frequently, suggesting that the model may perform worse during less volatile periods.

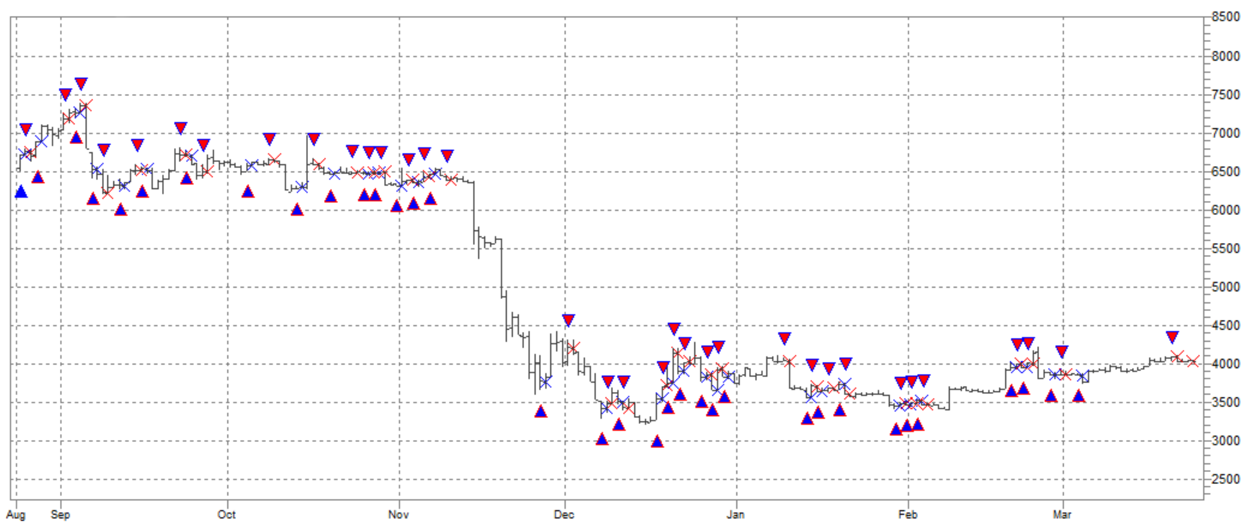

Figure 3: Bitcoin Buy/Sell trading strategy signals

[4]The ECC is a measure how close the equity curve is to a straight line with positive gradient.

Table 2 summarises the performance metrics for the trading strategies averaged over the eight cryptocurrencies tested.

Table 2: Buy/Sell Model - Trading Strategy Results

Trading Strategy	Return on Trades	Profitable Trades	ROA * ECC	Robust Sharpe Ratio	Average Trade Span (Days)
1 Day	203.3%	66.4%	171.9	3.10	4
3 Day	265.9%	70.7%	254.3	2.86	5
5 Day	219.5%	67.3%	211.5	2.39	5
1 Day (*)	116.6%	60.6%	92.8	1.80	4
3 Day (*)	98.1%	55.1%	79.2	1.59	4
5 Day (*)	103.6%	54.5%	90.3	2.00	4
B&H	-36.6%	12.5%	-54.8	0.00	212

The model with the best performance is the one incorporating news sentiment data and making predictions three days into the future. All the trading strategies based on the Buy/Sell model greatly outperform the buy-and-hold strategy. During the test period, the mean daily closing price of the eight cryptocurrencies decreases by 36.6%, which explains why the buy-and-hold strategy performs so badly. To beat this benchmark, the models are required to consistently predict correct price movement directions to maximise the Return on Trades (ROT), ROA * ECC and Robust Sharpe Ratio[5].

Figure 4 shows the cumulative return rate - averaged over the eight cryptocurrencies - for each trading strategy (TS) type. The return rate for all strategies, except the buy-and-hold strategy, exhibit behaviour close to a monotonically increasing function. Since there are hardly any drawdown periods in the curve, the strategies carry little risk over the test period, and thus result in an ECC value close to 1. Like the findings of Section 3.3, the strategies based on the models that use news data perform significantly better than those that do not over the test period.

[5]The Robust Sharpe Ratio computes the performance of an investment, adjusting for its risk, where values over 1.0 are desirable.

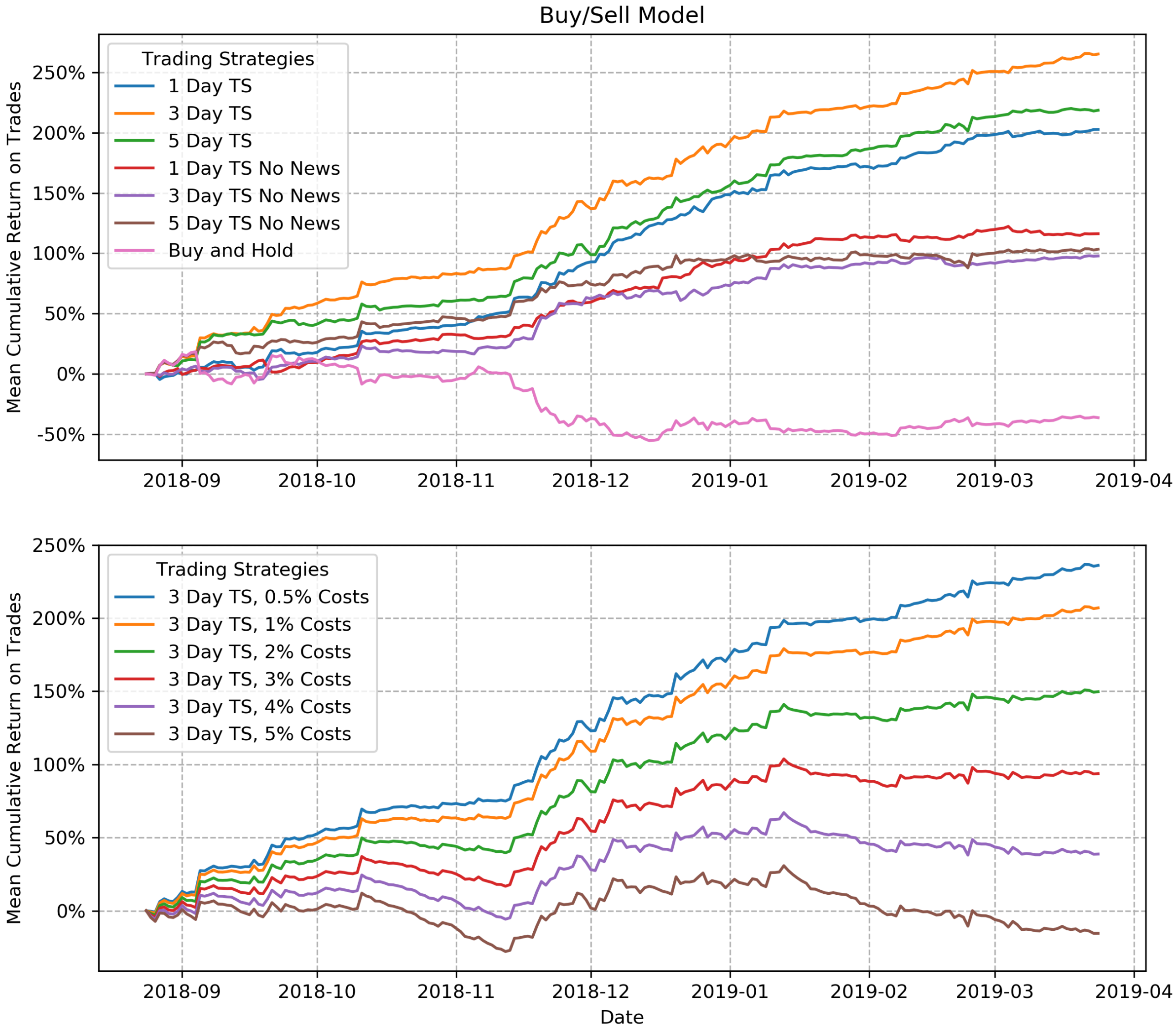

Figure 4: Mean Cumulative ROTs for the Buy/Sell model

The bottom graph in Figure 4 shows the impact of trading costs applied to the best strategy. The costs are calculated as the proportion of the size of a new open position. The reason the profitability decreases by such a large amount, to such an extent that at 5% costs the strategy makes a loss, is likely because the model makes trades very often, especially during periods of low volatility.

Chapter 4: Buy/Sell/Hold Model

Section 2.2 outlines the general pre-processing techniques that are used before training the models. In addition to these, a few key additions and changes are made for the Buy/Sell/Hold model. Firstly, periods with low future volatility are assigned a target of 0, periods of high volatility and a downwards movement, -1, and high volatility and an upwards movement, 1. Volatility is calculated by taking the standard deviation of the logarithm of the close values for 100 days preceding the prediction date. A value of 100 days is used since this is equal to the sequence length outlined in Section 2.2, meaning the volatility value is within the scope of the data that the model is able to see.

Since the prediction target can be one of three classes, the number of output units in the final `Dense` layer is changed from two to three. Aside from this, the model structure and training process is identical to that of the Buy/Sell model, with the exception that only models utilising news data are trained. This is because Section 3.4 demonstrates that trading strategies employing news data perform significantly better than those that do not.

Table 3 shows that, once again, the best performing model is the one making predictions three days into the future. It is worth noting that because the model now predicts one of three classes instead of two, random predictions would result in an accuracy of 33.3%. Therefore, an accuracy value close to 50% is likely to be relatively favourable.

Table 3: Buy/Sell/Hold Model - Validation Results

Model	Accuracy	Categorical Cross-Entropy
1 Day	62.6%	0.8180
3 Day	63.5%	0.9390
5 Day	59.2%	0.8070
7 Day	52.5%	0.7478

The trading rules for the new strategies are the same as those outlined in Section 3.4, with the addition of the 0 prediction targets being ignored, i.e., no new trades are generated during periods of low volatility.

Examining the trading signals generated by the three-day Bitcoin model shows that the trade frequency is lower compared to the Buy/Sell model, which may help to decrease overall trading costs. Additionally, most trading signals seem to be generated during volatile periods, which may improve upon the strategies made using the previous model.

Figure 5: Bitcoin Buy/Sell/Hold trading strategy signals

Table 4 and Figure 6 reveal that best performing model is the five-day prediction model, as it maximises both ROT and ROA * ECC. The one-day strategy performs significantly worse than the others. This is likely because volatility is calculated over a relatively large window, causing most training targets to be 0 - 66.2% are 0 in the case of Bitcoin - and thus hardly any trades are generated. A future improvement could be to set the volatility calculation period to be a function of the prediction period, i.e., the volatility calculation period would be smaller for shorter prediction periods.

The table also shows that the average trade span increases from five days for the best Buy/Sell model to fourteen days for the best Buy/Sell/Hold model, confirming that the new model makes trades less frequently. This could be the reason why the profitability remains high for the best strategy, even when accounting for relatively large trading costs of over 5%.

Since performance measured by the ROA * ECC metric increases for the new model compared to the previous one, the model adding technical analysis indicators should be based on the Buy/Sell/Hold model.

Table 4: Buy/Sell/Hold Model - Trading Strategy Results

Trading Strategy	Return on Trades	Profitable Trades	ROA * ECC	Robust Sharpe Ratio	Average Trade Span (Days)
1 Day	0.5%	44.8%	4.4	0.26	64
3 Day	199.3%	85.3%	193.0	1.28	15
5 Day	287.9%	91.3%	272.0	1.81	14
7 Day	246.7%	93.2%	240.6	1.54	17

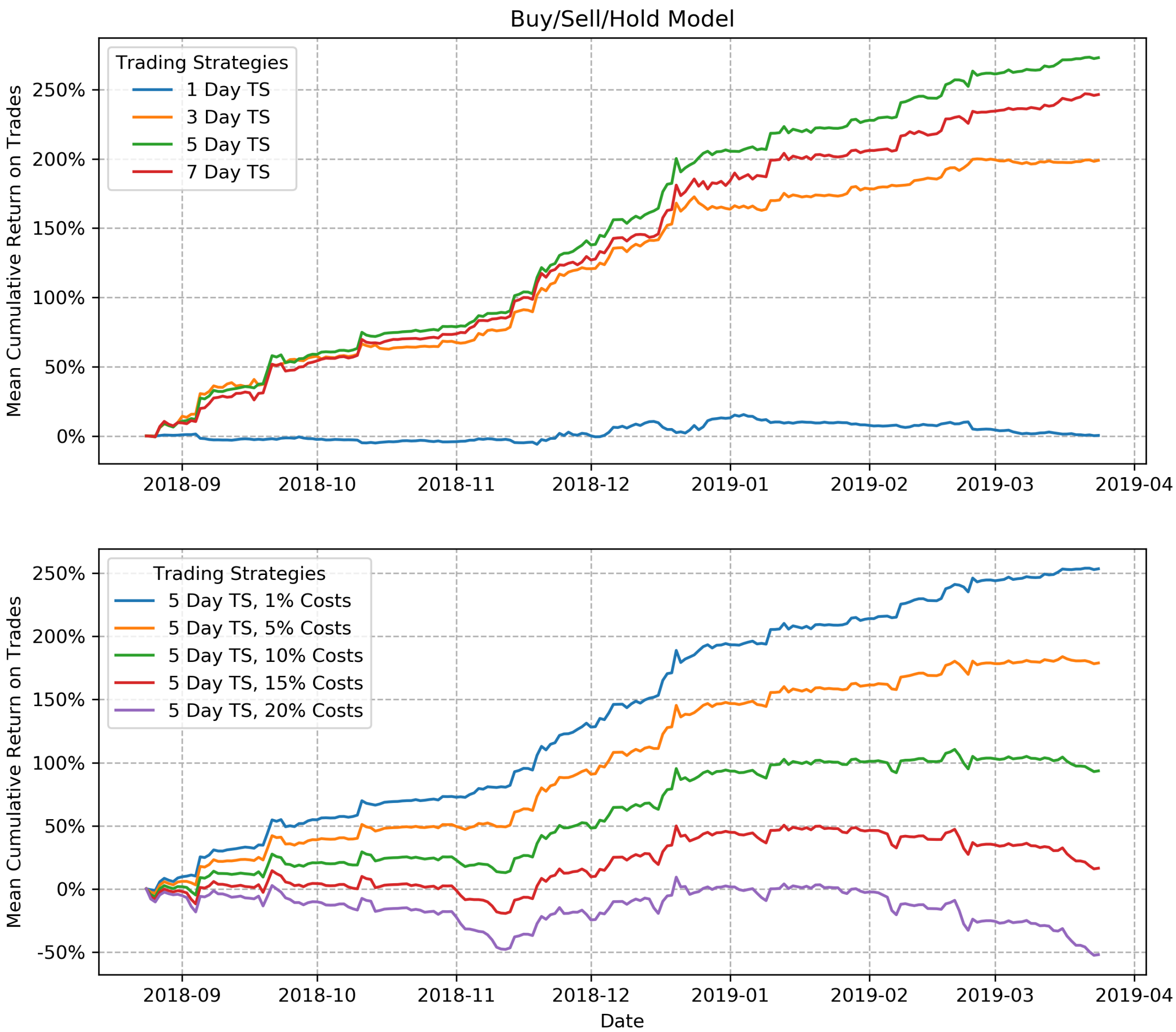

Figure 6: Mean Cumulative ROTs for the Buy/Sell/Hold model

Chapter 5: Buy/Sell/Hold + Technical Analysis Model

The data pre-processing for the new model is based on that of the Buy/Sell/Hold model, with the addition of various technical analysis indicators. These include popular indicators such as RSI, CCI and MACD, as well as many others. In total, 59 indicators are added to the data set of pre-processed price and news data. Figures 8-11 in the Appendix are some examples of the indicators used, before being pre-processed for training. Training for this model is identical to how it was performed for the Buy/Sell/Hold model, since the same three classes are being predicted.

Table 9 in the Appendix shows very similar results to those of the Buy/Sell/Hold model. This may be an indication that the model does not benefit from the technical indicators and assigns low feature importances to these in its internal structure. This is plausible, since the addition of technical indicators was the only change made to the model.

Similarly, the trading signals produced for Bitcoin in Figure 12 and the overall model results in Table 10 and Figure 13 in the Appendix closely resemble that of the Buy/Sell/Hold model. This suggests that the technical indicators added to the data set have little to no impact on the model's ability to predict price movements. However, this does not mean that TA indicators cannot contribute towards any model's predictability, only that they do not improve the models trained. Therefore, there is insufficient evidence to reject H_2.

Chapter 6: Conclusions and Recommendations

6.1 Conclusions

The inner workings of financial markets will most likely never be fully understood, as they are mostly governed by speculation and human error. However, the research in this paper has shown that there is some level of predictability in the short-term prices of the cryptocurrencies tested when a sufficiently large news data set is incorporated into the training set. This is demonstrated by the strategies beating the buy-and-hold benchmark and achieving high ROTs of up to 287.9% over the seven-month test period. In addition, the strategies are still profitable when accounting for significantly high trading costs. This suggests that H_0, as well as the EMH and the RWH should be rejected in the context of cryptocurrency trading. One reason why the EMH and RWH may not apply to cryptocurrencies is that their volume traded makes up less than 1% (at the time of writing) of the total volume of global financial assets traded, suggesting that large institutional investors may not have entered the market yet and exploited any available predictability.

It is worth stressing that no trading strategy will be able to accurately predict prices, however, the approaches used in this research could increase the odds of making profitable trades. It is also worth mentioning that the test period used could have led to favourable results by an element of chance, and the strategies may need to be validated again in the future when more price data is available. In addition, all the cryptocurrencies except for Litecoin experienced a downward trend during the test period. In contrast to this, most of the training and validation periods displayed an upwards trend. The fact that performance is good during the test period, given how different the training period is, suggests that the models can correctly identify whether the market is in an upwards or downwards trend.

Another main finding is that models utilising sentiment analysis performed over twice as well as those that did not, as measured by the ROA * ECC metric. This is likely due to the vast size of the news data set, since analysing so much news data could be compared to reading every article on a cryptocurrency and making a more informed trading decision. On the other hand, technical analysis indicators, at least the specific ones that were added to the data set, did not seem to have a significant impact on the performance of the trading strategies. This may be because they are features derived from the price data and thus do not add any new information that could benefit the models used.

The reliability of some of the news samples is questionable, especially in the case of the Reddit data where some posts are very off-topic. One way to mitigate this is to only include news data that contains a certain set of key words that could be related to price movements. These key words could be determined using a supervised learning algorithm, like the one used by Szarvas et al (2008) or Ko et al (2000).

On a final note, although the systems built may currently be profitable, large investors utilising high-frequency trading (HFT) would likely reduce the profitability of the strategies, as mentioned by Goldstein et al (2014). If this were to occur, the EMH would likely begin to apply to cryptocurrency assets.

6.2 Future Recommendations

In addition to news articles and Reddit submissions and comments, data from Google Trends and Twitter could be added to the data set. Google Trends measures how frequently certain keywords are searched on Google, which may be a good proxy for the popularity for certain cryptocurrencies. However, it is likely that Google search popularities will be a lagging indicator, as opposed to a leading indicator, meaning the data could have little predictive value. In addition, data from Twitter may not be of the highest quality and will only have an impact on prices if investors are reading it.

All the data used for this research is aggregated by day. Since the news data collected has a timestamp resolution of seconds and various cryptocurrency exchange APIs provide price data for each minute, it would be possible to reproduce this research using either minute-resolution or hourly data. This finer resolution could allow trades to be made before the market reacts to certain news data that has an impact on prices within a few hours. A main drawback is the computational expense associated with training models using a much larger data set. The walk-forward testing process took approximately ten hours for each test (of which there were fourteen in total, one for each model type and future prediction length), despite using a high-end GPU. Testing the models using minute data is likely to take over a week for each test.

Additionally, aggregating news data hourly would decrease the number of news records per sample compared aggregating daily. This could have the effect of increasing the amount of noise in the data.

An improvement that could be made to the trading strategies is to assign the size of each position using a function of the confidence value instead of a fixed dollar amount per trade. This may have similar effects to using the Kelly Formula, as described by Wu et al (2015), for position sizing.

Appendix

Training data variable names: `open`, `high`, `low`, `close`, `volume`, `market_cap`,

`news_body_sentiment_general`, `news_preview_sentiment_general`,

`news_title_sentiment_general`, `reddit_comment_body_sentiment_general`,

`reddit_submission_body_sentiment_general`, `reddit_submission_title_sentiment_general`,

`news_body_sentiment`, `news_preview_sentiment`, `news_title_sentiment`,

`reddit_comment_body_sentiment`, `reddit_submission_body_sentiment`,

`reddit_submission_title_sentiment`

The variable names ending in `_general` contain the average values of the sentiment scores for all cryptocurrencies in the data set, whereas the others are specific to the coin being tested.

Table 5: News Article Headlines

Headline	Sentiment Score
Spencer Bogart Maintains his $50k Bullish Target for Bitcoin	0.4019
Is Bitcoin ready for the Santa Rally: Off to $4400	0.3612
Now Is A Fantastic Time To Buy Bitcoin (BTC)	0.5574
Amazon Plays its Own Game With Enterprise Blockchain	0.4019
EOS Will Die in a Horrible Dumpster Fire in the Next Five Years	-0.886
Remember Atari? Now it's getting into blockchain gaming	0.0000
Western Union to add cryptos & Iran minister supports Blockchain	0.3612
Zurcoin Co-Founder Claims Exchanges Manipulating Prices	-0.3612
XRP Seems to Be the Top Candidate For the Bull Run in 2019	0.2023
FSMA Flags 14 Crypto Trading Platforms, as Crypto Scams Grow	-0.5859
Thousands of Unregulated Exchanges Generating Fake Volumes	-0.8893
Smart Dubai Office Wins Top Honors At Smart Cities Expo	0.9217
OKEX: Lousy Management Led to Losses	-0.8957

Table 6: Reddit Submission Titles

Title	Sentiment Score
Happy New Year to all guys who are making BCH the best coin for 2018!	0.9661
Honest question: why do a lot of people hate on TRX?	-0.1027
Bitcoin holding above $4,000, but market share is falling	0.2263
Bitcoin is dead, Bitcoin Cash is dead, mining cryptos is dead.	-0.9313
The Biggest Problem for ICOs? In 2018, It Was Their Own Investors	-0.4019
JP Morgan CEO Jamie Dimon Says Global Recession Not Coming	-0.0688
Venezuela Decrees Some Taxes to Be Paid in Cryptocurrency	0.0000
Hexx investors dump after accusing Lead Dev of exit scam	-0.9287
Hey everyone, new to Siacoin but love what I've learned	0.9572
Can someone answer this for me as I am not smart enough to...	-0.3089
I hope none of you are deep in aurora chain	0.4404
Russia to Buy Billions of Dollars Worth in BTC, Russian Economist Says	0.2263
SEC arrests and charged ICO creator with fraud	-0.9423
BTC is a great investment but it can also be a great gift	0.9788
Free chance to win $200 in BITCOIN Every hour!	0.9623

The plot on the next page shows how a drop in the sentiment of news article bodies preceded the drop in Bitcoin's closing price in mid-November. Additionally, the sentiment of Reddit submission bodies also seemed to drop before Bitcoin's close price did. Although this is just one example of changes in news sentiment occurring before changes in price, it gives an illustration of the sort of patterns that the model could learn to exploit.

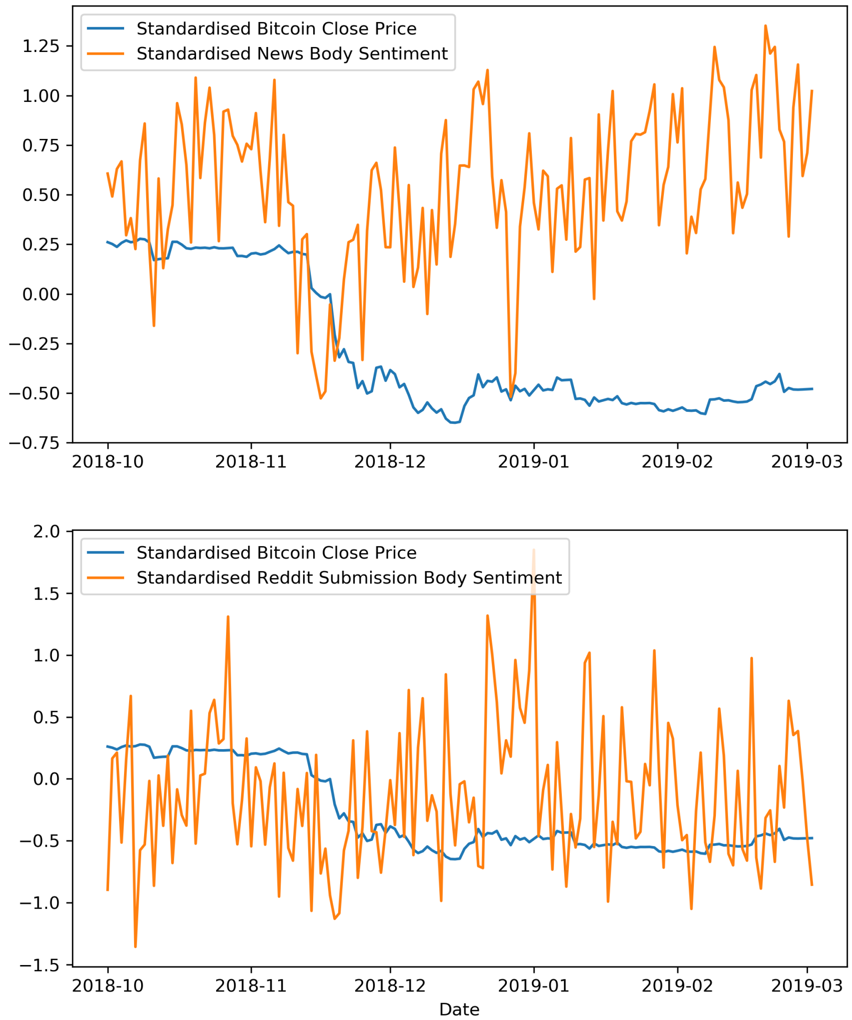

Figure 7: News Sentiment vs. Bitcoin Closing Price

The table below displays validation losses (not to be confused with the walk-forward testing losses) over various sequence lengths for the Buy/Sell model using targets five days in the future. At this stage in the research, it was not yet determined that three days was the optimal value, therefore, five was used as it performed well in preliminary tests.

Table 7: Sequence Length Comparison

Sequence Length	Categorical Cross-Entropy Loss
50	0.5950
75	0.5725
100	0.4023
125	0.5138
150	0.4531
175	0.4425
200	0.5138
225	0.5792
250	0.5693

Table 8: Buy/Sell Model Structure

Layer (type)	Output Shape	Parameter #
cu_dnnlstm_1	(None, 100, 128)	75776
dropout_1	(None, 100, 128)	0
batch_normalization_1	(None, 100, 128)	512
cu_dnnlstm_2	(None, 100, 128)	132096
dropout_2	(None, 100, 128)	0
batch_normalization_2	(None, 100, 128)	512
cu_dnnlstm_3	(None, 128)	132096
dropout_3	(None, 128)	0
batch_normalization_3	(None, 128)	512
dense_1	(None, 32)	4128
dense_2	(None, 2)	66

Total parameters to optimise: 345,698.

Technical analysis variable names: volume_adi, volume_obv, volume_obvm, volume_cmf, volume_fi, volume_em, volume_vpt, volume_nvi, volatility_atr, volatility_bbh, volatility_bbl, volatility_bbm, volatility_bbhi, volatility_bbli, volatility_kcc, volatility_kch, volatility_kcl, volatility_kchi, volatility_kcli, volatility_dch, volatility_dcl, volatility_dchi, volatility_dcli, trend_macd, trend_macd_signal, trend_macd_diff, trend_ema_fast, trend_ema_slow, trend_adx, trend_adx_pos, trend_adx_neg, trend_vortex_ind_pos, trend_vortex_ind_neg, trend_vortex_diff, trend_trix, trend_mass_index, trend_cci, trend_dpo, trend_kst, trend_kst_sig, trend_kst_diff, trend_ichimoku_a, trend_ichimoku_b, trend_visual_ichimoku_a, trend_visual_ichimoku_b, trend_aroon_up, trend_aroon_down, trend_aroon_ind, momentum_rsi, momentum_mfi, momentum_tsi, momentum_uo, momentum_stoch, momentum_stoch_signal, momentum_wr, momentum_ao, others_dr, others_dlr, others_cr

The plots below are of some popular TA indicators against Bitcoin's close price:

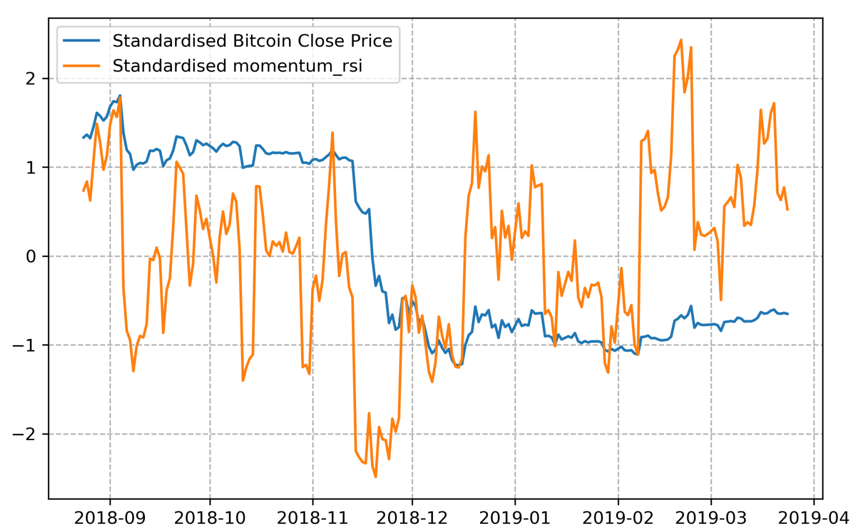

Figure 8: Relative Strength Index (RSI)

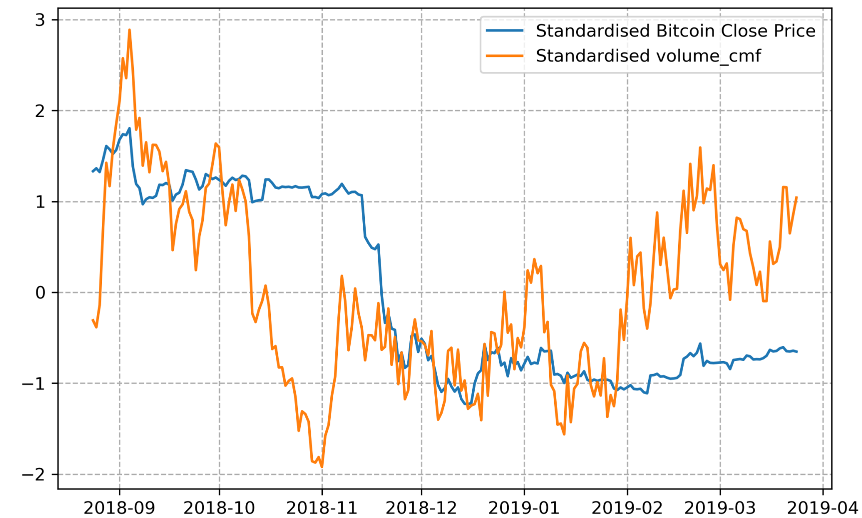

Figure 9: Chaikin Money Flow (CMF) Volume

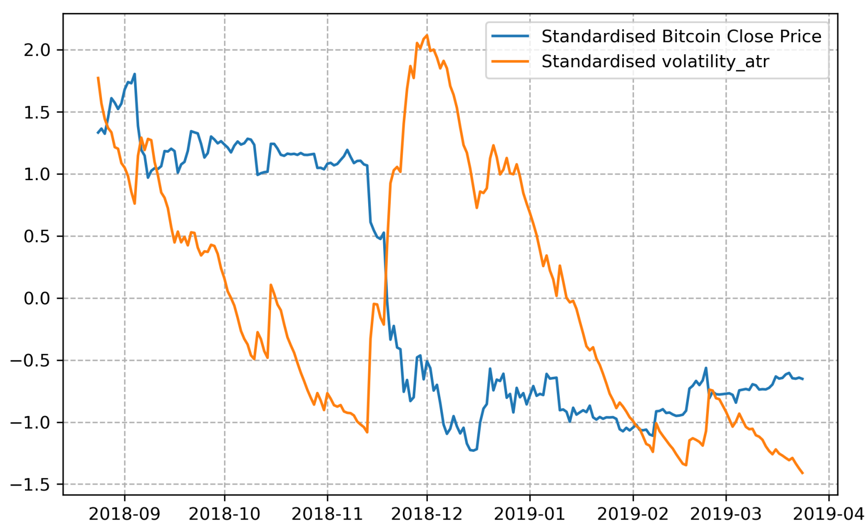

Figure 10: Average True Range (ATR) Volatility

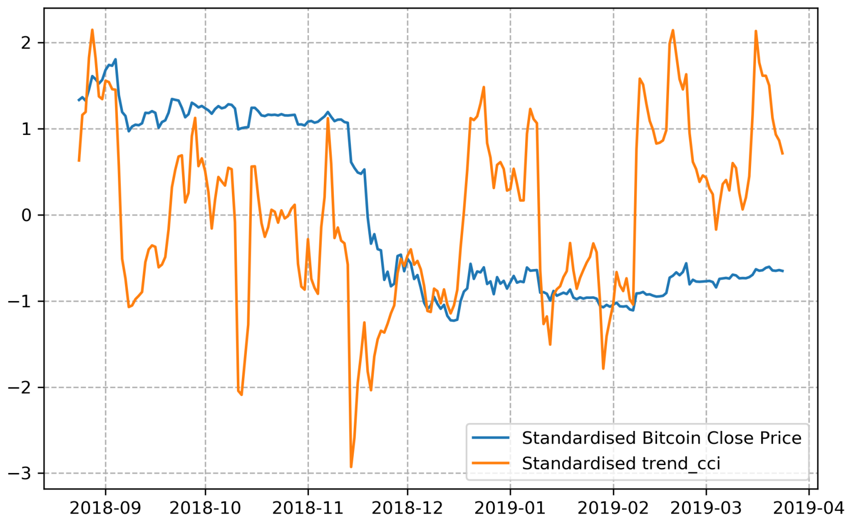

Figure 11: Commodity Channel Index (CCI)

Table 9: Buy/Sell/Hold + TA Model - Validation Results

Model	Accuracy	Categorical Cross-Entropy
1 Day	62.0%	0.7316
3 Day	61.1%	0.9113
5 Day	59.1%	0.6985
7 Day	52.2%	0.7178

Figure 12: Bitcoin Buy/Sell/Hold + TA trading strategy signals

Table 10: Buy/Sell/Hold + TA Model - Trading Strategy Results

Trading Strategy	Return on Trades	Profitable Trades	ROA * ECC	Robust Sharpe Ratio	Average Trade Span (Days)
1 Day	-5.9%	59.2%	-2.1	-0.07	45
3 Day	231.4%	83.2%	221.6	1.35	15
5 Day	277.0%	91.1%	271.5	1.73	14
7 Day	254.5%	95.4%	249.5	1.57	17

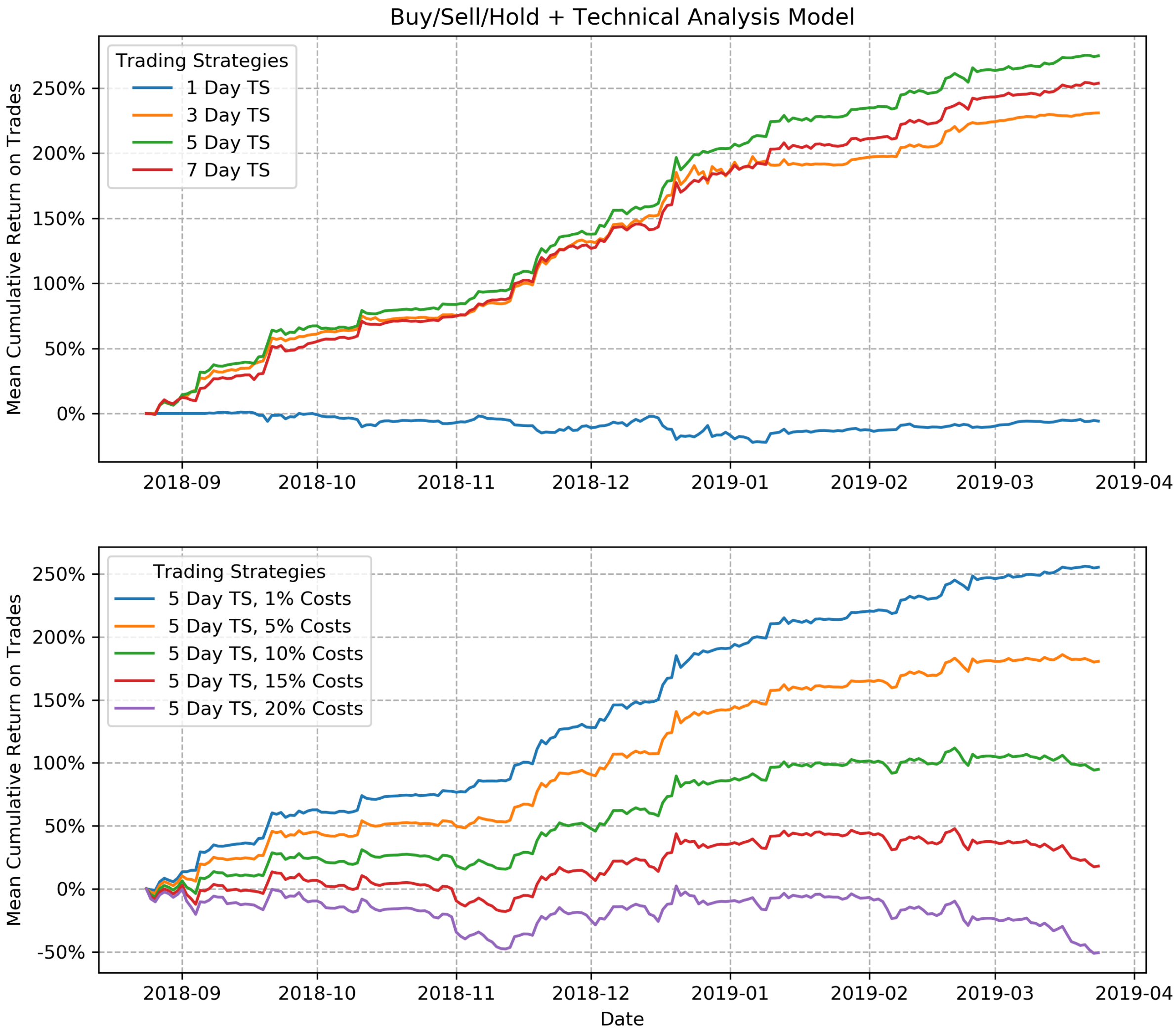

Figure 13: Mean Cumulative ROTs for the Buy/Sell/Hold + TA model

Figure 14 shows the cumulative market capitalisation of all cryptocurrencies in orange and the number of Reddit submissions and comments made each day for all of the 43 cryptocurrencies collected in blue. The values have been scaled between 0 and 1 to be displayed on the same graph. Since both curves appear to be correlated, the number of Reddit posts made each day could be used as an additional variable in the training data for the model. Of course, this variable will only be useful if the frequency of Reddit posts increases before the price of a cryptocurrency (which appears to be the case throughout much of 2017) or vice versa. Further research would need to be conducted to determine the impact of this data on the performance of the models.

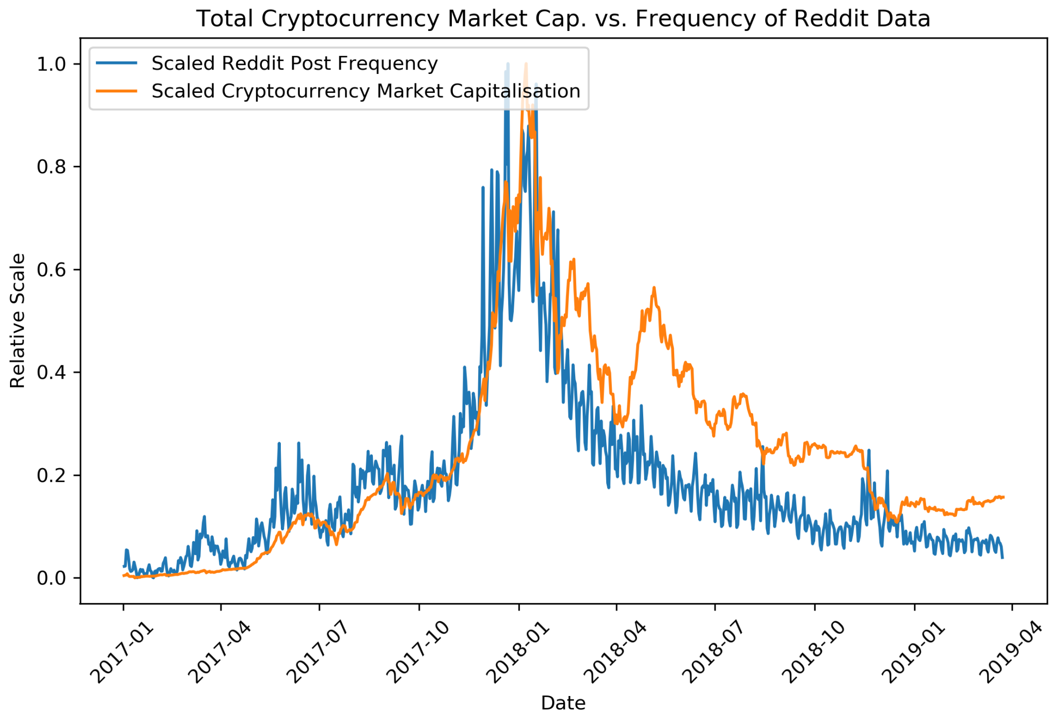

Figure 14: Total Market Capitalisation of all Cryptocurrencies vs. Reddit Post Frequency

Bibliography

[1] Al-Khazali, O., Elie, B. and Roubaud, D., 2018. The impact of positive and negative macroeconomic news surprises: Gold versus Bitcoin. Economics Bulletin, 38(1), pp.373-382.

[2] Darrat, A.F. and Zhong, M., 2000. On testing the random-walk hypothesis: a model-comparison approach. Financial Review, 35(3), pp.105-124.

[3] Li, T., Shin, D. and Wang, B., 2018. Cryptocurrency pump-and-dump schemes. Available at SSRN

[4] Hutto, C.J. and Gilbert, E., 2014, May. Vader: A parsimonious rule-based model for sentiment analysis of social media text. In Eighth international AAAI conference on weblogs and social media.

[5] Zhang, M., Zhang, Y. and Fu, G., 2016. Tweet sarcasm detection using deep neural network. In Proceedings of COLING 2016, The 26th International Conference on Computational Linguistics: Technical Papers (pp. 2449-2460).

[6]Schultze-Kraft, R. (2019). Don't be fooled — Deceptive Cryptocurrency Price Predictions Using Deep Learning. [online] Hacker Noon. Available at: https://hackernoon.com/bf27e4837151 [Accessed 16 May 2019].

[7] Huang, W., Lai, K.K., Nakamori, Y. and Wang, S., 2004. Forecasting foreign exchange rates with artificial neural networks: a review. International Journal of Information Technology & Decision Making, 3(01), pp.145-165.

[8] Lei, T., Zhang, Y. and Artzi, Y., 2017. Training RNNs as fast as CNNs. arXiv preprint arXiv:1709.02755.

[9] Ioffe, S. and Szegedy, C., 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167.

[10] Putchala, M.K., 2017. Deep learning approach for intrusion detection system (ids) in the internet of things (iot) network using gated recurrent neural networks (gru).

[11] Cesari, R. and Cremonini, D., 2003. Benchmarking, portfolio insurance and technical analysis: a Monte Carlo comparison of dynamic strategies of asset allocation. Journal of Economic Dynamics

and Control, 27(6), pp.987-1011.

[12] Fung, W. and Hsieh, D.A., 2004. Hedge fund benchmarks: A risk-based approach. Financial Analysts Journal, 60(5), pp.65-80.

[13] Metghalchi, M., Chang, Y.H. and Marcucci, J., 2008. Is the Swedish stock market efficient? Evidence from some simple trading rules. International Review of Financial Analysis, 17(3), pp.475-490.

[14] Szarvas, G., 2008. Hedge classification in biomedical texts with a weakly supervised selection of keywords. Proceedings of ACL-08: HLT, pp.281-289.

[15] Ko, Y. and Seo, J., 2000, July. Automatic text categorization by unsupervised learning. In Proceedings of the 18th conference on Computational linguistics-Volume 1 (pp. 453-459). Association for Computational Linguistics.

[16] Goldstein, M.A., Kumar, P. and Graves, F.C., 2014. Computerized and high frequency trading. Financial Review, 49(2), pp.177-202.

[17] Wu, M.E., Tsai, H.H., Tso, R. and Weng, C.Y., 2015, August. An adaptive kelly betting strategy for finite repeated games. In International Conference on Genetic and Evolutionary Computing (pp. 39-46). Springer, Cham.