Bitcoin Currency Fluctuation

Marius Kinderis[1], Marija Bezbradica[1] and Martin Crane[1]
[1]*School of Computing, Faculty of Engineering & Computing, Dublin City University, Glasnevin, Dublin 9, Ireland*
marius.kinderis3@mail.dcu.ie, {marija.bezbradica, martin.crane}@dcu.ie

Keywords: Bitcoin, Blockchain, Sentiment Analysis, NLP.

Abstract: Predicting currency prices remains a difficult endeavour. Investors are continually seeking new ways to extract meaningful information about the future direction of price changes. Recently, cryptocurrencies have attracted huge attention due to their unique way of transferring value as well as its value as a hedge. A method proposed in this project involves using data mining techniques: mining text documents such as news articles and tweets try to infer the relationship between information contained in such items and cryptocurrency price direction. The Long Short-Term Memory Recurrent Neural Network (LSTM RNN) assists in creating a hybrid model which comprises of sentiment analysis techniques, as well as a predictive machine learning model. The success of the model was evaluated within the context of predicting the direction of Bitcoin price changes. Findings reported here reveal that our system yields more accurate and real-time predictions of Bitcoin price fluctuations when compared to other existing models in the market.

1 INTRODUCTION

There are more than 900 cryptocurrencies currently available to invest in online; this number is consistently growing (Coinmarketcap.com, 2017). Of these cryptocurrencies, undoubtedly the most popular has been Bitcoin and it was also the first cryptocurrency in the market (Nakamoto, 2008). Several techniques have been used to give investors an advantage in predicting the price of Bitcoin at any given time. The strategies range from the Statistical (Chu et al., 2015) and econometric (Amjad and Shah, 2017) approaches to those that use machine learning to extract nonlinear relationships in the data (Żbikowski, 2016). In addition to technical analysis, traders can also gain necessary information about the market by extracting information. Such information may come from peers and news articles, which are often influenced by human emotion: whether market participants are feeling optimistic or pessimistic about the future state of the economy or a particular currency has an impact on its price (Georgoula et al., 2015).

This project aims to study the impact of human emotions on the price movements of the cryptocurrency, particularly Bitcoin, by analysing the effect of sentiment contained in Twitter posts (tweets) and news articles. This is done by implementing data mining techniques to collate tweets and scrape news articles relating to Bitcoin. Another objective of this project is to build a diverse trading model which gives traders an extra tool for predicting price direction using Natural Language Processing (NLP) techniques. This is in addition to technical analysis in the form of a Long Short-Term Recurrent Neural Network model (LSTM RNN).

This paper attempts to understand the factors that influence Bitcoin's popularity and uncover the variables that can affect its fluctuations from a financial perspective. The principal research question we address here is to what extent these Bitcoin currency fluctuations can be predicted. To standardize our main approach to the problem, we followed the example of previous studies (McNally, 2016) that utilized the Cross Industry Standard Process for Data Mining (CRISP-DM) model; this serves as a baseline for our project implementation.

CRISP-DM has been implemented elsewhere (Amjad and Shah, 2017) in financial applications. The methodologies used here are divided into four main parts: (i) exploration of the factors that influence Bitcoin's price fluctuation from the financial perspective as well as understanding of the field, (ii) data collection & preparation, (iii) predictive modelling and (iv) system deployment which includes keeping track of the changes occurring in the other related fields. This paper is organized as follows: Section 2 details the a review into the background literature to the project; Section 3 goes into System Design and Sec-

tion 4 its implementation; Section 5 describes the results obtained and Section 6 concludes the paper.

2 LITERATURE REVIEW

2.1 Creation of Bitcoin

In October 2008, Satoshi Nakamoto published the first paper (Nakamoto, 2008) on Bitcoin outlining its properties as a decentralized payment system. Shortly after, he published the first Bitcoin open-source software (GitHub/bitcoin, 2017) and the first units of the Bitcoin emerged. A growing community that actively uses Bitcoin has since been formed and Bitcoin's popularity has continued to increase as evidenced by over 14 million wallets registered worldwide (Park, 2017). Figure 1 highlights this increase in the number of Bitcoin users. This data is valuable as it has huge implications on how Bitcoin will fare in the market in future.

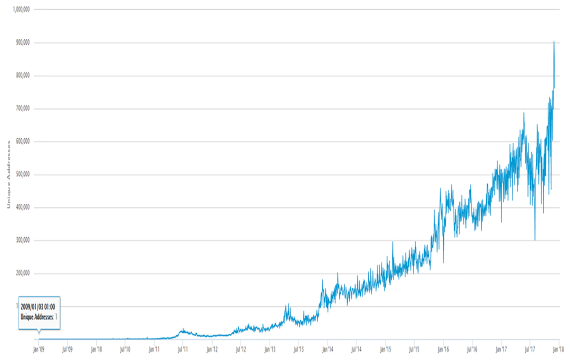

Figure 1: Total number of unique addresses registered on the Bitcoin blockchain from 2011-2017. (Blockchain.info, 2017)

2.2 Assessment/Commentaries

2.2.1 Technological Assessment

In contrast to other financial systems that require a third party (i.e. banks) to validate transactions, Bitcoin is based on blockchain technology (Badev and Chen, 2014) and is designed to make the transfer of value easier based on its peer-to-peer system.

Figure 2 shows how Bitcoin transactions are done by 4 users on the network where each trading is done directly from one user to another, hence the term "peer-to-peer".

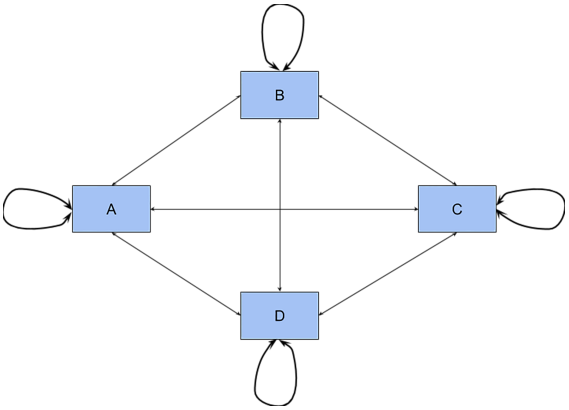

Figure 2: A peer-to-peer system showing four entities A, B, C and D transacting directly with one another and with themselves.

Blockchain technology operates as follows: payments on the user's network (Badev and Chen, 2014) are done by the chronological transactions recorded in a public ledger, called the *blockchain* (Figure 3).

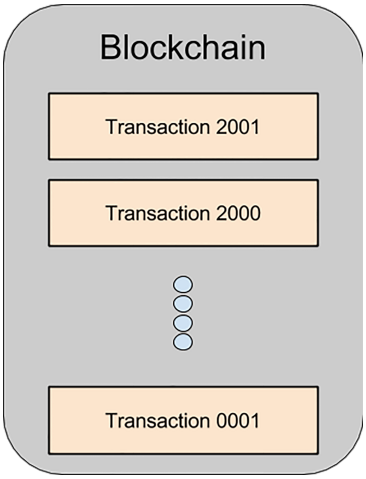

Figure 3: Transactions made with the Blockchain technology.

A monetary reward is reserved for recording Bitcoin transactions in the blockchain; and the users in this system compete for this by mining cryptographic problem to make records. Participants that mine Bitcoin are known as miners. Further, the affordability of Bitcoin transactions (a dollar per large transaction (Bitinfocharts.com, 2017)), makes the system very attractive to consumers. However, similar to all other currencies, Bitcoin units can be stolen, lost or confiscated. Hence, the risks associated with using Bitcoin units should not be underestimated by casual users.

It is also crucial to note that the whole technology relies mainly on cryptography, digital signatures and hashes to encode the transaction (the technical details will not be outlined here but may be found in sources such as (Badev and Chen, 2014)). Putting a trust in

such forms of encoding could lead to an implicit belief that they cannot be circumvented and this could be dangerous.

A recent study (Yli-Huumo et al., 2016) has provided us with insights into the current improvements of Blockchain technology through designing a map of raised and solved issues. This technology, however, has yet to tackle some issues[1]. Its security aspects have also been found to cause problems[2]. These issues will not be addressed further in this paper, but are worth noting as they must be addressed for the success and reliability of Bitcoin.

2.2.2 Financial Assessment

In (Dyhrberg, 2016), the authors explored the financial asset capabilities in hedging using Generalized AutoRegressive Conditional Heteroskedasticity (GARCH) models to highlight the fact that Bitcoin reacts relatively quickly to sentiment. They also found that its status in the market is in between a commodity and a currency as it combines some of the properties of both. The GARCH method has been used for portfolio management and risk analysis and is useful here for allowing an exploration of the relationships between Bitcoin and other more established commodities such as Gold, Copper, Cocoa etc. It is also useful as a statistical model to estimate the volatility of financial markets returns. It helps in determining which financial markets will provide higher returns and in forecasting the returns of current investments which is helpful in the budgeting process. For example, stock returns may look uniform for a number of years leading up to a financial crisis, but often a simple regression model would not detect this variation in volatility, the GARCH model will. There are many variations of this model, but (Dyhrberg, 2016) provides a new perspective on comparing Bitcoin with other financial markets.

Several attempts have been made to predict values of Bitcoin using blockchain network-based features (Greaves and Au, 2015; Madan et al., 2015). The consensus of such studies has been able to show up-down Bitcoin price fluctuations with a classification accuracy of roughly 55%; the model that exemplified the best accuracy using two hidden layers of neural network. One of the conclusions drawn from these studies was that only a limited amount of predictive information is embedded in the network features. This has proved that a better approach in predicting the price relies on the solid financial fundamentals. Moreover, it was agreed by the authors that the model that produced the highest accuracy on price prediction was LSTM RNN in (McNally, 2016) although Support Vector Machine (SVM), Random Forest and Binomial Generalized Linear Model (GLM) were previously used to explore how to efficiently trade with Bitcoin (Madan et al., 2015) at a smaller timestep with the accuracy of 58%.

(Georgoula et al., 2015) has explored the relationship between time-series and sentiment analysis using SVM algorithms to determine the factors that influence the price of Bitcoin. The authors found a positive correlation between Bitcoin price and Bitcoin users' sentiment and activity on Twitter regarding the cryptocurrency. Bitcoin price was also revealed to be correlated with the Euro-Dollar exchange rate, the number of Bitcoins in circulation and the level of the Standard and Poor's 500 (S&P500) stock market index (Us.pindices.com, 2017). This relationship is outdated, however, as the study was done in 2014-15, when the overall price of Bitcoin was generally going down. The price has changed since then and with changed S&P500 index. Several articles have also shown how social media apart from Twitter, blogs, articles and other sources of information impact on Bitcoin price (Matta et al., 2015; Mai et al., 2015) thus providing an opportunity to test other models on analysing Bitcoin-related data.

2.3 Algorithms used for predictive modelling

This section highlights the machine learning algorithms we explored in predictive modelling. These algorithms were used to classify the financial and sentimental data to give us a binary prediction helping us in the investment decision making process.

2.3.1 K-Nearest Neighbour

KNN is a supervised learning technique mostly used for classification and regression problems. It requires known data where usually the target variable is known beforehand. The algorithm doesn't have a training phase, unlike to other machine learning algorithms, the prediction of a test observation done based on the distance between observations.

The idea of this algorithm is to find K number of neighbours and given different classes assign a class to the unknown point (See example below):

[1] Latency, Throughput, Developer support, Size & Bandwidth issues.

[2] Currency exchanges and large mining pools are major targets of Distributed Denial of Service (DDoS) attack (News, 2017), various types of Bitcoin financial scams, Market-based centralization on mining power, and Duplicate key generation of elliptic curve cryptography (ECC) (Financemagnates.com, 2017).

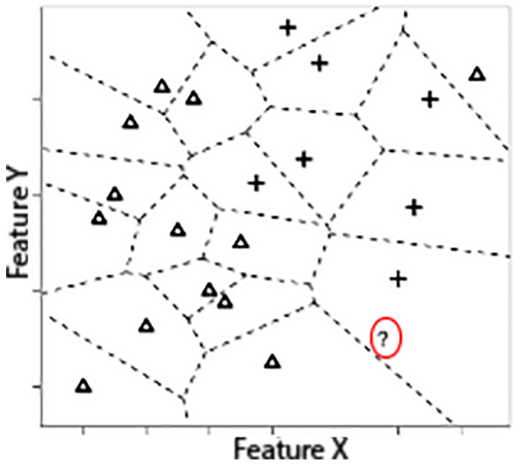

Figure 4: Voronoi partition by selecting K=1. The '?' belongs to '+'.

2.3.2 Logistic Regression

Logistic regression is a machine learning technique using mathematical formulas that converts an input interval [-∞, +∞] to [0, 1]. It is ideal for binary classification problems. Having said that, it requires training time.

The formula is given by:

$$\mathbb{M}_{\mathbf{w}}(\mathbf{d}) = \frac{1}{1 + e^{-\left(\sum_{j=0}^{b} \mathbf{w}[j]\phi_j(\mathbf{d})\right)}}$$

where **w** is the vector containing weights of the function and **d** the feature corresponding to the weight. To train this model we need a cost function:

$$J(\theta) = -\frac{1}{m}[\sum_{i=1}^{m} y^{(i)} \log h_\theta(x^{(i)}) + (1 - y^{(i)}) \log (1 - h_\theta(x^{(i)}))]$$

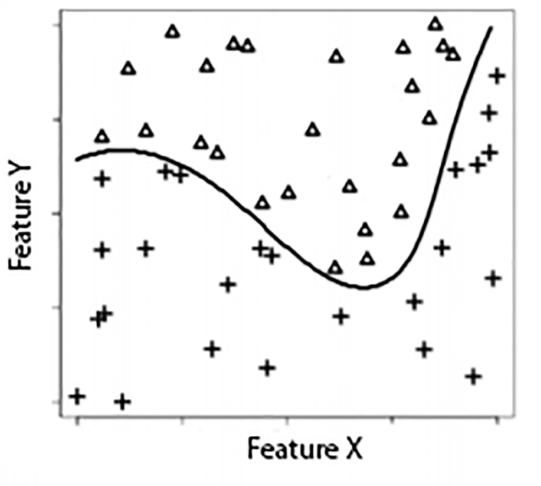

Figure 5: A Simple Example of Logistic Regression.

2.3.3 Classification Tree

The main idea of this machine learning algorithm is to split the dataset based on homogeneity of the data. Rigorous measures of impurity, based on computing proportion of the data that belong to a class, such as entropy or Gini index are used to quantify this homogeneity into Classification trees.

The usage of this algorithm is well explained at (Simafore.com, 2017). A simple example can be seen below in Figure 6:

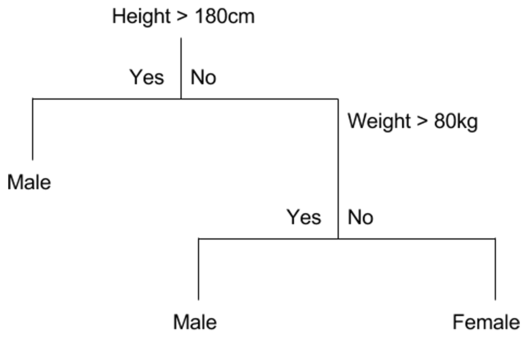

Figure 6: A Simple Example of a Classification Tree.

2.3.4 Support Vector Machine (SVM)

The main idea behind SVM is to find a hyperplane that separates different classes. A good example that illustrates this algorithm is in Figure 7. We can see that we can always find a plane that is capable of more or less separating two or more classes. The given example is shown in 2 dimensional space, but it can scale depending on the number of features.

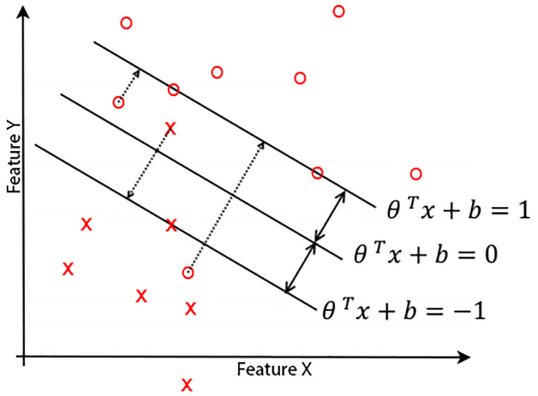

Figure 7: SVM with 2 features.

2.3.5 Gaussian Naive Bayes

This specific machine learning technique uses normal probability distributions and classifies the tested featured based on the probability. This method assumes that the data is normally distributed, as that way the classification process is more efficient.

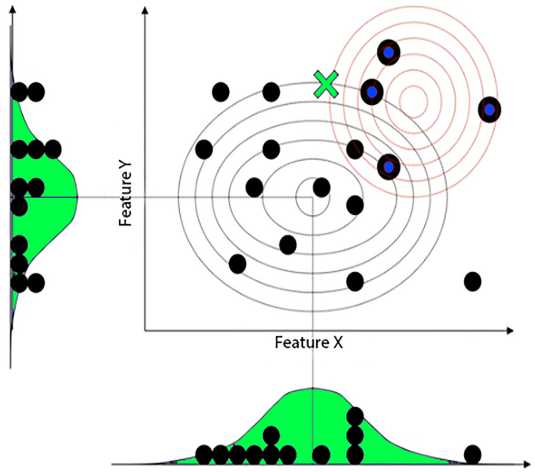

Figure 8: Representation of Gaussian Naive Bayes with 2 features.

2.3.6 Linear Discriminant Analysis

The main idea of Linear Discriminant Analysis (LDA) is that it reduces dimensions of a given classification task, focusing on maximizing the seperatibility among known categories. In practice, LDA creates a new axis by maximizing the distance between the means and by minimizing the variation. It then projects the data onto this new axis while reducing the dimensionality.

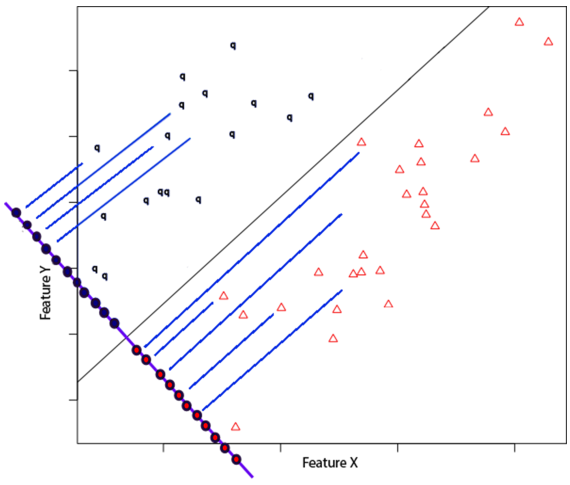

Figure 9: Representation of LDA with 2 features.

2.4 Likely Future Growth Areas

Blockchain is a huge source of information for Big Data (Chuen, 2015) and with its consistent growth the pace of technology has to keep up with increasing demands for the cryptography services required. The fact that a deep and organized market for high-quality Bitcoin-denominated bonds could emerge in the near future (Chuen, 2015), is adding a new point-of-view to the study of the price discovery process. At this point, it is still uncertain if liquidity, political and technological risks could influence interest rates on Bitcoin deposits, exchange rate stability or Bitcoin-denominated bonds. (Wang and Vergne, 2017) further highlight Bitcoin's investment potential citing opinion by Bernanke that it has potential to "promote a faster, more secure and more efficient payment system". The demand for Bitcoin for Bitcoin is also likely to be infuenced by the decision in December 2017 by US regulators to approve the trading of the Bitcoin derivatives. This decision has the potential to increase numbers trading Bitcoin, by allowing the Chicago Mercantile Exchange (CME) and the Chicago Board Options Futures Exchange (CBOFE) to offer contracts for futures of Bitcoin (DW, 2017).

Nevertheless, building a strong credibility appears to be the biggest challenge that Bitcoin faces in developing a viable bond market. Thus, analysing people's views about Bitcoin and the financial factors that influence its marketability can help us predict the next stage in the evolution of Bitcoin.

3 DESIGN OF SYSTEM

3.1 Concept

We first determined the main variables that influence a model prior to designing a system. Based on the studies done in (Georgoula et al., 2015; Dyhrberg, 2016), we observed that the two main categories that influence fluctuations in the price of Bitcoin are Finance and Sentiment.

In finance, the rise and fall of markets such as S&P500 and several commodities (Baur et al., 2017) influence the price of Bitcoin. In the same way, social media is known to play a very important role in Bitcoin price fluctuation (Matta et al., 2015). That is, if more and more people put their trust in Bitcoin by simply relying on others' positive social media posts on Bitcoin, its price will likely go up as the demand will increase; with the reverse is true when the posts about Bitcoin have a negative tone. To effectively understand these fluctuations, we explored and analysed

the most informative sources of sentiment concerning Bitcoin, using Twitter (Twitter.com, 2017) feeds and articles (CoinDesk, 2017) from www.coindesk.com to achieve this goal.

We then determined the most appropriate input and output data before finally building the system. This was performed by investigating correlations between different variables. Here we used Quandl.com's API of (Quandl.com, 2017), huge database of financial datasets to collate our financial data. We also carried out correlation checks on 152 different commodity prices and S&P500 index and found high positive correlations between the price of Iridium, Palladium, Aluminium, Cobalt and Random Length Lumber Futures and that of Bitcoin.

According to (Simafore.com, 2017), Iridium, Palladium and Cobalt are some of the rarest elements on Earth. Their annual production is low which makes them expensive. Their market values tend to surge fast due to a high demand for their properties and applications in scientific fields. On the other hand, Lumber and Aluminium are common commodities but also highly in-demand explaining their high and positive correlation with Bitcoin.

Finally, to streamline the complex steps of collecting and preparing text-related data, we generated an automated pipeline that scraped articles and tweets directly from the Twitter website.

3.2 System Architecture

Using our large collection of finance and sentiment data inputs, we reduced our objective to applying a classical binary classification approach that predicted the daily direction of Bitcoins currency (whether upward or downward). Each sentiment source, tweet and article on a given day, had three polarity variables: positive, negative and neutral as well as subjectivity variable, while every financial source had a single variable.

To qualify a sentiment, Natural Language Processing was performed with the TextBlob (Textblob.readthedocs.io, 2017) library by extracting an informative set of data from the text, giving us an overall sentiment indicator of the day, such served as the predictor to the next days' sentiment about Bitcoin.

Upon determining the values that describe the sentiment of different days and the financial predictions for the next day, we further used different classification algorithms to predict the direction of Bitcoin's price.

In summary, our system was built by combining multiple models. Predicted prices were identified by analysing financial data from Quandl's API (Quandl.com, 2017) using LSTM RNN on the actual Bitcoin price. LSTM RNNs are very efficient in predicting time-series related data (Indera et al., 2017). The sentiment-related data were determined from posts/news collected from Twitter (Twitter.com, 2017) and www.coindesk.com (CoinDesk, 2017) by implementing NLP. Classification algorithms were then used to group these two major sets of data into several factors that influence the (upward or downward) direction of Bitcoin price (Figure 9).

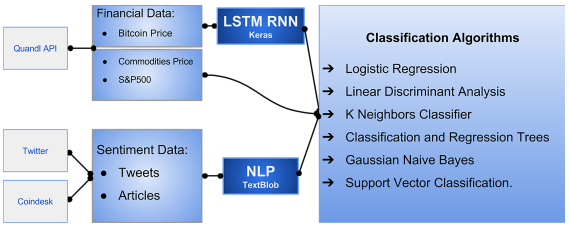

Figure 10: Design of the system architecture.

4 IMPLEMENTATION OF SYSTEM

We implemented our system using several high-level Python modules listed in the code uploaded in DCU gitlab (GitLab/kinderm3, 2017).

4.1 Dataset choice

To implement our idea exactly, we first decided which data sets and time-step to use. The financial data was only available on a daily basis for commercial reasons. Thus, we were only able to work with the time-step of one day. We decided to set the opening price of the day as the price used for all our variables. This time coincide with 9:30 am UTC-05:00.

4.2 Data Collection

Following the collection of substantial amount of financial data from Quandl's API, we were confronted a dilemma of efficiently collating Bitcoin-related tweets and articles necessary for our sentiment analysis. Thus, we created multiple bots to accomplish this task. Several machines and servers were used to execute those bots. We used Selenium (Selenium-python.readthedocs.io, 2017) as a crawler to emulate a real browser. This is essential to get an Ajax (Seguetech.com, 2017) type of data stream from Twitter. The method involved scrolling down and loading as many tweets as possible ranked by the

popularity for a specific date. Despite a large collection of an average of 6000 tweets per specific date, our resources were limited by the RAM capacity of the machine as all of the tweets with the webpage had to be preloaded before the storing process as we used Mozilla Firefox as our main web browser.

The same method was applied for collecting articles, the difference was that instead of scrolling down, we had to click on the page that brings us to the previous article until the very end. Taken together, we were able to collect 8620 articles and over 7,000,000 tweets dating from 2013. Even though multiple machines were used throughout this stage, the whole data collection process took a full one month.

4.3 Data preparation

4.3.1 Financial data preparation

The preparation of our collected financial data was straightforward: the missing values were supplemented with the previous date since commodity financial markets are not open during the weekends. Financial input data with correlation coefficients (r) beyond the +/-0.7 threshold (-0.7 < r < +0.7) were deemed insignificant and were hence discarded. As discussed in the previous section, only Iridium, Palladium, Aluminium, Cobalt, Random Length Lumber Futures, S&P500 and Bitstamp prices remained.

4.3.2 Text preparation

The tweets, on the other hand, had to be stripped of emoticons, numbers, hashtags (#), '@', '\n', etc. leaving only words for downstream analysis. Duplicated tweets or retweets were also removed. Running this process brought the tweet countdown to approximately 5,000,000. Apart from eliminating the numerical contents, no other pre-processing steps were done with Bitcoin-related articles.

4.3.3 Natural Language Processing on textual data

This stage involved conversion of our text data to numerical values using Natural Language Processing in tandem with a Python library called TextBlob (Textblob.readthedocs.io, 2017). This module returns the polarity of a text, which can either be positive (>0), negative (<0) or neutral (=0). This module also puts value of subjectivity which can be measured by the number in the interval [0, 1]; 0 corresponds to an objective statement and 1 to a subjective statement. TextBlob (Textblob.readthedocs.io, 2017) was also used to calculate the number of positive, negative and neutral texts (tweets or articles) of the day with a degree of subjectivity and then we output the percentage of each one of them. This gave 8 variables for both Twitter and Bitcoin-related article data outputs. These variables were consequently used to obtain the best predictions for our Bitcoin price movement which was explored in the second phase of the modeling process.

4.4 Modelling

The modelling process was divided into two stages: (1) building an appropriate price predictive model using LSTM RNN that is capable of adapting to the movement of the prices, and (2) resolving the binary-classified problem given multiple inputs.

4.4.1 First Phase

For the first stage we utilized LSTM RNNs. Our main goal was to predict the direction, not the actual price. Thus, we trained our model onto getting the shape of the actual Bitcoin price fluctuation using Keras (Keras.io, 2017) with tensorflow (TensorFlow.org, 2017) as the back-end, which has a built-in LSTM network off the shelf.

Tensorflow (TensorFlow.org, 2017) is an open-source software library used to create machine learning models such as Neural Networks. This tool can be easily deployed on Google's platforms and it greatly scalable.

Keras (Keras.io, 2017) is a high level programming library that utilizes the tools provided by Tensorflow (TensorFlow.org, 2017) albeit in a faster and more efficient way giving us the opportunity to readily create networks.

To effectively predict the direction of Bitcoin price, we first had to differentiate the series and render them stationary. This step was essential as this gave us a series independent of time, de-trended and without seasonality as shown in Figure 10.

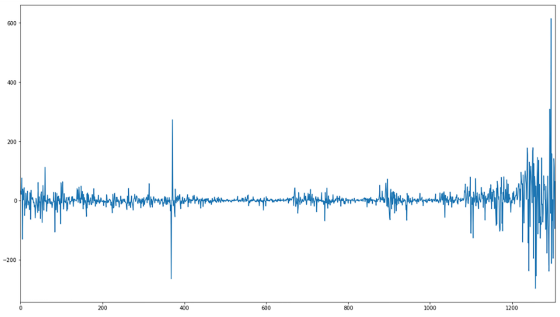

Figure 11: Stationary time-series used as an input for the LSTM.

To train the LSTM model, we tried to minimize the loss represented by the mean absolute error, (Figure 12) and improve the prediction accuracy (Figure 11). By doing so, we managed to make our LSTM RNN produce the same behaviour as the actual Bitcoin price.

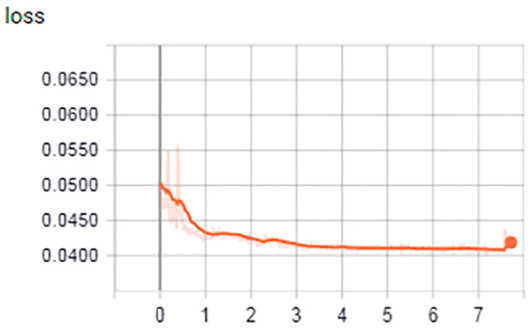

Figure 12: Mean absolute error of LSTM RNN.

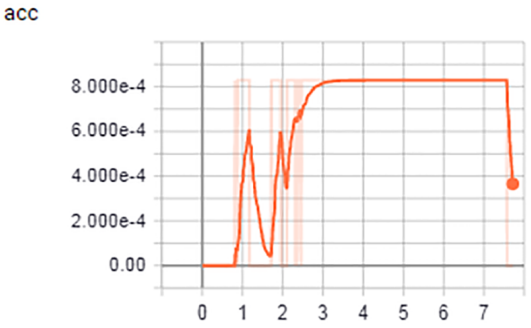

Figure 13: Accuracy of LSTM RNN.

4.4.1.1 Training setup

To optimize our model, we identified the highest number of neurons to use while selecting the best optimizer. We further performed several tests on our model with a high processing capacity. For computation purposes, we used Google cloud services, by setting up an instance with multiple GPUs (Graphics Processing Units) which considerably accelerated our testing flow by at least 10 times. To streamline the process, we used the following built-in optimizers with their default parameters: Stochastic gradient descent (Keras.io, 2017), RMSProp (Keras.io, 2017), Adagrad (Duchi et al., 2011), Adadelta (Keras.io, 2017; Zeiler, 2012), Adam (Kingma and Ba, 2014), Adamax (Keras.io, 2017), Nesterov Adam (Dozat, 2016). Given a favorable set of results, we opted not to create creating a custom optimizer for this task.

A suitable procedure to determine the best optimizer and the number of neurons needed was to loop through the list of optimizers and test different neuron configurations. Our results showed that the best optimizer for predicting the upcoming price is Adam (Kingma and Ba, 2014). By progressively increasing the number of neurons we were able to minimize the loss (mean absolute error) as an example shown in Figure 13. The number of neurons that were tested in the Figure 13 are: 40, 50, 60, 70 and 80 represented by Dark Blue, Dark Red, Light Blue, Bright Pink and Dark green, respectively. It was clear that in order to avoid a local minima, it was necessary to gradually increase our training time, reinitialise the learning rate. Additionally, to refrain from overfitting our model, we used a dropout layer with the rate of 0.2.

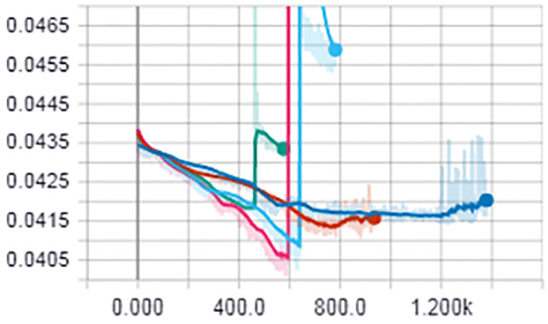

Figure 14: Mean absolute errors while training the LSTM different configurations (40 to 80 neurons) and ADAM optimizer.

We also ensured that on every reduction of loss (mean absolute error), the current state of the model was saved and that the error was as low as possible so the training process continued from where it left off. For this we used the built-in callback function to constantly save our model on each loss reduction. For the final layer of LSTM configuration, we used 60 neurons, adam as an optimizer and trained our model for 5000 epochs.

The training stage was initialized by setting the learning rate and hidden state/variables to random values using default parameters of Keras. Further, with callback functions embedded in the Keras library, we established that each instance **that** the mean absolute error function reaches a new minimum, the entire model is automatically saved. This method ensured the full optimization of LSTM layer.

The process was repeated multiple times with randomly generated initial steps. This is to ensure that any error does not get stranded in the local minima and that the results we get are robust. Approximately 70% of the input data was used for training while the remaining 30% was for testing. Six months' worth of data was needed to complete the second phase of the final model.

Overall, the LSTM excels in predicting the sequences which allowed us to configure a well adjusted one-layered network and input the sequence of our

stationary data capable of returning a prediction for the next day.

4.4.1.2 Training time

We trained our LSTM network for 3 days to obtain the right fluctuations shown in Figure 14, where the blue and yellow lines represent the actual and predicted values, respectively. Despite low accuracy, it was clear that our model was able to anticipate or predict the sudden shift of direction, conforming to our main goal. These results allowed us to compare the predicted against the actual price direction. The success rate of the direction prediction is on average 61.3% showing that not only LSTM RNN is capable of predicting the direction, but it is also adequate for learning the fluctuation.

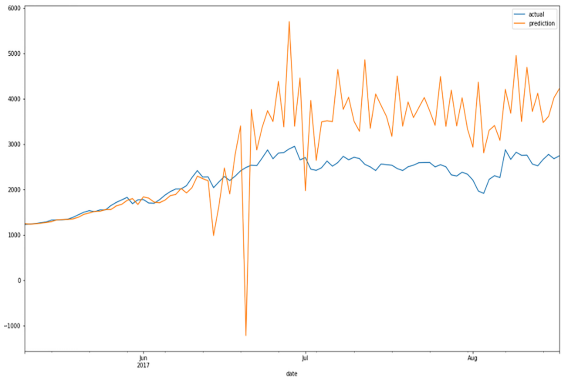

Figure 15: Mean absolute errors while training the LSTM different configurations (40 to 80 neurons) and ADAM optimizer.

4.4.2 Second phase

The second phase of our modelling largely involved classification of the collected data. This was initiated by splitting our dataset into training and testing portions, usually 0.7/0.3, respectively.

A total of 15 input data sets were used in this stage: the predicted price direction of Bitcoin (0 or 1) obtained from the initial phase of modeling, market prices of Iridium, Palladium, Aluminum, Cobalt and Random Length Lumber, S&P500 index, 8 sentiment variables from Bitcoin-related articles, and Twitter posts relevant to Bitcoin. Our main target variable was the actual Bitcoin price direction of the next day(0/1).

The financial input was then combined with the sentiment-based data and the future predicted price using LSTM RNN allowing for a highly accurate data set.

Multiple algorithms in the Scikit-learn (Scikit-learn.org, 2017) library were utilized to implement the classification process. These included Logistic Regression, Linear Discriminant Analysis, K-Nearest Neighbour, Classification and Regression Trees, Gaussian Naive Bayes and Support Vector Machine. In addition, we used Keras to generate customized neural networks to compare whether we can beat specialized algorithms, such as Support Vector Machine. The configuration involved 2 layers of 64 densely-connected neural networks, with a dropout of 0.5 each, targeting one output with Sigmoid activation. For this network the error measured was binary cross entropy and the optimizer used was RMSProp (Keras.io, 2017). We then identified which of these algorithms have highest accuracy rate to finalize our model.

5 EVALUATION OF SYSTEM

5.1 Result analysis

The following results were obtained for each classification algorithm after running the second phase of modeling process with 15 inputs and targeting the feature of the actual direction.

Table 1: Mean Accuracy score of the algorithms tested.

Classification Algorithms	Average Accuracy (%)
Custom NN	52.9
K-Nearest Neighbour	58.9
Classification Tree	60.6
Logistic Regression	62
Support Vector Machine	64.8
Gaussian Naive Bayes	67.4
Linear Discriminant Analysis	67.6

Table 1 presents that Gaussian Naive Bayes and Linear Discriminant Analysis algorithms rendered the most accurate predictions. Although it is noteworthy that average accuracy varies with the amount of input data and that all of the above algorithms can be used for predictive modeling.

Our analysis revealed that our model had higher accuracy in predicting the direction of Bitcoin price movements in contrast to those used in previous studies, which had the accuracy between 51-61% (Greaves and Au, 2015; Matta et al., 2015; McNally, 2016). This accounts for the fact that our model managed to incorporate user sentiments in the prediction process. This in turn allowed us to additionally assess the results produced by LSTM RNN and predict the direction of the price. We acknowledge, however, that further adjustments can be applied to refine our system.

Moreover, our model showed that sentiment does not have an immediate effect on the market or the currency. That is, the negative or positive user sentiments usually take some time to impact Bitcoin price and that a one day time-step works more appropriately for this type of model. Finally, treating Bitcoin as a commodity provided us with an opportunity to design a model that somehow reflected the reality of Bitcoin's behaviour as a rare commodity rather than a mere cryptocurrency.

6 CONCLUSION/FUTURE WORK

We created a model that is able to predict Bitcoin's price fluctuation. Initial observations revealed that changes in Bitcoin price movements were mainly influenced by the sentiment of the community through the social media and the real time events happening in the financial market.

This project allowed us to validate this hypothesis and gave us a better understanding on how Bitcoin behaves in the market. Consequently, the goal of our system was to combine both of these sources of information and integrate it into one model.

We built our SUT in two phases. First, LSTM RNN was modeled as a tool for price direction prediction and secondly, the additional data was added to improve the prediction of the fluctuation.

Overall, cryptocurrencies particularly Bitcoin are still in their infancy making it challenging for users to predict how they will evolve in the future. The model we developed in this paper, therefore, sheds light on this issue as it does not only predict the price fluctuations of Bitcoin but also has the potential to evaluate and anticipate the market behaviour of other available cryptocurrencies.

Potential avenues for further improvement of this work would be to use other social media platforms i.e. feeds from New York Times (nytimes.com, 2017), Bloomberg (bloomberg.com, 2017), etc. that will provide more accurate and real-time information for sentiment analysis. Additionally, including time-series models for predicting the price direction of Bitcoin, as well as evaluating the correlation with other cryptocurrencies will greatly improve the presentation of our current model.

REFERENCES

Amjad, M. and Shah, D. (2017). Trading bitcoin and online time series prediction. In *NIPS 2016 Time Series Workshop*, pages 1–15.

Badev, A. I. and Chen, M. (2014). Bitcoin: Technical background and data analysis.

Baur, D. G., Hong, K., and Lee, A. D. (2017). Bitcoin: Medium of exchange or speculative assets? *Journal of International Financial Markets, Institutions and Money*.

Bitinfocharts.com (2017). Bitcoin avg. transaction fee chart. https://bitinfocharts.com/comparison/bitcoin-transactionfees.html#3m.

Blockchain.info (2017). Bitcoin block explorer. https://blockchain.info/.

bloomberg.com (2017). Bloomberg. https://www.bloomberg.com/.

Chu, J., Nadarajah, S., and Chan, S. (2015). Statistical analysis of the exchange rate of bitcoin. *PloS one*, 10(7):e0133678.

Chuen, D. L. K. (2015). *Handbook of digital currency: Bitcoin, innovation, financial instruments, and big data*. Academic Press.

CoinDesk (2017). Coindesk. http://www.coindesk.com.

Coinmarketcap.com (2017). cryptocurrencies, coinmarketcap. https://coinmarketcap.com/all/views/all/.

Dozat, T. (2016). Incorporating nesterov momentum into adam.

Duchi, J., Hazan, E., and Singer, Y. (2011). Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul):2121–2159.

DW (2017). Us approves bitcoin derivatives trading on major exchanges 02.12.2017. http://www.dw.com/en/us-approves-bitcoin-derivatives-trading-on-major-exchanges/a-41626578.

Dyhrberg, A. H. (2016). Hedging capabilities of bitcoin. is it the virtual gold? *Finance Research Letters*, 16:139–144.

Financemagnates.com (2017). Blockchain warns of duplicate bitcoin addresses on android. www.financemagnates.com/cryptocurrency/news /blockchain-warns-of-duplicate-bitcoin-addresses-on-android.

Georgoula, I., Pournarakis, D., Bilanakos, C., Sotiropoulos, D. N., and Giaglis, G. M. (2015). Using time-series and sentiment analysis to detect the determinants of bitcoin prices.

GitHub/bitcoin (2017). Bitcoin. https://github.com/bitcoin/bitcoin.

GitLab/kinderm3 (2017). Source code. `http://gitlab.computing.dcu.ie/kinderm3/practicum_2017_Bitcoin_currency_fluctuation/tree/master`.

Greaves, A. and Au, B. (2015). Using the bitcoin transaction graph to predict the price of bitcoin.

Indera, N., Yassin, I., Zabidi, A., and Rizman, Z. (2017). Non-linear autoregressive with exogeneous

input (narx) bitcoin price prediction model using pso-optimized parameters and moving average technical indicators. *Journal of Fundamental and Applied Sciences*, 9(3S):791–808.

Keras.io (2017). Keras documentation. https://keras.io/.

Kingma, D. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Madan, I., Saluja, S., and Zhao, A. (2015). Automated bitcoin trading via machine learning algorithms.

Mai, F., Bai, Q., Shan, Z., Wang, X., and Chiang, R. (2015). From bitcoin to big coin: The impacts of social media on bitcoin performance. *SSRN Electronic Journal*.

Matta, M., Lunesu, I., and Marchesi, M. (2015). Bitcoin spread prediction using social and web search media. In *UMAP Workshops*.

McNally, S. (2016). *Predicting the price of Bitcoin using Machine Learning*. PhD thesis, Dublin, National College of Ireland.

Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system.

News, B. (2017). Major ddos attacks hit bitcoin.com - bitcoin news. https://news.bitcoin.com/ddos-attacks-bitcoin-com-uncensored-information/.

nytimes.com (2017). The new york times. https://www.nytimes.com/.

Park, H. K. (2017). How many people in the world own bitcoin or ethereum? https://hankyulpark.wordpress.com/2017/03/24/how-many-people-in-the-world-own-bitcoin-or-ethereum/.

Quandl.com (2017). Quandl's api. https://www.quandl.com/.

Scikit-learn.org (2017). Scikit-learn. http://scikit-learn.org/stable/.

Seguetech.com (2017). What is ajax and where is it used in technology? https://www.seguetech.com/ajax-technology/.

Selenium-python.readthedocs.io (2017). Selenium with python. http://selenium-python.readthedocs.io/index.html.

Simafore.com (2017). Simafore. http://www.simafore.com/blog/bid/62482/2-main-differences-between-classification-and-regression-trees.

TensorFlow.org (2017). Tensorflow. https://www.tensorflow.org/.

Textblob.readthedocs.io (2017). Textblob, textblob 0.13.0 documentation. http://textblob.readthedocs.io/en/dev/.

Twitter.com (2017). Twitter. https://twitter.com/.

Us.pindices.com (2017). S&p500 prices. http://us.spindices.com/indices/equity/sp-500.

Wang, S. and Vergne, J.-P. (2017). Correction: Buzz factor or innovation potential: What explains cryptocurrencies' returns? 12:e0177659.

Yli-Huumo, J., Ko, D., Choi, S., Park, S., and Smolander, K. (2016). Where is current research on blockchain technology?a systematic review. *PloS one*, 11(10):e0163477.

Żbikowski, K. (2016). Application of machine learning algorithms for bitcoin automated trading. In *Machine Intelligence and Big Data in Industry*, pages 161–168. Springer.

Zeiler, M. D. (2012). Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*.