Bitcoin price prediction using reddit data with VADER sentiment analysis

Filip Tobolewski
Department of Computer Science
University of Exeter, Exeter, UK
ft303@exeter.ac.uk

Supervised by: Dr Riccardo Di Clemente
Lecturer in Data Science
Department of Computer Science
University of Exeter, Exeter, UK
r.di-clemente@exeter.ac.uk

***Abstract*— In recent time, cryptocurrencies have seen a huge rise in awareness. Once, only discussed on niche technology-focused forums, they are rapidly becoming of interested to major financial institutions as an area to explore for portfolio allocation. In this paper we discuss possible investment strategies where we focus on the oldest of these currencies, Bitcoin** [1]**. We use VADER** [2] **to extract sentiment from the Reddit forum of r/cryptocurrencies. We use a LSTM sentiment-based model to predict 1-day log percentage returns over the 2013-2020 period. We then compare to a baseline model that uses only financial data. We find that a VADER sentiment model augmented with several crypto-specific phrases can outperform the baseline, with an RSME error of 4.13 for the sentiment-based model compared to 6.08 for financial data only LSTM model.**

Keywords—Cryptocurrency, Bitcoin, sentiment analysis, finance, reddit, VADER, long short-term memory.

I. INTRODUCTION

For thousands of years, one of the most important societal structures has been the presence of money [3]. Usually defined as a store of value, a unit of account and a medium of exchange. We may be adding a new description to its' definition: digital [4]. With the release of the original Bitcoin whitepaper in 2009, a small and quiet revolution started taking place. With the backdrop of social, political, and economic upheaval of the 2007-08 financial crisis, few took notice of this 9-page masterpiece in cryptography. With the described hash-based proof-of-work method to tackle the double-spending problem that plagued earlier attempts at creating a truly decentralized digital currency, the first successful electronic currency was created.

One of the main motivating factors is the impossibility of the task at hand. Working on extremely complex systems that cannot be solved in an exact way means that you are always attempting to approximate the truth with limited information. There is a fundamental beauty in attempting to model complex beings like people and how they act in an even more complex system like the financial markets. What we see, especially in the cryptocurrency markets, is human irrationality laid bare. With the knowledge that approaches from the fields of mathematics and physics were successful in mathematising the stock market for Jim Simons [23], we wish to make our small contribution to turning finance into a science.

Financial investment strategies based on sentiment have a relatively long history in the stock market. One reason to focus on new approaches is the competitive nature of the financial industry. Opportunities are very time sensitive and when one is engaged in investing other people's money, such as a pension fund, there is a fiduciary responsibility to ensure the clients investment is not exposed to unnecessary risk. As such, many approaches in sentiment-based investing have been taken. There are older existing strategies such as taking surveys of individuals who are invested in the stock market [5] which show that behavior of investors can be estimated. More modern approaches can succeed by improving the performance of a portfolio by around 2% by using social media sentiment for stock movement prediction [6].

As there are existing strategies that can inform a sentiment based strategy, it is also important to discuss the focus on cryptocurrency. During the last 12 year, cryptocurrency has grown from a little known idea to an ecosystem with a market capitalisation of aroun 2 trillion dollars [7]. There are new investing opportunities that can be used to generate returns for investors. Due to the relatively young age of this market, it may still me in the price discovery period which means that returns can increase still.

Another reason to focus on an investment strategy based on sentiment is due to how sensitive cryptocurrency is to the changing opinions of its' users. It is not uncommon for prices to swing wildly in both directions over a relatively short period of time. The potential returns can be quickly be wiped out. Creating a better, statistically sound machine learning model could account for the fallability of any one individuals' investment position. By treating social media comments as 'guess' about the true value of an asset we can use the law or large numbers and 'wisdom of the crowds' phenomena to our advantage. This effect has the following structure: suppose we wish to estimate some measure of interest, as we increase the number of informed estimates, the aggergate of all these estimates approaches the real value. This means that a well informed population which has 'skin in the game' [12] is likely to approximate the true value for the measure we are interested in. This of course relies on the assumption that comments represent honest, relevant opinions and refer to the subject we are interested in, namely future price estimates of cryptocurrencies.

Cryptocurrencies are a new area of portfolio allocation that most professional managers as well as retail traders will have to famialrise themselves with. Therefore it is important that new avenues of research are taken to optimise future investement strategies. This is what we hope to contribute towards.

II. Aims & objectives

The aim of this project is to explore if a machine learning model that is informed with sentiment can outperform a model that relies only on financial data. If a method of outperforming more traditional financial models could be found, this may suggest that social sentiment is not priced into the current market price and this information could be used to build a successful trading strategy. We can have several success criteria to measure the performance of a model, several of which we will discuss in section IV and V.

The reason in selecting VADER as the tool for sentiment extraction is the methodology behind it. VADER is a rule-based lexicon that used 10 independent raters in assigning sentiment to words and takes the average score of those 10 values to rate the sentiment of the word. There are 4 metrics it uses, where we are interested in the 'compound sentiment score'. The reason for selecting this metric is due to it being a unidimensional measure of sentiment which is a weighted, normalized composite score. Using this methodology outperforms any individual rater on a large text corpus sentiment classification task. We wish to explore if this method can be further extended with a 'wisdom of the crowds' [10] approach to sentiment-based investing [9]. By using VADER on a cryptocurrency forum to extract sentiment, we are treating the sentiment as having some information about the true future value of a particular cryptocurrency, with our focus being on Bitcoin. Therefore, if a method of sentiment classification of 10 raters can outperform individual human raters, it is interesting to explore the possibility for this method to extend to thousands of investors speculating on assets which are extremely susceptible to investor sentiment.

We will explore 3 different machine learning models Linear Regression, ARIMA and Long-Short Term Memory Neural Networks (LSTM). After initial results gathering, we will explore the most promising of these models in more depth. We will see if this model provides robust results across different methods for sentiment analysis and various financial targets such as price and returns. We will then explore if we can further incorporate economic and social features. We will use the Pushshift [11] API to collect the initial data. Exploratory analysis and feature engineering will be completed mostly with the NumPy, Pandas and MatPlotLib libraries. Completing the machine learning portion of the project will utilize SciPy and LSTM models will be build using Keras.

The aims are outlined, and we discussed the path we will take. We hope to meet the objective of building a machine learning model that uses a sentiment-based investment strategy. We will see if exploring this avenue of research could lead to fruitful results, as there is a potential 2 trillion-dollar market to explore for financial opportunities, a market of this size so dependent on sentiment has never existed in human history.

III. Background

Financial investing has been a very active area of research. From behavioral economics to quantitative analysis, investors look to every discipline for an advantage in the market. The cryptocurrency market is no different except for the short time that it has existed. The research is a product of it's time. As some of the most exciting scientific developments have offered in artificial intelligence and machine earning, so too has the research focused on utilizing these methods to make progress.

A time series analysis by Wooley et al [8] showed promising results. Being able to use reddit sentiment to build 112 time series features to guess the direction of bitcoin and etherium with around 73% accuracy. [13] Shows us a method utilising a simple evolutionary model model which can estimate the time a cryptocurrency will hold in market capitalisation rankings, with ranks 2-6 occupied by an average of 12 weeks. [14] Gives us a descriptive structure of cryptocurrency development, where it is possible to correlate cryptocurrency returns with github code developer contributions to the crypto projects. A network of developers is uncovered which work on multiple projects and the returns of these projects tend to track eachother.

Several LSTM approaches are also found in the literature. [15] Describes a RNN and LSTM approach to bitcoin price prediction. They achive a 52.78% accuracy in guessing price direction although they use 50-100 epochs and 2 hidden layers to ahcive this, until 1 and 10 for our model. In [16] We see an LSTM applied to blockchain statistics to predict the bitcoin futures market, which succeeds in reducting the RMSE by around 46% in 30-day return prediction with blockchain statistics compared to baseline. We also have an LSTM and ARIMA approach by [17] which compares the two and after more training, the LSTM model can outpreform ARIMA. ARIMA also faced the problem of precision decresing as time went on.

A sentiment-based LSTM model with a very similar structure to our proposed solution is shown in [18]. It focuses on Telegram data and also augments the VADER lexicon with more crypto-related terms. The 5-day prediction for bitcoin performs above the baseline. Another high-level study shows that using bitcoin price parameters LSTM models have the lowest RMSE compared to other RNN-based models [19].

We will briefly state why we focus on LSTM models and the technical aspects that make them suitable for our aims and give an overview of their structure so as to make the paper self-contained as the specification mentions.

LSTM: Long short-term memory [20] neural networks are based on recurrent neural networks (RNN). RNN is a type of neural network that can use time series data as an input. They have the right architecture to tackle a problem like sentiment-based investing as information from previous days is very informative of what is likely to happen the next day. They go through a learning process on past data and update their prediction based on prior inputs. They differ from standard neural networks, as standard neural networks usually assume that the data we feed in and the prediction we wish to make are independent from each other. For RNN's the outputs not only depend on the input parameters but also on prior outputs. Using this sort of structure preserves some basic facts about time series financial data such as that yesterday's market movements are still somewhat influential on today's market movements. This is to say that price on day *n* and *n+1* are linked and price on *n+1* is highly dependent on the previous day.

LSTM's build on the useful structure of RNN's as a solution to the vanishing gradient problem [21]. The problem lies in the fact that the gradient decreases as it back propagates in time therefore as the gradient keeps getting smaller, it begins to have diminishing contributions to the learning rate. In simple words, RNN's will fail to update their predictions with previously seen patterns as the sequence length of data keeps increasing. LSTM's and RNN's use a hidden layer with a function, such as the hyperbolic tangent, to regulate the values between -1 and 1. The difference lies in the operations that take place in the LSTM's that allow it to forget or keep information it deems useful or not. This is done by the cell states and gates inside the LSTM. The information that is deemed useful can propagate through the entire sequence and be kept by the model. Therefore, information in earlier steps can still update the model many steps after it was first processed. The gates mentioned contain a sigmoid activation function that returns values between 0 and 1. Therefore, at least in theory, useful information should be multiplied by a number close to 1 and remain to update the model intelligently, and useless information should be forgotten by being multiplied by a number close to 0. Therefore, the output should be approximating the true value as the sequence goes on due to useful information updating the model. As RNN's and LSTM's take up entire chapters in various books to explain, I hope this high-level discussion of the theory behind it helps explain why the structure of LSTM's is suitable for financial data prediction.

In summary, as previous days of financial data and sentiment should hold some ground truth about future price and returns, the structure of LSTM's allows it to learn to interpret these sequential inputs, keep useful information, and output a value that closely approximates the true value of the price or returns.

IV. Experiment design & methods

The project was organized into distinct sections, we can split them into two components, Data Engineering (DE) and Data Science (DS). We began with exploring potential datasets for the project, we found an easy-to-use dataset on Kaggle which consisted of reddit comments from November 2017 to March 2018. We ran a small-scale experiment on this dataset which returned some promising results, such as that comments which mentioned words like "bitcoin" and "BTC" had a far higher positive sentiment compared to comments which lacked them. This was true until December 19th, 2017, the peak of the bitcoin bull run. At that point the sentiments started to become the same, with eventually with comments that did not discuss bitcoin having a far higher sentiment, please see figures C and D in the appendix for a graphical representation. It became apparent that there may be something interesting in this area therefore I moved towards expanding the scope of the project. The chart below briefly shows the work undertaken for the project under the two sections of DE and DS in order of completion:

1. DE	Use Pushshift to collect reddit comments from start of forum to May 31st 2021
2. DE	Select relevant data, clean text, feature building
3. DE	Use VADER to compute sentiment of comments, explore results and build features on sentiment
4. DE	Update VADER lexicon with cryptovader phrases, added several financial/crypto specific phrases not present by using synonyms or pairs
5. DE	Financial data collection, exploration, feature building
6. DE	Extracting 8 emotions from reddit comments with NRC lexicon
7. DS	Exploratory models with linear regression and ARIMA
8. DS	Building LSTM model for price prediction, returns based on financial and sentiment features.

1. DE, Initial data collection

We began with downloading all comments from the subreddit of r/cryptocurrency. The reason for choosing this subreddit instead of others such as r/bitcoin and r/Ethereum is to avoid echo chamber effects. The culture of the subreddit is very important to make sure the sentiment that we extract is useful, therefore using single cryptocurrency forums would add a bias that we cannot describe mathematically. The world of cryptocurrencies can be very tribal and heavily driven by the human stories and rivalries, such as an Etherium co-founder leaving the project to begin his own rival cryptocurrency, Cardano. Therefore to account for this tribalism, using a general forum which discusses many currencies may be the best compromise.

To avoid the reddit time limit API query limitations of 100 comments per 60 seconds [22], we used Pushshift to collect the historical data. As the servers can be offline, a check was introduced to ensure all shards of the API server were online before API query requests were made to ensure no data was missing. We collected around 11 million comments spanning 8 year, around 5GB of data in total. One drawback of Pushshift is that it is maintained for free by a single person, therefore more recent data from 2021 can at

times be missing and it can take several months to be updated. Therefore, we chose to take a longest unbroken chain of available comment data for sentiment analysis to avoid discontinuities in the inputs, as a sentiment based model must have a sentiment value for the entire period we wish to explore. Therefore we kept comments from 29/04/2013 to 01/03/2021, taking our comment total from 11 million to around 9 million. With that step 1.DE was completed.

2. DE, Feature building and data cleaning

For the following section of 2.DE, we undertook initial data exploration and cleaning. From all the available features reddit comments contain, we selected 4, These being 'Author', 'body', 'created_utc' and 'score'. The initial dataframe can be seen below.

	author	body	created_utc	score
0	TechnoMagik	I'm not sure how you eliminate spread.. If I a...	1368332818	1.0
1	mytwobitcents	fixed thanks	1368321753	2.0
2	sex_with_a_goat	The Spanish one is wrong, we don't use 'y' wit...	1368318717	2.0
3	sex_with_a_goat	You mean "criptomoneda".	1368318564	1.0
4	davidpbrown	Yes, Russian Trolls are the most obvious answer.	1368298185	2.0

Figure 1. Initial Dataframe collected

There are several issues with the data before we can engage in any meaningful sentiment analysis. I will briefly discuss one such issue in depth and omit discussions of other for the sake of brevity. The forum is structure so that a bot account with the author name as 'AutoModerator' replied to threads with several possible replies. These replies sometimes state forum rules, or why a posters' comment was deleted and in other instances the auto-reply consist of reasons for closing a discussion thread. In all these cases, the sentiment metric we use gives these auto-replies a weakly negative compound sentiment of -0.17 or a moderately positive score of 0.34. This means that when a very newsworthy even occurs we see an influx of new accounts, which receive a reply from the AutoModerator reminding them of the rules, which is moderately positive in the compound sentiment metric. This means that if we maintain these comments in the dataset we use for calculating sentiment, we can get a shift in the measured sentiment on days with significant news and new users signing up to the forum. But this shifted sentiment does not correspond to any forum members opinion of feeling, as these are simply autogenerated replies. Thus, these comments are removed to preserve the integrity of the extracted sentiment.

The cleaning occurred in the following order:

1.	Clean_removed_deleted function: Drop all comments where body == [removed] or [deleted], as this signifies moderator removal or author deletion of comment respectively.
2.	Remove_bots function: We found the most active bot accounts and if author == bot_list we drop the comment
3.	Clear_hyperlinks function: Removes hyperlinks and replaces with a whitespace

After these steps are taken in dropping comments and limiting the comments present to longest uninterrupted chain we are left with 7 million comments. One small comment to make is that we do not remove stopwords, at this stage, as the VADER package we use for sentiment analysis does this automatically. We then move onto using the 'created_utc' feature to create a function which takes in the UTC as an input and return the day, month and year so that we can later collect all comments and aggregate the sentiment by day. We then move onto extracting sentiment.

3. DE, Sentiment extraction

We begin section 3.DE by creating two functions. Using the VADER package we create a function get_sentiment which passes in the comment text into the inbuilt VADER sentiment score function called polarity_scores. This function is called for all comments in the get_sentiment_scores function, returns 4 scores where we only select for the compound sentiment metric. Finally, the result is inserted as a column in our dataframe. In the exploratory analysis we found that the distribution of the sentiment is not gaussian, therefore for the sampling of the sentiment we take the median as the sentiment value for the day. The reason being that the median will give us the sentiment where exactly half of the users are above and below this value, best approximating the overall mixed sentiment. Plots of distributions are available in the appendix.

4. DE, Expanding VADER lexicon

With section 4.DE we use the same function as in section 3.DE with the addition of an updated VADER lexicon. The original VADER lexicon does not capture some very specific language that is very important, when checking a comment such as "I am bearish on Bitcoin", VADER returns a compound score of 0. This is due to the limitation that VADER was created to be effective very general social media contexts therefore it does not preform as very in a very specific domain. The updated lexicon is created from the contributions of a GitHub user who extended the lexicon to include several crypto specific phrases. Some synonyms were used where available and where synonyms were difficult to find, we used polar opposite score scores for linked phrases. An example would be "bull market" and "bear market", which have the same score in the updated lexicon with different connotation (positive and negative, +2, -2 respectively) as it is not possible to assign a synonym.

Another issue at this point in the project become incorrectly assigned sentiments due to complex language structure of the sentence that VADER did not capture. Around the documentation stated threshold of -0.05 for negative sentiment and +0.05 positive sentiment, there are frequent misclassifications, we found some examples of comments given a +0.35 scores that to a human reader would be in the negative sentiment group. As a possible way to exclude misclassified comments, I created features where sentiment threshold for keeping a comment for consideration was increased to +0.25 and -0.25 compound sentiment score.

This will act as an experiment to see if the model would work better by excluding possible misclassified comments.

Our two different sentiment extraction methods and sentiment threshold mean that we will have the following structure when comparing models:

Baseline Model	Vader Sentiment ± 0.05	Crypto Vader Sentiment ± 0.05	Vader Sentiment ± 0.25	Crypto Vader Sentiment ± 0.25

We will compare the sentiment-based models with a baseline model based only on financial data. We can then also compare the performance of the different thresholds and the performance of added crypto-specific words.

5. DE, Financial data feature building

We then moved onto the final DE part, financial data feature engineering. The data is collected from coinmarketcap which aggregates the data from many major trading exchanges giving a good overview what is generally occurring with the cryptocurrency market. We build 1-day log returns based on the following equation:

$$log(P_t - P_{t-1})/P_t$$

Where P_t is the price at time t, and 30-day differenced returns as possible candidates to evaluate the model results. We focus on predicting close price and one-day returns, and one-month differenced returns. One reason to focus on the returns of an asset is that usually they approximate a gaussian distribution and can be normalized. With a metric such as price we cannot state a specific maximum value with great confidence, but one day returns are unlikely to lie very far away from the average. It is very unlikely our model would have to predict an extreme return of say -90% or of +400% on any given single day, but with price alone it may have to predict prices never seen before in the past, several times in a relatively short period of time, making this metric very difficult to predict.
We explore the possibility of models being macro-economically sensitive, with the macro in Bitcoin being the block reward size which is reduced by 50% roughly every 4 years. Our entire data sequence covers 3 of these bitcoin halving events which means there are 3 different supply-demand dynamics at play. We build further models based on the only fully available bitcoin halving cycle data, this being 2016-2020. This should act as a check to see if limiting the scope to these 4-year bitcoin cycles helps improve model performance.

Another reason not to put too much focus on the absolute price is the long-term trend for bitcoin is positive, from around $100 in 2013 to around $64000 in May 2021. We can tackle this issue by detrending, but a successful trading strategy does not have to rely on always going long on the asset. Bitcoin market has evolved in the last few years with the ability to use leverage as well as to go short or trade futures now open to everyone. Therefore, if our model predicts returns, we can attempt to play both sides of the trade depending on the predictions, but for this to be the case our model would have to preform well with both long and short predictions.

6. DE, Exploratory emotional analysis

We engaged in preliminary experiments to include emotions as inputs which was extracted using the NRC lexicon, unfortunately this was later deemed beyond the scope of the project. Some exploratory analysis and models are included in the submitted code, but due to time constraints the full implementation of this is left for a future extension to build upon this project.

7. DE, Initial model building and selection

At this point we stop work on any data engineering and focus on bringing all the features built together to explore potential candidate models. In the initial exploratory model building, I attempted 3 different models, these being Linear Regression, ARIMA and LSTM. Our initial expectations being that ARIMA would be a suitable candidate for exploration as it can take in inputs as exogenous and endogenous, such that the price or returns are informed by the change in sentiment. However, it quickly proved difficult to find a good method of implementation. I will briefly

Linear Regression

Linear regression models are very popular in the field of finance and at their most basic they are based on the equation for the line of best fit. They have a long history for being used to predict stock prices. It is possible to adapt this model to a different asset class such as cryptocurrencies. They calculate a relationship between various inputs and attempt to estimate the target output. For our specific case we predict price with the inputs as financial data and sentiment. This model did not lead to promising results and as it fundamentally relies on past data heavily there are other more suitable models for implementing the idea for this project we discussed in sections III. We can see the result of linear regression predicting 30 days in figure 2.

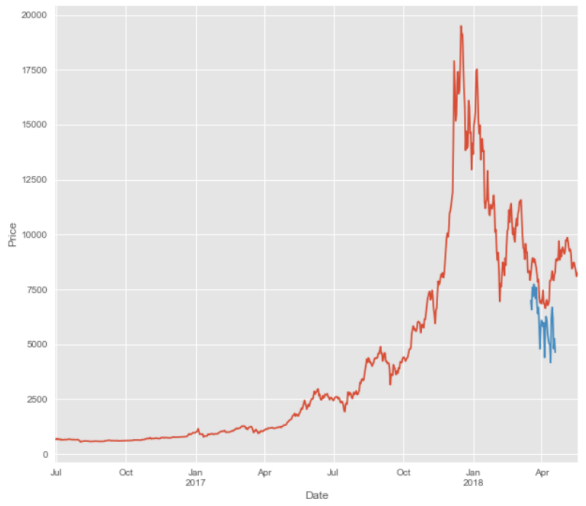

Figure 2. Linear regression model prediction for 30 days

ARIMA

An AutoRegressive Integrated Moving Average model takes in historical time series data from t = 0 to t = n-1 and attempts to make a prediction about the target variable's value at t = n. The model must be made stationary which we ensure by using differencing and then using the Dicky-Fuller test. The model can be split into 3 core components:

AR: The autoregressive part refers to the portion of the model where the predicted target variable value depends only on the past values, where the order *p* refers to the number of part values to consider that inform the prediction. This can be described as the linear regression part of the model.
MA: The moving average portion is dependent on the prediction depends on the lagged forecast errors, where the error terms are the errors from the autoregressive portion of the respective lags.
I: The integrated portion is the differencing that we do so that the series is stationary, therefore we work with the difference between values rather than values themselves.

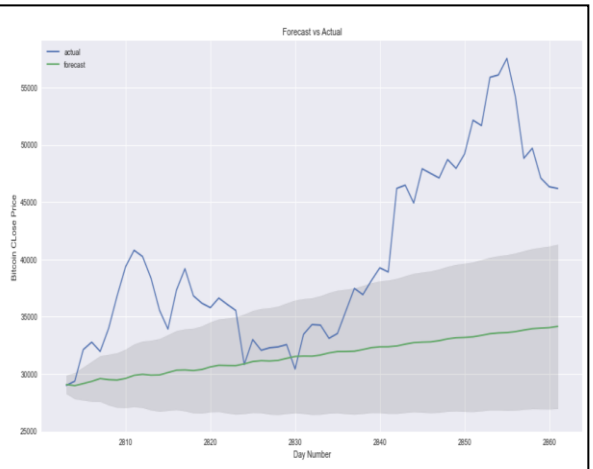

Figure 3. ARIMA model prediction for rapid price changes.

The results for an ARIMA model are highly dependent on past data inputs. In figure 3 we can see how the ARIMA model behaves when working on predicting a period of rapid price change, lower, upper bounds and price prediction remain rather stable while the actual price behaves very erratically. We can see this in figure 3.

Therefore, we move onto the most promising model, this being LSTM. As this model is so extensively used and explored, we will discuss it at length in the results section, as the discussion on linear regression and ARIMA serve the purpose of exploratory data science investigation and preliminary model building.

V. Results

The main focus of our results will be on our LSTM model and how it preforms compared to baseline models based only on financial data. In section II we discussed the theoretical underpinnings of the LSTM model and its' appropriate structure for the research question at hand. In section IV we stated why we focus on the LSTM model over linear regression and ARIMA. Therefore, we can begin presenting the results.

Initial tests

For the LSTM model we began by building a model based only on financial data as inputs and with predictions being metrics such as returns and price. After testing several parameters, the best performers seemed the correspond to the ones listed below in the parameter figure 4, these parameters are the same for all models displayed.

LSTM Parameter	Parameter Value
Look_back	60
LSTM_layers	32
Epochs	10
Batch_size	128

Figure 4, parameter settings in all models presented

Look_back: refers to the amount of sequential data the model considers for predicting the target variable. Setting it at 60 means that the model will consider 60 days' worth of data when attempting to predict the target value. Testing both 30 and 90 days resulted in similar outcomes to 60.
LSTM_layers: This parameter sets the number of layers in our LSTM network. Each of these hidden layer nodes has the gates to maintain or drop information that the model considers useful or useless. Tests at 64 layers resulted in slower but similar results.
Epochs: This parameter defines the number of time that the model will go through the training dataset. After 4-6 epochs the loss for training and validation sets usually asymptotes therefore, we keep the parameter at 10.
Batch_size: For the model to learn the patters we are interested in, we have to feed in data in batches and we tested 32, 64 and 128 as the batch sizes. The latter lead to the best performance.

The financial data (FD) feature in the figure 5 refers to the inclusion of the volume, market capitalization, open, close, high, and low prices. Volume refers to the amount of bitcoin traded in US dollar terms. Market capitalization is another classic metric to use which represents the amount of a financial asset available multiplied by the current spot price. The open, close, high, and low prices refer to price range as well as the price at market open and close.
These models serve the purpose of providing a useful baseline comparison to sentiment-based models. The comparison will display if including the extracted sentiment in the form that was described in section IV improves the performance of the models. We use the root mean square error to measure the performance of our models. The RMSE is a measure of the difference between the actual values we wish to predict and the predictions our model's output. Therefore, the smaller this value the better the prediction. The other measure we use is the mean absolute error, which is a measure of errors between paired observations, which is a arithmetic average of the absolute errors between the actual values and predicted values.

Features	Close price	Log 1-day Returns	30-day Returns
FD	x	x	x
Results			
RMSE Train	449.46	4.51	433.35
RMSE Test	2231.76	4.06	1158.19
MAE Train	190.24	2.79	179.34
MAE Test	1062.27	2.67	639.35

Figure 5. Baseline model results for financial data only.

The baseline models measure very different quantities therefore comparing them to each other will be fruitless. One interesting thing to note is that the 1-day returns model, based on a simple metric of the expected log returns if an investor were to buy 1 bitcoin today and sell it tomorrow, seems to preform very well when comparing the train and test sets. It is the only baseline model which preforms better in the testing set than training, with the other models performing far worse. We can see in figure 6 the graphical representation of performance in predicting the close price, when comparing to linear regression and ARIMA we can see that this model works better in predicting over the entire dataset.

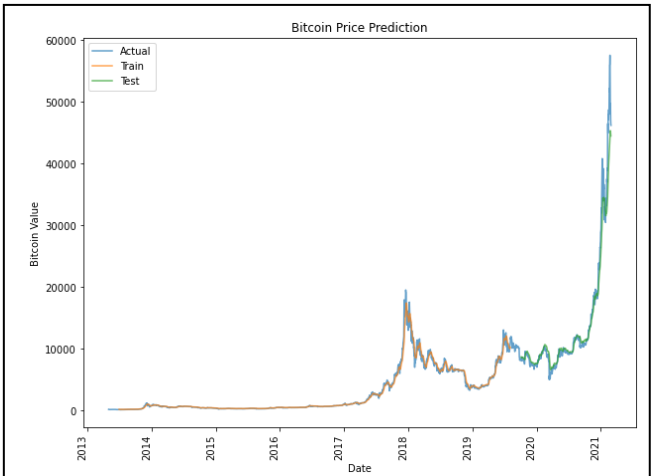

Figure 6. Baseline close price train and test prediction

We will briefly move onto the close price prediction and monthly return models before turning to the more successful and interesting 1-day returns model, explaining further experimental steps

Close price & sentiment.

Feats	Baseline	V0.05	VC0.05	V0.25	VC0.25
FD	x	x	x	x	x
NS		x	x	x	x
PS		x	x	x	x
Result					
RMSE Train	489.78	922.28	658.89	448.33	483.00
RMSE Test	3613.05	4086.63	3137.84	2667.57	2220.40
MAE Train	209.96	649.81	410.67	229.24	298.42
MAE Test	1918.56	2954.96	2431.82	1193.96	1410.97

Figure 7. Results for close price & sentiment model

With the baseline models in place, we can continue to build the first sentiment based models and explore their results. In figure 6 we see the experimental set up and results of the test. We have NS and PS representing the negative sentiment and positive sentiment respectively. V with either 0.05 or 0.25 represents the VADER sentiment value threshold that was passed as a feature into the model. VC present the augmented VADER lexicon with the addition of the crypto and financial specific lexicon sentiment with the threshold used to define positive and negative sentiment.

We can see that sentiment-based models preform slightly better than the baseline in terms of the RSME. This is an encouraging indication that there may be a method to include sentiment to better inform an investment model. In terms of the MAE the models roughly preform as well as the baseline. The only exception in both metrics is for the original extracted VADER sentiment with the threshold at the original definition while VC0.05 also. This suggests that having a higher threshold, more specific phrases in the lexicon, or both can improve the predicative power of our model.

30-day differenced returns model & sentiment.

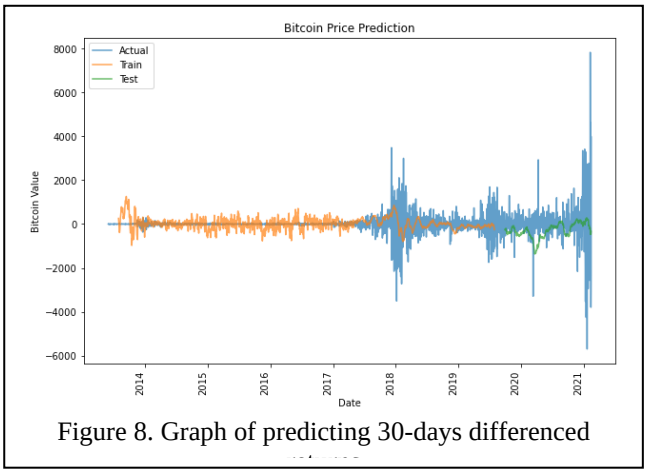

Figure 8. Graph of predicting 30-days differenced

Feats	Baseline	V0.05	VC0.05	V0.25	VC0.25
FD	x	x	x	x	x
NS		x	x	x	x
PS		x	x	x	x
Result					
RMSE Train	1187.23	1645.62	1308.42	1521.31	1346.67
RMSE Test	2828.29	3403.22	3078.89	2998.06	3354.80
MAE Train	560.33	880.22	714.92	750.89	795.99
MAE Test	1841.53	2817.38	1898.38	1953.35	2042.47

Figure 9. Results for 30-day differenced returns & sentiment

Feats	Baseline	V0.05	VC0.05	V0.25	VC0.25
FD	x	x	x	x	x
NS		x	x	x	x
PS		x	x	x	x
Result					
RMSE Train	4.53	4.73	4.56	4.70	4.66
RMSE Test	6.08	4.80	4.13	5.13	4.16
MAE Train	2.80	3.12	2.88	3.05	3.03
MAE Test	3.58	3.23	2.78	3.96	2.80

Figure 10. Results for Log 1-day returns & sentiment model

The methodology behind working out the 30-day returns is similar to 1-day returns. We take the price difference between price at n+30 and calculate the different between that price and at price at n. This roughly approximates the returns an investor can expect if they are to go long and buy 1 bitcoin at some time and sell it in 30 days' time. We difference these returns to make the time series stationary. The results can be seen in in figure 8.

We can see from the results that at none of the sentiment-based model perform better than the baseline for the 30-day returns. The RMSE and MAE are higher or the same as the baseline. This is likely due to the inherent difficulty in guessing the future returns on a longer timescale such as 30 days. At many points in the training and testing periods the model must guess a very unexpected return due to the volatility present in cryptocurrency market, especially towards the end of the test dataset. The model also must contend with 3 different supply and demand epochs due to bitcoin halving rewards. Therefore, one can conclude that there may be more successful methods in building our trading model.

Log 1-Day returns

The method for 1-day log returns was described in DE.5 in detail. Much of classical statistics assumes a normal distribution, therefore engineering a feature like log 1-day returns gives us the benefit of providing a target that when normalized approximates the normal distribution rather closely. Another important thing to note is that the input variables for all the financial data and the sentiment is for day n, while the target return the model must predict is the return for day n+1. In simple words the setup is such that the model attempts to predict the returns tomorrow if we decide to buy 1 bitcoin today. This differs from the previous models which attempt to either predict the closing price today using today's information. Or in the case of the monthly returns, they attempt to predict a return removed 30 days in time. Therefore, this may be the most interesting experiment as it act as a check to explore if the sentiment we compute today, holds some ground truth about the price tomorrow. This has a clear application in a short-medium term investment strategy.

From the test results the best performing model was the VADER sentiment model. It achieved the lowest RMSE in the testing data. One thing to take away is that every sentiment model had a lower RMSE in the test case when comparing with the baseline case. This suggests that at least in this particular setup sentiment helps to inform the model to make predictions closer to the ground truth than with just the price data alone. This means that there exists a possibility that sentiment from a large-scale forum could include information about the ground truth future price.

Further experimental results

One important factor which could be influencing the results is the scale of the dataset and the fact it covers 3 different supply-demand epochs. As briefly described before, bitcoin artificially constrains the mined block reward (supply) which heavily influences the price. As bitcoin was a niche asset only 10 years ago, there are still brand-new investors coming into the market. Therefore, we have a two-fold effect, where over the long term more money in flowing into the asset and the supply is being constrained, giving us an upward pressure on the price. We wish to explore if focusing on the period between July 2016 and May 2020 gives better results as the model would have to predict the same macroeconomic supply situation.

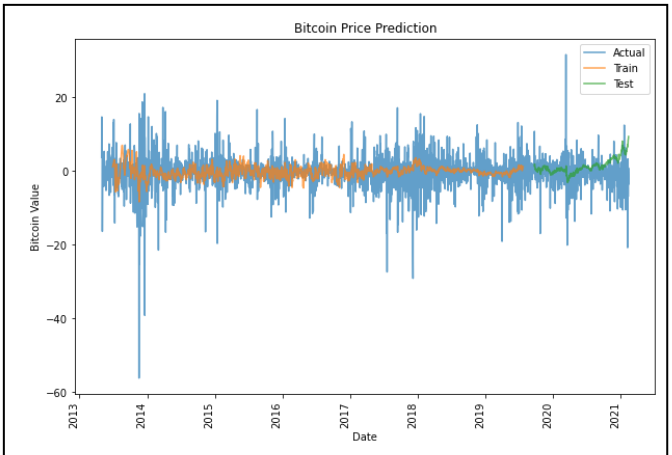

Figure 11. 1-day Log returns

Feats	VC0.05 original	VC0.05 3rd epoch	VC0.25 Original	VC0.25 3rd epoch
FD	x	x	x	x
NS	x	x	x	x
PS	x	x	x	x
Result				
RMSE Train	4.56	4.50	483.00	625.33
RMSE Test	4.13	4.61	2220.40	511.43
MAE Train	2.88	3.16	298.42	427.43
MAE Test	2.78	3.14	1410.97	373.88

Figure 12. Comparison of best models with 3rd epoch focus

To run this test we select the best results to see if they can be further improved. We select the close price VC0.25 model along with the VC0.05 1-day log return model.

At first glance the results may look promising, but one must keep in mind that the RSME metric relies on the absolute distance between prediction and actual value. As we constrain the time period we are interested in, we are constraining the possible past and future prices. For the entire dataset the price varies from around $100 to more than $60,000. When focusing on the period of the 3rd halving, we only consider prices in the range of around $500 to just under $20,000. Therefore, this range reduction partially acts to lower the RMSE metric. The 1-day log return model does not improve. It may be the case that constraining the dataset reduces the ability of the model to learn on historic data.

To summaries the results section, there certainly exists a possibility that a sentiment-based investment strategy could prove a worthwhile pursuit. From the results gathered, focusing on improving the 1-day log return would likely prove to be the endeavor that is most fruitful when investigating an investment strategy to implement in a real-world scenario.

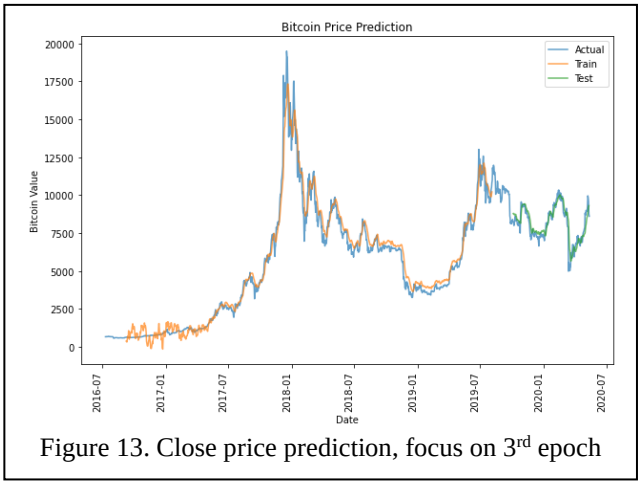

Figure 13. Close price prediction, focus on 3rd epoch

VI. Discussion

The aim of our thesis is to find the best possible candidate for an investment strategy based on sentiment. We explored 3 models as potential solution. We found that models such as Linear regression and ARIMA can achieve good results such as the stated literature, but to implement a real-world sentiment-based investment strategy, better results are necessary.

We focused on LSTM based models as the most promising solution due to the way they handle time series data. Their ability to update predictions from past data while learning what data improve the predictions the most means they managed to outperform the previously explored models. Further support was found in the literature available on the topic as other research found this to be the case on many different types of data such as blockchain statistic to telegram sentiment-based data. We found that an LSTM model can guess directionality and at times can extend to the magnitude of change of the financial metrics we tested.

There are some issues to raise in the spirit of scientific enquiry and honesty. We will discuss some of them in detail.

Reddit comment data

One of the pivotal assumption we make is that reddit comment data gives us an insight into the thoughts of cryptocurrency traders and investors. This is only partially true, and we do not know to what extent this is true. There are certainly members on this forum that hold and trade bitcoin among other cryptocurrencies, but there is certainly a portion of the commenters who add to the sentiment of this crowd while not having a position in the market. The sentiment we extract is representative or a very mixed cohort therefore we can claim that commenters have a certain sentiment, but not that investors share this sentiment.

Size of comment data per day

It would be important to note that the number of comments is not the same. The reason this is important to note is that in the early section of the data (pre-2015) the comments number at no more than the 100's with a few exceptions during the 2013 bull run. This means that on some days a comments' sentiment has far more weight than at other time periods. An individual comment during the 2017 bull run is one of around 10k-50k of comments per day, whereas in 2013 it could be one of only 50. This means that there is erratic sentiment behavior that occurs in the earlier section of the data that the LSTM model learns from due to each comment having a higher weight to the overall sentiment we extract.

Lack of true price estimation

Traditional financial instruments can have their intrinsic value estimated in various ways. One widely used metric for company stock valuation is the asset book-value. The book-

value, which works as a fair market price estimate of all the assets a company holds, gives an indication of the tangible of what can be recovered if a company fails. In the case of bitcoin and cryptocurrency in general, this is a lot harder to estimate. Some well known figures in the financial industry give the long-term price projection for all cryptocurrencies of $0. Other gives estimates that would put the bitcoin market cap at the same level or even above the market cap for gold, currently around $11 trillion. To this end, our model could certainly be improved by adding some measure of the true price of bitcoin. One such metric could be the electrical mining costs of bitcoin. There are some ranging estimates that are given by the CBECI which tracks the theoretical upper and lower bounds [24]. Incorporating such data could potentially give a a baseline price metric which could inform the model about the price of creating bitcoin, such a metric is discussed in [25]. To incorporate into our model we would have to collect data on bitcoin miners and the average electricity prices which vary worldwide. This could be a potential improvement.

Added VADER phrases

VADER follows a specific methodology described in section II, which is why we initially selected it as our method of sentiment extraction. One drawback with our added phrases is that they are not scored for sentiment by 10 independent raters. Some of them come form one of the updated GitHub from a user who states he followed the methodology [26], other crypto phrases are symmetrically added by us. An obvious improvement would be to follow the original methodology in future experiments.

Further improvements

New sentiment extraction methods have been developed in recent years such as the use of transformers. In [27] we see a state-of-the-art method that outpreforms any lexicon-based method for sentiment extraction. It makes the further improvement in entity extraction, where HuggingFace can link the sentiment and entity. This means that it is possible for it to understand what the sentiment is relating to and recognising the difference between objects such as different cryptocurrencies. In our results there is no distinction between sentiment that refers to bitcoin or sentiment that may refer to an irrelevant discussion that could have occurred due to an off-topic comment. This could be a potential way to improve the current results and increase the reliability fo sentiment extracted.

We could extent the social media sentiment data further to Twitter. As Twitter also provides an open mixed platform for opinion sharing, mining tweets with crypto-related hashtags present, we could run a comparison between reddit data and Twitter data based models. This could act as a benchmark for which sentiment informs the model most succesfully. An extention could use both data in tandem. A further benefit would be the presence of hashtags which could act as a proxy for entity extraction, as the HuggingFace model stated earlier is computationally expensive.

VII. Conclusion

We motivate our scientific inquiries in this area by aiming to provide improvements to sentiment-based investment strategies. Original reddit comment data was collected using the Pushshift API. We extracted the sentiment using VADER at different threshold and using an augmented crypto lexicon. Then engaged in some exploratory analysis of the emotional language structure using the NCR lexicon. We discuss the main aims and motivations of using VADER due to the wisdom of the crowds effect. A discussion of the current literature follows and describes other results that have been gathered, but due to the wide-ranging time period of the data we collected, no other paper presents the application of these methods across 8 years of data.

Different models were implemented to check their suitability for the task, choosing an LSTM due to the initial results. We checked the performance against the benchmark model RMSE of 6.08, and found the LSTM sentiment model with augmented crypto lexicon at the original VADER threshold of 0.05 outperforms with an RMSE of 4.13 when predicting the 1-day log returns percentage.

By focusing on an asset class which is highly susceptible to shifting investor sentiment, new methods and avenues of capital allocation can be found. Economic measurements by standard metrics such as GDP growth have stagnated in several places such as Europe and the United Stated. Due to this, institutions and retail investors will have to look to new markets to see their assets grow. Exploring the cryptocurrency space for this new area of growth can be beneficial for investors due to the high growth of the market. By informing their strategies using the methods we describe in this paper, we hope to better inform investors on how to maximize their capital allocation in the cryptocurrency market.

Cryptocurrencies are still in their relative infancy, as time goes on, they may become less susceptible to sentiment of retail traders when institutional investors increase their holding. Therefore, exploring these methods is very timely and an intellectually interesting pursuit. Humanity is fundamentally interesting. In these comments we collect, we see the percolations of what drove our ancestors to develop our civilization, the need for better, and the need for more.

VIII. Declaration

Declaration of Originality. I am aware of and understand the University of Exeter's policy on plagiarism and I certify that this assignment is my own work, except where indicated by referencing, and that I have followed the good academic practices

Declaration of Ethical Concerns. This work does not raise any ethical issues. No human or animal subjects are involved neither has personal data of human subjects been processed. Also no security or safety critical activities have been carried out

IX. References

[1] S. Nakamoro, "Bitcoin: A Peer-to-Peer Electronic," *bitcoin.org,* 2008.

[2] C. &. G. E. Hutto, "VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text," *Eighth International Conference on Weblogs and Social Media (ICWSM-14),* June 2014.

[3] N. Ferguson, The ascent of money: A financial, Penguin, 2008.

[4] M. Felix, Money: The Unauthorized Biography, From Coinage to Cryptocurrencies, Vintage, 2015.

[5] J. W. Malcolm Baker, "Investor Sentiment in the Stock Market," *Journal of Economic Perspectives,* vol. Volume 21, no. 2, pp. 129-151, 2007.

[6] K. S. J. V. Thien Hai Nguyena, "Sentiment analysis on social media for stock movement prediction," *Elsevier,* vol. 42, no. 24, pp. 9603-9611, 2015.

[7] "CoinMarketCap," CoinMarketCap, 19 08 2021. [Online]. Available: https://coinmarketcap.com.. [Accessed 19 08 2021].

[8] A. E. A. a. S. K. S. Wooley, "Extracting Cryptocurrency Price Movements from the," *IEEEE,* vol. Vol 18, no. 1, pp. 500-505, 2019.

[9] I. Welch, "THE WISDOM OF THE ROBINHOOD CROWD," *NATIONAL BUREAU OF ECONOMIC RESEARCH,* 2020.

[10] J. Surowiecki, The Wisdom of Crowds, Anchor, 2005.

[11] J. Baumgartner, "pushshift," 19 08 2021. [Online]. Available: https://pushshift.io/.

[12] N. N. Taleb, Skin in the Game: Hidden Asymmetries in Daily Life, Random House, 2018.

[13] L. A. A. K. R. P.-S. a. A. B. Abeer ElBahrawy, "Evolutionary dynamics of the cryptocurrency market.," *Royal Society,* 2017.

[14] L. A. B. L. A. G. A. B. Lorenzo Lucchini, "From code to market: Network of developers and correlated returns of cryptocurrencies.," *SCIENCE ADVANCES,* vol. vol 6, 2020.

[15] J. R. S. C. Sean McNally, "Predicting the Price of Bitcoin Using Machine Learning," *IEEEE,* vol. 26, 2018.

[16] L. l. e. al, "Bitcoin Options Pricing Using LSTM-based," *IEEE,* 2019.

[17] Y. Hua, "Bitcoin price prediction using ARIMA and LSTM," *Proquest,* vol. 218, 2020.

[18] N. Smuts, "What Drives Cryptocurrency Prices? An Investigation of Google Trends and Telegram Sentiment," *Performance Evaluation Review,* vol. 46, no. 3, 2018.

[19] I. G. N. G. A. G. Apoorva Aggarwal, "Deep Learning Approach to Determine the Impact of Socio Economic Factors on Bitcoin Price Prediction," *IEEE,* 2019.

[20] S. &. S. J. Hochreiter, "Long short-term memory," *Neural Computation,* vol. 9, no. 8, p. 1735–80, 1997.

[21] J. Korstanje, Advanced Forecasting with Python, Apress, 2021.

[22] "Reddit," Github, 19 08 2021. [Online]. Available: https://github.com/reddit-archive/reddit/wiki/API.

[23] G. Zuckerman, The Man Who Solved the Market: How Jim Simons Launched the Quant Revolution, New York: Portfolio, 2019.

[24] "CBECI," 19 08 2021. [Online]. Available: https://cbeci.org/.

[25] A. Hayes, "A Cost of Production Model for Bitcoin," *The New School for Social Research,* 2015.

[26] J. Badiola, "GitHUb," 15 6 2019. [Online]. Available: https://github.com/cjhutto/vaderSentiment/pull/81. [Accessed 19 08 2021].

[27] L. D. V. S. J. C. C. D. A. M. P. C. T. R. R. L. M. F. J. D. S. S. P. v. P. C. M. Y. J. J. P. C. X. T. L. S. Thomas Wolf, "HuggingFace's Transformers: State-of-the-art Natural Language Processing," *arXiv,* 2020.

X. Appendix

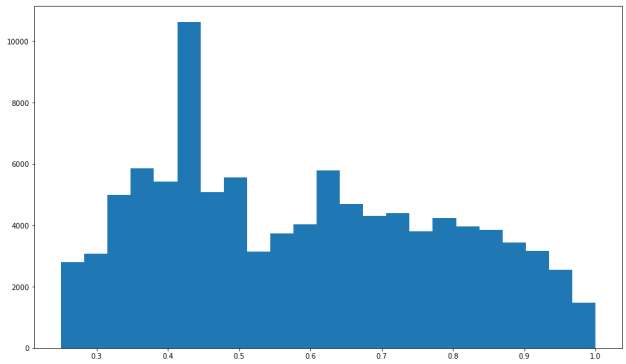

Figure A. Positive sentiment comment distribution

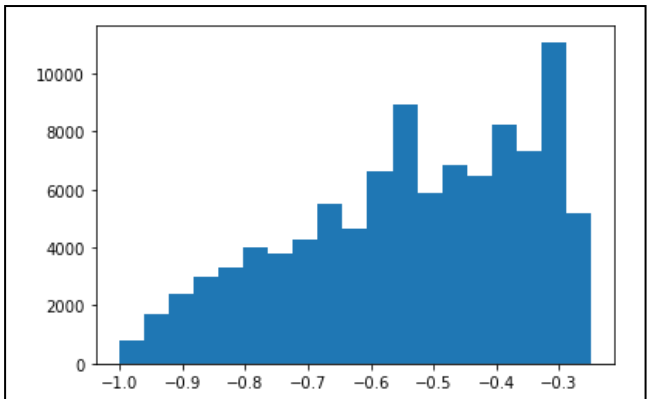

Figure B. Negative sentiment comment distribution

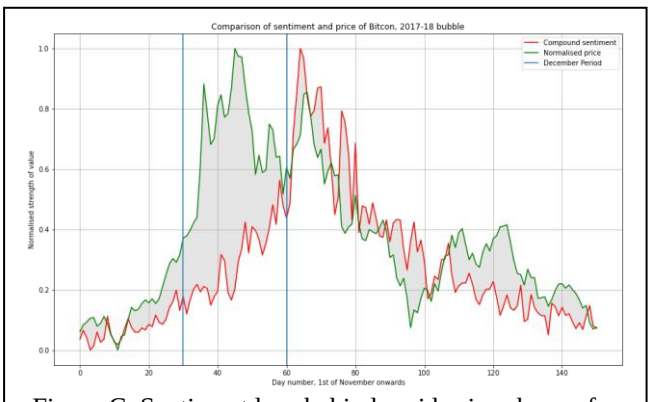

Figure C. Sentiment lags behind rapid price change for most comments

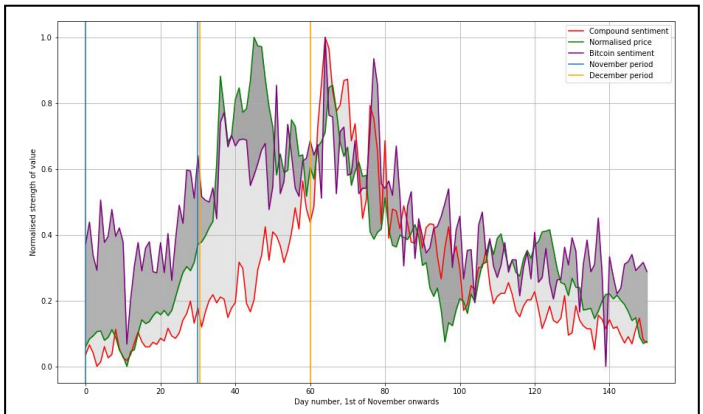

Figure D. Sentiment of comments containing ‘bitcoin’, non-bitcoin comments, price normalized, 2017-18 bullrun

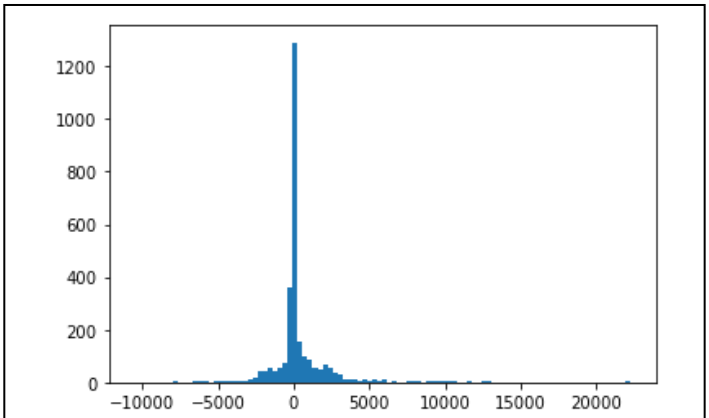

Figure E. Distribution of returns, note the extreme tails

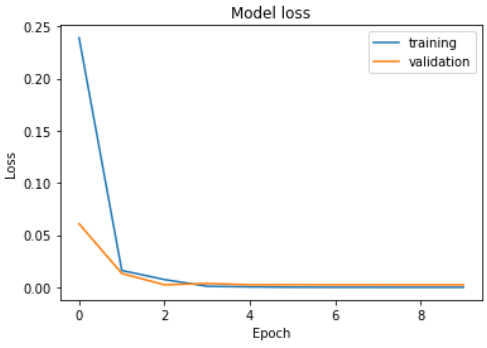

Figure F. Typical training and validation model loss, usually asymptote at 4-6 epochs.