Faculdade de Economia
Universidade do Porto

Social Media Sentiment in Cryptocurrencies Markets:
Application of granger causality and deep learning for price prediction

by

Luan Fermino Pires

Dissertation Master's in Finance

Supervised by:

Júlio Fernando Seara Sequeira da Mota Lobão

2021

Acknowledgements

I would like to give my gratitude to Professor Julio Lobão (PhD), who kindly and knowledgably guided me thorough this research, and introduced me to the mysteries of behavioural finance.

To my mother and sister and my family for their unconditional support both mentally and emotionally. Thank you from making me who I am today, and shall my success be always ours.

Finally, to my friends who kept me entertained through this challengingly journey with their contagious laughs.

Resumo

A literatura em finanças comportamentais sugere a existência de racionalidade limitada no estado mental dos investidores, maioritariamente derivado de enviesamentos emocionais incutidos na nossa natureza humana. Consequentemente, isto pode afetar o processo de decisão. A questão coloca-se em saber se estes enviesamentos emocionais afetam podem afetar a sociedade em geral, o seu processo de decisão coletivo e se é possível extrair e testar a previsibilidade que o sentimento formado tem nos mercados financeiros.

Mais recentemente, o crescimento de plataformas de redes sociais tem chamado a atenção como uma fonte valiosa de sentimento de investidor, mais especificamente entre investidores de retalho do qual estão mais sujeitos a enviesamentos emocionais, contrariamente a investidores institucionais, como sugerido em estudos anteriores.

O propósito desta tese é em primeiramente testar se o sentimento de investidor derivado do Twitter é significante para a previsão dos retornos nas "criptomoedas". O texto é analisado recorrendo ao "Valence Aware Dictionary for Sentiment Reasoning" (VADER) e os tweets são classificados em polaridades positivas, negativas e neutras. Uma análise de Causalidade de Granger é usada para testar se o sentimento formado pelo VADER prevê alterações nos retornos da Bitcoin (BTC). Os nossos resultados sugerem que o sentimento dos 3 dias anteriores é estatisticamente significante na previsão de retornos de BTC. Segundo, usando técnicas de aprendizagem profunda ("deep learning"), uma rede neural é criada para classificar texto derivado de tweets do Elon Musk, de acordo com o subsequente movimento dos preços da BTC e Dogecoin (DOGE) após o tweet ser postado. O modelo é capaz de prever movimentos de preço a curto-prazo de 1-minuto e 30-minutos, mas incapaz de para períodos mais longos de 1-dia. Concluímos que o sentimento derivado do Twitter é uma ferramenta importante na previsão nos movimentos de preço das criptomoedas, visto os investidores aparentarem ser mais influenciados por sentimento invés de fundamentais quando transacionam criptomoedas, questionando a Hipótese de Mercados Eficientes, para este segmento de mercado.

Abstract

Behavioural finance literature suggests the existence of bounded rationality within investors state of mind, largely derived from emotional biases embedded in our human nature. Consequently, this can affect the decision-making process. The question remains on whether these emotional biases can affect society in general and their collective decision making, and whether we are able to extract and test the predictivity that this formed sentiment can have in financial markets.

Most recently the rise of social media platforms has drawn attention as a valuable source of investor sentiment, more specifically along retail investors which are more prone to emotions biases, contrary to institutional investors, as suggested in previous studies.

The purpose for this thesis research is on firstly testing whether investor sentiment derived from Twitter is significant in predicting cryptocurrency returns. We analyse the text using the Valence Aware Dictionary for Sentiment Reasoning (VADER) and classify tweets into positive, negative, and neutral polarities. A Granger Causality analysis is used to test whether sentiment measured using VADER is predictive of changes in Bitcoin (BTC) returns. Our results suggest that sentiment from the previous 3-days is found to be statistically significant predictor of daily BTC returns. Secondly using deep learning techniques, we create a neural network to classify text from Elon Musk tweets accordingly to the subsequent price movement of BTC and Dogecoin (DOGE) after the tweets posting. We find that the model is able to predict very short -term price movements of 1-minute and 30-minute, but unable for longer time-periods of 1-day.

We conclude that sentiment from Twitter presents itself as a powerful tool for the prediction of cryptocurrencies price movement as investors seem to be more driven by sentiment rather than fundamentals when trading cryptocurrencies, questioning the Efficient Market Hypothesis (EMH) for this market segment.

Keywords: Investor sentiment; Neural Network; Cryptocurrencies, Bitcoin.

Contents

List of Tables

List of Figures

1. Introduction

An everlasting dilemma in financial literature is on whether asset prices are predictable. If indeed markets embed all available information, then any deviations from its equilibrium can only occur when new information arrives, behaving no different from a random walk (Fama et al., 1965). This is known as the Efficient Market Hypothesis (EMH), where markets are efficient and thus prices are not predictable. However, EMH is sustained on the assumption of human rationality, but advances in behavioural finance have questioned this rationality. Specifically, on the difficulty that economic agents have in making optimal decisions given their cognitive limitations, complexity of the problem and available time for the decision (Hutto et al., 2013).

Rationality is then "*bounded"* by these limitations, usually identified as biases. One interest experiment of such biases was conducted by Kahneman and Tversky (Tversky et al., 1981) which suggests that people tend to focus not only on the information that was presented to them, but also on how it was presented. In one of the experiments studied in the paper, participants were divided into two groups. Both groups received the same following problem with two sets of possible alternatives:

- "*Imagine that the U.S. is preparing for the outbreak of an unusual Asian disease, which is expected to kill 600 people. Two alternative programs to combat the disease have been proposed:*"

Solutions: 1st Group:

- "*If Program A is adopted, 200 people will be SAVED* "- 72 % of the participants chose A.
- "*If Program B is adopted, there is a 1/3 that 600 people will be SAVED and 2/3 probability that no people will be SAVED*"- 28 % of the participants chose B.

Solutions: 2nd Group

- "*If Program C is adopted, 400 people will DIE*"- 22 % of the participants chose C.
- "*If Program D is adopted, there is a one third probability that nobody will DIE, and 2/3 probability that 600 people will DIE*" - 72 % of the participants chose D.

In both scenarios the alternatives yield the same outcomes, however, the answers greatly differed amongst the two groups, simply because in the first, the alternatives were presented with a positive orientation, while in the second with a negative orientation. This phenomenon is known as the framing bias effect.

Interestingly, this bias suggests a link between cognitive limitations and emotional state of people. It suggests that people feel more comfortable and less guilty if in beliefs that they are *"saving lives"* and the opposite if believed that they are "*letting people die"*, suggesting that the context of information can impact the emotional state of individuals which in turn leads them to make biased decisions. If individuals act on manifestations of their emotional states, then it is reasonable to assume that they also act accordingly in both the economy and the financial markets. This general perception of public mood in financial markets is termed investor sentiment and can be broadly defined as the belief in future cash flows or investment risk not justified by facts at hand (Baker et al., 2007). Market bubbles are clear examples of such phenomenon, where prices usually rise at rates not justifiable by fundamental factors such as the formation and bursting of the Dot.com bubble categorized by a period of over optimism leading to a rise in price of speculative and difficult to value technology stocks in the late 1990s, which eventually crashed (Baker et al., 2007). Most recently the same phenomena occurred in the cryptocurrency markets with the unjustifiable rise and subsequent crash of Bitcoin prices late in 2017, largely attributed to over optimism in the cryptocurrencies (Chen et al., 2019).

Traditionally, aggregated investor sentiment or market sentiment is extracted through the use of surveys, such as the Purchasers Managers Index (PMI), the Economic Sentiment Indicator provided by Directorate General for Economic of the Financial Affairs, or the Consumer Confidence Index (CCI) and several studies analysed the link between these indicators and market movements such as in Xing (2018) and Lee (2019). However, gathering the data for these surveys can be both expensive and resource intensive.

This cost inefficiency, led to the creation of new methods to capture sentiment. With the rise of large-scale online data ("Big Data*"*) a new extensive amount of information regarding people's feelings and opinions is available for research. One common practice is on evaluating the semantic content on news media (e.g.: WSJ articles)

and capture its sentiment, such as Tetlock et al. (2008) which finds that negative wording from Wall Street Journal (WSJ) predicts negative information of firm-specific earnings, and that nearly 80 % of the information is immediately incorporated on prices. Alternatively, with the popularization of social media sharing, connecting opinions and ideas has never been easier, and for this reason investors have been paying close attention on the content of such information. A study conducted by Connel (2015) concluded that almost 80 % of institutional investors include social media information in their regular workflow and 30 % of these investors state that its content directly influenced their investment decision. Presumably if investors both rely and express their opinions on social media platforms, and such information has predictive value in the form of sentiment, then it is possible to test the link between sentiment formed in social media feeds and market returns. One plausible market segment to test such relationship is the cryptocurrency market, specifically Bitcoin.

Bitcoin was originally created has an electronic version of cash that would allow peer-to-peer payments without the need of financial intermediaries or the oversight of a central bank (Nakamoto, 2008). Recent developments now perceive Bitcoin has an investment opportunity quite comparable to gold (Hougan, 2018), especially among retail investors. According to Blockware Solutions data (2020) the volume traded on retail exchanges such Bitfinex and Coinbase was larger thorough 2017-2018 individually, and thorough 2019-2021 combined when compared with volume traded in the Chicago Mercantile Exchange (CME). Because social media are mostly used by retail investors (individuals) then it is reasonable to expect that these investors will discuss assets which they mostly trade on, such as Bitcoin. Thus, social media may play an important role, in understanding cryptocurrencies price movements.

This research studies this link, specifically, our goal is to test the predictive power that sentiment formed in tweets (feed messages) from Twitter - a platform with approximately 300 million registered users - has on cryptocurrency returns. We select cryptocurrencies not only because of their intense retail trading activity, but also because of their speculative nature.

This research follows De Long et al. (1990) behavioural financial theory which predicts that noise trading from investors affects financial markets if these types of

investors are plenty and limits to arbitrage are in place. Specifically in short-term temporary price deviations from the theoretical fundamental value derived from sentiment as suggested by Tetlock (2008).

Twitter is used given its wide acceptance in financial literature as a source of sentiment (Bollen et al., 2011; Mao et al., 2011; Sprenger et al., 2014). Sentiment is captured using two different Natural Language Processing (NLP)[1] techniques.

For the first, we use a pre-build lexicon classifier (VADER) that analyses tweet content for a given day and attributes a polarity score reflecting the sentiment orientation of the tweet into either positive, negative, or neutral and tests whether it can predict Bitcoin returns. We find statistical significance relationship between positive tweets and Bitcoin returns, however, note that the computed sentiment is not enough to explain cryptocurrencies price variation in the long run.

In the second methodology we investigate whether investors can predict cryptocurrencies price movement embedded in the textual information in tweets from Twitter, by exploring the use of deep learning[2] models for financial prediction for one particular user – Elon Musk. We narrow the scope of analysis because of the low explanatory power of sentiment on BTC returns found in the first methodology and focus on the significant relation found by Ante (2021) between Elon musk tweets and price variations of two cryptocurrencies - Bitcoin and Dogecoin. We contribute to the literature by presenting a model framework of a text classifier both adaptable to other market segments (e.g.: stocks) or sentiment sources (e.g.: Reddit or financial news), considering whether price movements are predictable by training a neural network. We find that the network is able to predict short time price movement at timeframes of 1-minute and 30-minute after the tweet is posted, but poorly performs for 1-day.

The rest of this research is organized as follow: In Chapter 2 we review the literature on investor sentiment analysis and its relevance in the financial markets.

[1] - Natural Language Processing is the process of computer interpretation and manipulation of natural language text.

[2] - Deep Learning is a subfield of artificial intelligence (AI) function used in processing data and creating patterns for use in decision making.

Chapter 3 describes the process of collection and cleansing of the data and the methodologies employed in capturing market sentiment and is subdivided into two parts. The first studies the broader relation between Twitter sentiment and Bitcoin returns through the use of a Granger Causality test. The second provides a narrower segment focusing on Elon musk tweets which recent research suggests, has a significant effect on price variations of Bitcoin and Dogecoin (Ante, 2021), through the creation of a Recurrent Neural Network (RNN) with Long Short-Term Memory (LSTM) units for text classification. Chapter 4 describes the results from the two methodologies and Chapter 5 summarizes the final conclusions and improvements for future research.

2. Literature Review

This chapter is divided in five sections. The first reviews the concept of investor sentiment in the context of behavioural financial literature advancing two proposed approaches to measure sentiment as suggested by Baker et al. (2007). The second presents the different measurement proxies for sentiment commonly used in studies of this field. The third explores the application of NLP techniques for sentiment classification of textual data, the methodology employed for this thesis research. The fourth reviews the literature on sentiment analysis from social media and the forms of testing its relationship with market returns. The fifth reviews the task of sentiment classification in the financial markets using deep learning techniques.

2.1. Investor sentiment analysis

Originally, investor sentiment was studied by observing the tendency of aggregate market returns to mean revert, or by testing the predictability of simple ratios based on fundamental factors (e.g.: dividends) to stock market value (Baker et al., 2007). The theoretical reasoning for this testing, is that if markets are truly efficient, then prices should reflect the true fundamental value of assets and any deviation from this value should be short lived. However, the persistence of several market bubbles suggest otherwise.

The reasoning for this mispricing can be largely attributed to two factors (Baker et al., 2007). Firstly, it is assumed that sentiment affects irrational investors which affect the prices and secondly, limits to arbitrage are in place not allowing rational investors to trade on the mispricing since betting against sentimental investors can be both costly and risky (Shleifer et al., 1997).

Baker et al. (2007) identify two ways of measuring sentiment: the “bottom up” approach and the “top-down approach”. The first relies on the use of individual investor psychology bias such as overconfidence, representativeness, and conservatism to explain overreaction and underreaction in financial markets, such as in models presented in Barberis et al. (1998) and Daniel et al. (1998). The second approach advanced by Baker

et al. (2007) “focuses on the measurement of reduced-form, aggregate sentiment and traces its effects to market returns and individual stocks”. That is, it focuses on explaining which assets are more dependent on sentiment and further advances that this dependency is greater the more difficult the asset is to be valued, typically associated with speculative assets - low capitalization, younger, unprofitable, high-volatility, non–dividend paying, growth companies or stocks of firms in financial distress. Furthermore, the authors also point out that these speculative assets are more costly to arbitrage, consistent with Shleifer et al. (1997) findings on limits to arbitrage and the perdurance of sentiment in markets.

This second approach suggests that sentiment is more pronounced on speculative issues. Most recently, studies suggest that cryptocurrencies fit this speculative class. Liu et al. (2021) shows that traditional asset pricing models and standard risk factors do not help explaining cryptocurrencies returns. Cheah et al. (2015), claim that the fundamental value of Bitcoin is zero meaning that financial models relying on fundamental factors for valuation cannot be applied for cryptocurrencies.

2.2. Measurements of investor sentiment.

Investor sentiment can be explicitly derived by looking at the chain of events after the occurrence of an exogeneous shock on sentiment which can be traced by changes in observable patterns of how securities trade. Some proxies for sentiment derived in such manner include surveys; mood proxies; retail investor trades; mutual fund flows; trading volume; premia on dividend-paying stocks; closed-end fund discounts; option implied volatility; first day returns on initial public offerings (IPOs); volume of initial public offerings; new equity issues; and insider trading. (Baker et al., 2007). However, all of these measures assert on some sort of fundamental benchmark to compare how pronounced is the mispricing from the proxied fundamental value.

Alternative, sentiment can be implicitly derived through opinion mining. This is usually done through the application of NLP techniques, computational linguistic and text analytic that identifies and extracts subjective information in source materials (Batrinca et al., 2015) separating emotions from textual data (Fang, 2015).

However, textual information can either express a fact or an opinion (Liu et al., 2012), meaning that information within text does not need to express an opinion. That is, it can be objective rather than subjective. "*Earnings per share are above expected"* is a fact, not an opinion since it is an actual observable occurrence. But we are not interested in the objectivity of the content in the text, but rather on the reaction of society towards the text. In other words, we are interested not only on the opinions expressed in text, but also on how investors react to the information available. In our example, it is expected a positive reaction, since the company appeared to be more profitable than expected, which is good, leading to a subsequent positive price movement.

Typically, sentiment classification problem for textual data can be formulated either as two separate classification problems, as a three-class or as a multi-class classification problem (Liu et al., 2012).

The first relates to both, the evaluation of the degree of subjectivity/objectivity of the textual information defined as subjectivity classification (Hatzivassiloglou et al., 2000) and to the classification of the subjective sentences has either positive or negative.

The second, extends the former by classifying subjectivity as either positive, negative, or neutral, where the latter reflects lack of opinion within text (Pang et al., 2009).

The third, also referred as ordinal classification is used when a three-class classification is not enough to capture the higher degree of classification desired by the researcher (Pang et al., 2009).

2.3. NLP Techniques for financial text

NLP techniques are the process in which text is converted into one of the chosen classifications above described, allowing researchers to identify the relationship between text sentiment and human behaviour (Wang et al.,2018). There are two general accepted methodologies for NLP's (Bukovina, 2016) – the lexical method and the machine learning method.

The first, also known as "*knowledge-based approach"* consists in comparing a pre-tagged list of words with the retrieved text and attributing a classification to it. The chosen list depends on the researcher intent of analysis and ranges into several domains. For example, Henry Word List (Henry, 2008) and the Loughran & McDonald Word List (Loughran et al., 2011), consist in hand tagged words specific for financial report.

Several studies use this sentiment measure to extract semantic information from financial markets. A widely applicable case use is on classifying text from financial news articles. Li et al. (2014) classifies words from financial news using Loughran & McDonald Word List and compare this sentiment with stock returns. Schumaker et al. (2012) conduct a similar research but use Arizona Financial Text (AZFinText) system on financial news as their sentiment classifier.

Most recently, the rise of social media platforms prompts several researchers to capture market sentiment derived from these platforms. Bollen et al. (2011) using OpinionFinder classify tweets from Twitter and test its predictability on Dow Jones Industrial Average (DJIA) returns. Kirlić et al. (2018) test similar predictability, but on Microsoft (MSFT) stock returns and related tweets, using VADER to classify sentiment from tweets.

It saves the researchers time by using pre-build lexicons models since no pre-tagging is necessary, however, it has the shortcoming if specific text from a different domain is to be analysed (Annett et al., 2008). In other words, the chosen model may not be appropriate for the sentiment task in analysis.

The machine learning approach overcomes this latter problem. By using intelligent modules which learn from historical data and contrarily to the rule-based approach, automatically induce rule from the training data (Khan et al., 2016). When applicable to NLP it allows the classification of specific text, suitable to create the intendent text classifier. One traditional classification algorithm based on this approach was proposed by Kalra et al. (2019) by using Naïve Bayes classifier to categorize financial news text as either positive and negative sentiment and use it to predict stock market daily movements. Most recently, recurrent neural networks – a deep learning technique - have been widely used given its excellent ability of extracting features and process variable

length text (Zhou et al., 2016). Further explanation of deep learning for text classification applicable for financial domain is presented in chapter 2.5.

2.4. Investor sentiment on social media

As observed, capturing sentiment can be done through NLP techniques and each technique is dependent on the domain of analysis. In line with the rise of large-scale online data - Big Data - a numerous amount of new research on sentiment analysis has grown in different fields such as in predicting elections (Jungherr et al., 2012), natural disasters populational activity (Wang et al., 2018) and pandemics developments (Raamkumar et al., 2020). As social media adherence rises both investors and companies must acknowledge its presence since it can significantly impact its reputation, sales, and in extreme cases, survival (Kietzmann et al., 2011).

Consequently, as people become more interconnected and the flow of information becomes more easily accessible, even a single user is capable of influencing an entire market sector. Most recently, this phenomena has manifested itself in the cryptocurrency sector with what can be called the "Musk Effect". This refers to the market price reaction followed by tweets from Elon Musk and was observed both in Tesla stock price (De Roo et al., 2020) and in cryptocurrencies in which the author tweets about, such as BTC and DOGE (Ante, 2021).

For these reasons, several new financial researchs study the link that sentiment retrieved from social media platforms has on financial markets. All of them agree on the existence of bounded rational investors who seek information through these platforms subsequently affecting their trading but are divided on whether investors are less sophisticated and trade on noise, or if sophisticated and contribute to market efficiency (Da et al., 2011). Sophistication is achieved through information demand where investors rely on a new information channel (social media) for their investment decisions (Bukovina, 2019), which can in turn permanently increase firm's valuation by reducing the frictions in information channels as suggested by Merton (1987). Noise trading is mostly associated with the over/under market reaction of investors to this new information, consistent with bounded rationality and is more in line with this research.

Karabulut (2013) using a vector autoregressive (VAR) framework finds that Facebook Gross National Happiness Index (FGNHI) predicts both daily returns and trading volume in the U.S. stocks, followed by a reversal in the following weeks, consistent with noise trading. The model statistical significancy is kept even after controlling for daily macroeconomic conditions.

Siganos et al. (2014) follows a similar approach using FGNHI within 20 international markets, explore the relation between daily sentiment stock returns, trading volume and price volatility and find a positive relationship in the first and a negative in the latter two. They also find reversal in the following weeks.

Mao et al. (2011) studies the different sentiment sources and their relationship with market returns. Using the Tweet Volumes of Financial Search (TV-FST) - total tweet volume of specific tickers - and the Twitter Investor Sentiment (TIS) - ratio of bullish and bearish tweet - they find that both indicators enhance daily return predictability. The relation is captured using a Granger Causality test using TIS as their sentiment explanatory variable. However, TIS semantics are too simplistic because each tweet is simply defined as positive(negative) if containing the terms "*bullish*" ("*bearish*"). Thus, a tweet such as "*I was bullish on AMZN, but now I'm bearish*" has a clearly negative semantic orientation, however, TIS would fail to identify it.

Nisar et al. (2018) following a similar approach of Karabulut (2013), compared daily changes in mood from Twitter with the FTSE 100 finding correlation among them, but no statistical significancy. Sentiment is captured using Umigon- a lexicon-based classifier specifically designed to detect sentiment in tweets (Levallois, 2013). It shares some classification characteristics of VADER – such as global heuristics, by considering the importance of emoticons and emojis have on sentiment derivation, and an n-gram decomposition, looping through each n-gram and checking its presence on several lexicon lists. However, as noted by the authors, when tested for performance, Umigon fails to identify negative sentiment in tweets, with a precision score below 50 %.

Sprenger et al. (2014) focus on companies quoted in the S&P 500 index and retrieve tweets regarding the tickers that compose this index. The text is classified using a Naïve Bayesian classification method. This methodology is a machine learning technique where the conditional probabilities of the messages belonging to a particular

class are estimated based on manually coded documents, used as training sets. As previously stated, one advantage of this methodology is that bullishness and(or) bearishness are manually tailored by the authors and not dependent on the positive and(or) negative classification of pre-built lexical lists. Interestingly, the authors fail to find a lagged relationship of bullishness with abnormal returns, however, they find the opposite to be true. That is, abnormal returns are followed by more optimism in social media feeds. Furthermore, they also find that Twitter community can distinguish users who provide high quality advice, from does who do not, but still unable to distinguish valuable piece of information, concluding that "*picking the right tweets remains just as difficult as making the right trades*" (Sprenger et al., 2014), suggesting that user influence (relevance) plays an important role in analysing market sentiment from social media.

Bollen et.al (2011) resorts to both lexicon-based and machine learning-based approaches. Firstly, through a Granger Causality test, the authors test the relationship between Dow Jones Industrial Average (DJIA) and the semantic orientation found in Twitter feeds. Sentiment is captured using different lexical based approaches, OpinionFinder - a binary classifier - and Google-Profile of Mood States (GPOMS), which classifies sentiment into six dimensions (Calm, Alert, Sure, Vital, Kind and Happy). They find that, only the Calm dimension from GPOMS over the past 3 days (optimal lags) had predictive power over DJIA price variations.

Regarding the effects of sentiment from social media for the cryptocurrency markets, Kim et al. (2016) capture sentiment with VADER pre-built lexicon and using a Granger-causality test, find that positive user comments significantly affect BTC price movements, and that negative comments and replies affect Ethereum and Ripple price movements. Kraaijeveld et al. (2020) using a similar approach find that average estimated sentiment can be used to predict the price returns of Bitcoin, Bitcoin Cash and Litecoin.

The first part of this research follows up in similar methodology of those employed in Bollen et al. (2011), Mao et al. (2011), Kim et al. (2016) and Kraaijeveld et al. (2020). We employ a Granger Causality analysis, to test the broader relationship between the lagged optimal sentiment variables computed from VADER pre-built lexicon list, like the one used by the two latter authors which attributes the polarity score for each

tweet and tests the null hypothesis: "Does *sentiment from Twitter granger causes Bitcoin Returns?".*

2.5. Deep learning in Finance

This chapter demonstrates how market sentiment can be captured through deep learning. Contrarily to linear regression models, such as Granger causality, deep learning can capture the non-linearity in timeseries such as the stock market (Lapedes et al., 1987). This is particularly important since the relation between public mood and the stock market is almost certainly non-linear (Bollen et al., 2011).

Furthermore, deep learning provides modelling at a high level of abstraction leading to a model that is flexible to input changes (Sohangir et al., 2018). That is, the network created within the learning process is invariant to changes on input data, since the optimal weights were already defined. When applicable for financial text mining, the input data can be the result from lexicon-based approaches and the output to be predicted the market price variation. For instance, Bollen et al. (2011) employs a Self-Fuzzy Neural Network between public mood and stock market values using the 3-day lagged period of public mood as input variables, found in the Granger Causality test, to predict changes in DJIA and successfully demonstrate that the network improves the accuracy in predicting DJIA market variations from changes in public mood.

Alternatively, words within text can be embedded into numerical vectors and through training, the network can learn to store the context of the text, in a low dimensional space (Salton et al., 1988).

Sohangir et al. (2018) for instance, explores the text classification of bullish and bearish investors formed from StockTwits – a social media platform, closer to Twitter, where each message can be labelled as "bullish" or "bearish" for a particular ticker – by training both a Convolutional Neural Network (CNN) and a RNN with LSTM units on the retrieved data. However, these models only account for general opinion on a particular ticker and are not necessarily connected to the price movement of that ticker.

One form to directly classify text according to market movement is by analysing the post market effect after the text source release. Kraus et al. (2017) using a RNN with LSTM units show that "deep learning can enhance financial decision support by explicitly incorporating word order, context-related information and semantics", by classifying financial disclosure accordingly to the subsequent price movement after the disclosure release. Souma et. al (2019), using the same neural network framework, classify sentiment as the 1-minute time window, before and after the release of articles from SeekingAlpha, where positive(negative) articles lexicon is expected to be followed by positive(negative) returns. Text it then classified according to the market orientation.

In the second part of this research, we create a text classifier using similar methodology approach as Kraus et al. (2017) and Souma et al. (2019) by developing an RNN with LSTM units for the classification of Elon Musk tweets accordingly to the market movement observed after the posting of the tweet, with the purpose of attributing sentiment to vocabulary directly related with price movement. In other words, we train the model to label the data and objectively define sentiment as the true price variation.

3. Data and Methodology

3.1. Data Collection

Twitter data is chosen given its acceptance as a sentiment tracker in the financial community (see chapter 2.4). Because each methodology on this research depends on different timeframes, two distinct datasets were collected and from now on are referred as DATA1 for the first methodology and DATA2 for the second. For the Granger Causality test we collected tweets containing either “BTC” or “Bitcoin” in the collected text from Kaggle, an open-source dataset, on a minute basis, from 01 Feb. 2017 to 29 April 2019 00:00 [GMT] for a total of 1 048 575 tweets. Retweets (re-messaging of the tweets) were removed since they provide the same semantic information. No personal data was used or revealed as part of the study. Pricing data was collected from another Kaggle open-source dataset on an hourly basis for Bitcoin OHLCV (Open, High, Low, Close) prices, and the daily volume traded as well with a total of 20 354 samples collected.

For the RNN with LSTM units, we extract all Elon Musk tweets containing the terminology related with Bitcoin and Dogecoin between 25 Apr. 2020 to 4 Jun. 2021 [GMT], for a total of 31 tweets, using Twitters API. Pricing data for BTC and DOGE was collected for the same timeframe using Binance API to a total of 578 375 samples for each cryptocurrency.

3.2. Data Cleansing

One problem with microblog text data, is the existence of irrelevant characters that provide no semantic information. These include user mentions (e.g.: @user), URL links (“http”) and unimportant special characters or punctuation. We follow a similar cleansing approach used in Bollen et al. (2011) and remove user mentions, URL links and special characters from tweets as illustrated in Table 1. Stop-words are also removed from

each tweet using the Natural Language Toolkit (NLTK*) stopwords.py*[3] (see Appendix 6.1).

However, contrary to the authors we kept specific punctuation such as question marks ("?") and exclamation points ("!") since our lexicon-based approach (VADER) can capture the magnitude of intensity derived from these characters without modifying the semantic orientation of the text (Hutto et al., 2013). For the same reason, ALLCAPS letters were also kept. Furthermore, we also preserved parenthesis ") (" and colons ":" since they are commonly used in typing of emoticons such as ":)", and emojis were also kept as recent research suggest its crucial importance in the automated sentiment classification of informal texts (Kralj et al., 2015).

Text	Filter	Filtered Result
"@[USER_NAME]: Bitcoin and crypto brace for a European Central Bank bombshell: https://t.co/e75Fr9WrjM by @[USER_NAME_N]"	@mention	"*Bitcoin and crypto brace for a European Central Bank bombshell: https://t.co/e75Fr9WrjM by"*
"*Bitcoin and crypto brace for a European Central Bank bombshell: https://t.co/e75Fr9WrjM by"*	URL Links	"*Bitcoin and crypto brace for a European Central Bank bombshell*:"
"*Bitcoin is now at 18K #BTC#ETH#XRP"*	Special Characters	*"Bitcoin is now at 18K BTC ETH XRP"*

Table 1: Text pre-processing filters.

Contractions were handled (e.g.: "isn't" = "is not") using a python module-*contractions.py* (see Appendix 6.2). To cope with the acceptable vocabulary of VADER,

[3] - Python module library corpus reader, which contains a pre-built list of stop words such as "is" , "at", "on" or "the", that are identified in the text and subsequently removed, at the exception of "*not*" and "*no"* which can change the semantic orientation of a sentence."

we identified the language in the text using FastText - *fasttext.py* [4] - and only kept English lexicon in the data (see Appendix 6.3).

Furthermore, Twitter has a limit of 140 characters per tweet but strangely, after the filtering, certain tweets were above this mark in DATA1. To cope with this issue, only tweets between [10-140] characters were maintained. A minimum of 10 characters filter was subjectively chosen, since text with few words usually does not have semantic meaning. This subjective character restriction is plotted in the Boxplots portrayed in Figure 1. After cleansing and filtering of the data a total of 529 670 tweets are kept in DATA1 and 31 tweets in DATA2.

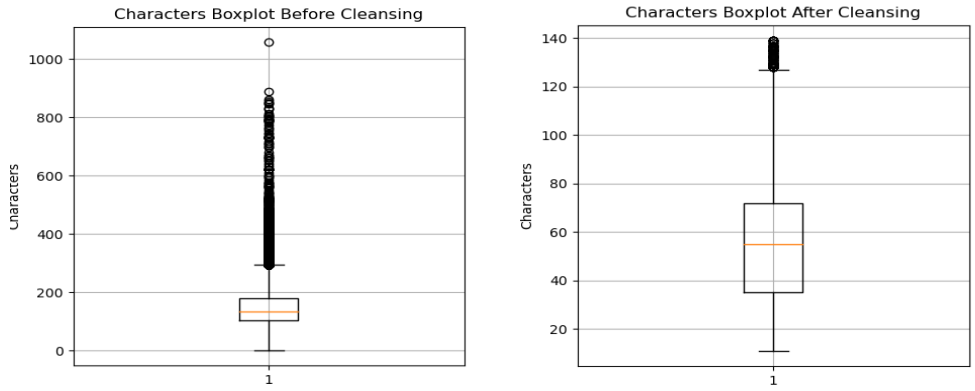

Figure 1: Characters distribution in Boxplot before and after data cleansing.

3.3. Spam detection

A problem regarding social media data is the existence of spam[5]. Spamming is the use of message system for sending unsolicited information. Evidence suggests a growing number of fake accounts and the use of bot activities (Ferrara et al., 2016)- software's designed for automated specific tasks, such as "*Twitter bots"* – which control fake accounts and automatically act by tweeting, re-tweeting, liking, (un)following and direct

[4] - Python module library which contains pre-trained models with vectorized English words retrieved from Facebook, and only kept English lexicon in the data.

[5] - SPAM is the use of message system for sending unsolicited information for other users.

messaging users. Spamming can maliciously affect the financial markets as suggested in Cresci et al. (2019) where the authors find coordinated groups of bot's spam social media feeds, promoting low value stocks by exploiting the popularity of high value one - which they coin as "*cashtag piggybacking"*.

However, the existence of spam is not necessarily bad since our purpose is to capture market sentiment. Indeed, as Cresci et al. (2019) suggest, spam is intended to affect public sentiment and even if malicious and uninformative, it can bear semantic orientation. As such, we will be splitting the DATA1 into two datasets, one containing spam and the other without spam. This filtering is only applicable to DATA1 since DATA2 is composed of Elon Musk tweets which do not contain spam.

Spam can be removed by training a model using a similar machine learning techniques previously mentioned such as deep learning (Wu et al., 2017), however, a machine learning spam classifier is beyond the scope of this research.

Alternatively, we resort to existing literature that identifies typical characters found in text spam, specifically for Twitter. Kwak et al. (2010) characterized spam tweets by those containing shortened URL, recommending their removal, while Cheong et al. (2010) focused on user spam accounts, identifying characteristics that are typical observed in those accounts such as the exclusion of certain biographic information. Following a similar approach, we filter spam by firstly removing tweets containing URL mentions such as "*https"* and "www." Secondly, we identify terminology that can be prone to *spam* activity in tweets specific for the dataset using a *"Word Cloud"* which presents the most frequent words used in the text data. This is done using an open Python library – *wordcloud.py*- which shows the most frequent words in the text in sizes. Larger (Smaller) sizes correspond to higher(lower) word frequency. The key metadata values are shown in Figure 2.

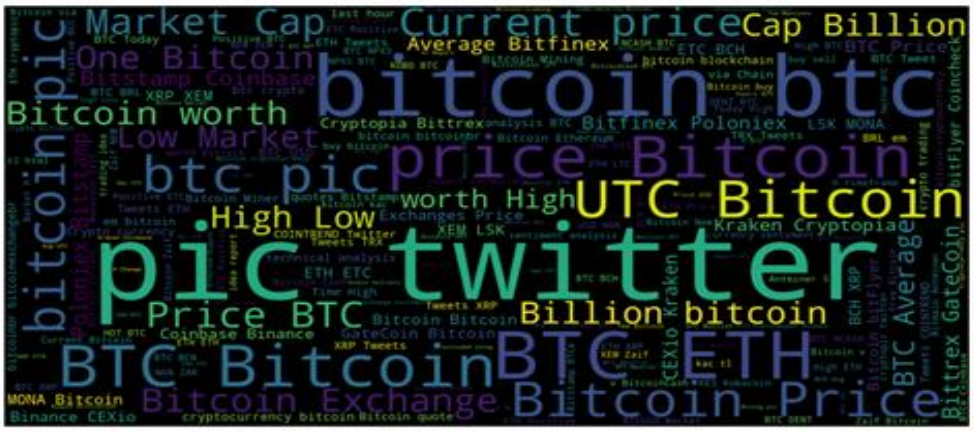

Figure 2: Word Cloud for most frequent words found in the dataset.

The numerous referrals of exchanges and trading platforms such as Bittrex, Kraken or Coinbase are suspicious and upon manually inspection we confirm the existence of spamming bots on the referral of exchanges and trading platforms and remove tweets containing this information. Thirdly, because of the high degree of Twitter bots, majority of the spam comes from repeated posting. Thus, users who are consistently posting content are assumed as spammers. This assumes that users are not consistently posting or discussing about Bitcoin. To confirm the hypothesis, we count the number of tweets each account has posted in the dataset and order users according to their count. We select the top 100 users and check both their username and text to see if the account derives *spam activity*. We confirm such activity and filter out these users from the dataset. Fourthly, we remove duplicates consistent with the spam detection approach employed in Kim et al. (2016). Finally, tweets not containing a semantic orientation, appear to be associated with spam. This is confirmed through visual inspection of the data for tweets with a neutral polarity score equal to 1 derived from VADER.

These five approaches allow us to partially filter out spam tweets from the dataset, but never fully. This is because of its subjectivity and on the arbitrary way in which we classify spam in tweets. Furthermore, this approach will also potentially remove non-spam tweets from the dataset. For instance, one URL mention is not necessarily spam. However, the number of expected spam removals outweighs the number of non-spam removals, since the identification is based on typical spam classification.

After the filter, we estimate that nearly 80 % of the dataset is comprised of spam, greatly differing from the usually 10-14 % observed and suggested in Kraaijeveld et al. (2020).

3.4. Sentiment Attribution (Lexicon-Based Approach)

To extract sentiment from each tweet and define our sentiment variables we use VADER, a rule-based model for sentiment analysis regarding social media text, inspired in sentiment word banks such as the Linguistic Inquiry and Word Count (LIWC) and the Affective Norms for English Words (ANEW), extending them by incorporating lexical features that are common in social media text. These include emoticons (e.g.: ":)"); emojis (e.g.: "😊"); acronyms and initialism (e.g.: "LOL") and common slang (e.g.: "nah" or "meh") summing up to a total of 9000 lexical features candidates. Once collected, each lexical candidate was rated by 10 independents human ratters, for a total of 90 000 reviews. Next, the model is passed through a deep qualitative texting verification that incorporate human heuristics in the evaluation of text such as punctuation, capitalization, degree modifiers (such as adverbs) and contrastive conjunctions (such as "but"). The model sets itself as a gold-standard classifier for microblogging context such as Twitter (Hutto et al., 2014).

The model is a trinary classification where each tweet produces a vector of sentiment scores divided into positive, negative, and neutral normalized between 0 and 1 and a compound score which aggregates all the other three, normalized between -1 and 1 – illustrated in Table 2.

Clean Tweet	Compound	Neg	Neu	Pos
"*Bitcoin breaks cryptocurrency value continues surge coo David Sapper talks bitcoin price adoption thanks great article*"	0.8555	0	0.58	0.42
"*Crypto bad case Monday's bitcoin dips bitcoin cryptocurrencies ripple Ethereum*"	-0.5423	0.28	0.72	0
"Markets update btc prices suffer loss since December"	-0.7003	0.492	0.508	0

Table 2: VADER Sentiment Attribution Examples for the three polarity scores
Neg stands for negative tweets, *Pos* for positive and *Neu* for neutral tweets. *Compound* is the normalized results of the three polarity scores.

3.5. Descriptive Statistics

3.5.1. DATA1 (Lexicon Based Approach)

The descriptive statistics for the lexicon-based approach methodology are summarized in Table 3 of the pricing data for BTC minute prices and BTC daily returns and sentiment data derived from VADER. Compound*;* Neg (Negativity); Pos (Positivity); Neu (Neutral).

	Compound	*Neg*	*Neu*	*Pos*	*Price*	*Returns*
Count	211 235	90 428	529 375	169 430	20 353	848
Mean	0.08	0.03	0.89	0.07	5512.41	0.3%
St. Deviation	0.30	0.09	0.16	0.13	3523.19	4.5%
Min	-0.99	0.00	0.00	0.00	760.38	-16.1%
Max	0.98	0.90	1.00	1.00	19869.86	27.6%
Skewness	-1.32	3.01	-1.32	1.85	1.04	0.36
Excess Kurtosis	0.94	10.49	0.94	3.22	1.35	0.94

Table 3: Descriptive statistic of DATA1 for sentiment and price data.
Pos represents tweets with a positive polarity. *Neg* represents tweets with a negative polarity. *Neu* represents tweets with a neutral polarity. *Compound* represents the combined polarity from *Pos*, *Neg* and *Neu*. *Prices* are the Bitcoin minute prices. *Returns* are daily Bitcoin returns of the open prices.

The count values differ because certain tweets do not have polarity scores (equal to zero), thus only *Neu* variable presents the count for the full dataset. Positive tweets outweight negative ones almost twice consistent with Kennedy et al. (2006) results which find that lexicon-based approaches generally have a positive bias. Both positive and negative sentiment variables present a high skewness of 1.85 and 3.01 and excess kurtosis 3.22 and 10.49, respectively, showing a peak distribution skewed to the left. These values range between [0-1] which indicates that majority of text in tweets does not have defined semantic orientation. This is further supported by the negative skewness in neutral sentiment variable which indicates that these values are concentrated around 1. These results indicate a small degree of semantic orientation within the data and a bias favouring positive tweets.

3.5.2. DATA2 (Machine Learning Approach)

The relevant descriptive statistics for DATA2 are summarized in Table 4 for all 31 collected tweets. Each tweet is labelled into a binary classification accordingly with the subsequent price movement of 1 for positive returns and 0 for negative returns for 1-minute, 30-minute and 1-day after its post.

	Price Movement Binary Class.		
	1-Minute	**30-Minute**	**1-Day**
Count	31	31	31
Positive Count	25	23	13
Negative Count	6	8	18
Mean	0.81	0.74	0.42
St. Deviation	0.40	0.44	0.50
Min	0	0	0
Max	1	1	1

Table 4: Descriptive statistic of DATA2.
The statistics refers to the binary classification of price movements. Positive (Negative) counts represent the number of times the price of either BTC or DOGE went up(down) after 1-minute, 30-minute or 1-day the time of the tweet posting.

Labels are either 0 or 1 for each tweet with a total of 25, 23 and 13 positive labels and 6, 8 and 13 negative labels for the 1-Minute, 30-Minute, and 1-Day resolution, respectively, indicating positive sentiment in shorter time windows. This implies that the

classification task will be positively biased for the shorter time windows when compared with 1-day window and the methodology results affected by it.

3.6. Granger Causality Test

Grangers Causality is a statistical hypothesis used to test if one time series improves the predictability of another time series. It is a mathematical formulation based on linear regression modelling of stochastic processes (Granger, 1969). If variable *X* "*granger causes"* variable *Y*, then, *X* past values should help predicting *Y* values. In other words, *X* changes systematically occur before *Y* changes and the model will exhibit the statistical significancy correlation that *X* has with *Y*. However, correlation does not mean causation even if suggested by the name of the method.

Firstly, the BTC time series, denoted as R_t is defined, reflecting the daily returns and is given by $R_t = \frac{BTC_t}{BTC_{t-1}} - 1$. The restricted autoregressive model (AR) is computed with the lagged values of R_{t-i}, given by equation 1:

$$(1) \quad R_t = \alpha + \sum_{i=1}^{N} \beta_i R_{t-i} + \varepsilon_t$$

Secondly, the sentiment variable time series, denoted as X_t are defined reflecting the relative changes in the sentiment variables (*Pos*, *Neg*, *Compound*, *Pos(SPAM)*, *Neg(SPAM)* and *Compound(SPAM))* and corresponding lagged values are added to form the unrestricted regression, given in equation 2.

$$(2) \quad R_t = \alpha + \sum_{i=1}^{N} \beta_i R_{t-i} + \sum_{j=1}^{N} \theta_i X_{t-j} + \varepsilon_t$$

To test whether each sentiment variable (X_t) independently predicts changes in Bitcoin prices, the variance of the two models is compared through an F-test for all values of J being jointly equal to zero. Priorly to test the hypothesis we first need to test whether

the data is stationary or not – if the joint probability distribution of a stochastic process does not change when shifted in time. Using relative variables should make the time series stationary, however, since this is an iterative process the series may not be stationary.

To test the if the time series are stationary, we employ an Augmented Dicker-Fuller (ADF) test for Bitcoin price differences and sentiment variables in analysis given by equation 3.

$$(3)\ \Delta R_t = \alpha + \beta_t + \lambda R_{t-1} + \sum_{i=1}^{N} \theta_i \Delta R_{t-i} + \varepsilon_t$$

We test the null hypothesis that $\lambda = 0$ (unit root) against the alternative $\lambda < 0$. Rejecting the null hypothesis suggests stationarity in the time series. The Δy_t corresponds to the first difference of the variable and the corresponding *t* lag. The terms α and β correspond to the constant and trend factor and can be either equal or unequal to zero, depending on which specification of the model pursued.

3.7. Recurrent Neural Networks (RNNs)

3.7.1. Deep Learning

In chapter 3.6 we tested the link between sentiment and BTC returns. The process depends on a pre-built lexicon-based classifier - VADER which depends on pre-classified vocabulary. However, this classification may ignore important text features that better reflect investor sentiment towards BTC. A positive(negative) tweet derived from VADER may not reflect a "*bullish" ("bearish")* view on BTC. Take the following example:

Clean Tweet	*Compound*	*Neg*	*Neu*	*Pos*
"I used to have over 300.00 in bitcoin in March of 2014. :(:(:("	-0.8271	0.42	0.58	0

Table 5: Tweet polarity attribution example.

This tweet was written in 06 of January 2018 at 18:23, when BTC price was around $ 17 000 per BTC, while in 2014, Bitcoin prices ranged around $ 300- $ 800 per BTC. Clearly, the user is regretful of disposing its BTC, representing a negative mood. However, given the context, the sentence should be regarded as bullish, since it shows a missed investment opportunity in BTC, but the polarity scores indicate otherwise.

Thus, it may be preferable to evaluate the text itself rather than the polarity scores and evaluate its predictive power over cryptocurrency price movements. The classification of text can be done using deep learning techniques, attributing value to words accordingly to a context of bullishness or bearishness in prices and independently from pre-classified dictionaries. This reasoning is illustrated in Figure 3 retrieved from Kraus et al. (2017) and adapted for our case.

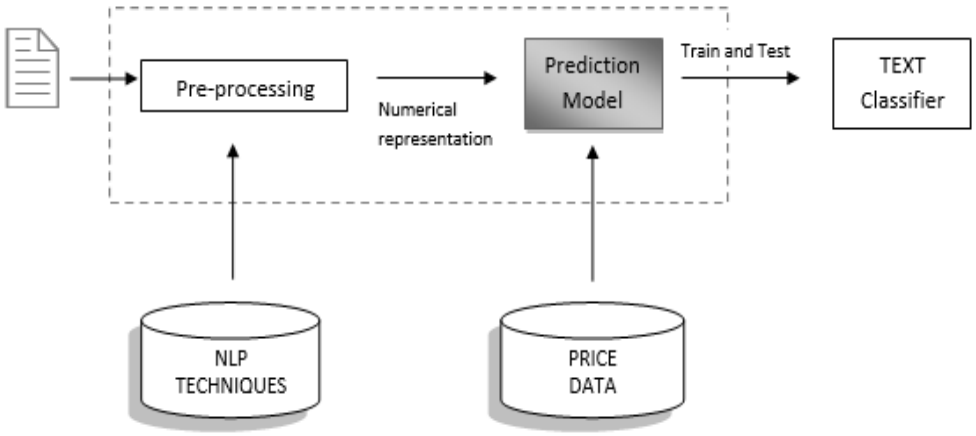

Figure 3: Unwrapped text classifier derived from prediction model.
(Figure adapted from Kraus et al., 2017)

The process starts by selecting the relevant text data – the tweets. We pre-process the text using NLP techniques using similar techniques defined in chapter 2.3. We then embed each tweet into numerical vectors using Python library - *tokenizer.py* [6]. This process is necessary since neural networks learn from numbers, not directly from text itself. Because tweets differ in size, we "*pad*" [7] the vectors for equal length (see Appendix

[6] Python module library used to convert input text into a stream of tokens, where each token can be a word, number, special character, punctuation, etc.

[7] Padding refers to the process of transforming the vectors into equal length, by adding zeros at the beginning or the end of the retrieved vector.

6.4). The vectors are then fed into a prediction model as input and the corresponding price variation as our output vectors. The model is trained and tested by splitting the vectors and price movement into training set and validation set. The former is used to train the optimal weights within the model. After trained, the model accuracy is evaluated over the validation set.

In this chapter, we present the use of deep learning to define our prediction model, by summarizing the framework of the proposed neural network. We take a similar approach to Kraus et al. (2017) and Souma et al. (2019) and train an RNN with LSTM units capable of reading Elon Musk tweets, learning how to classify them as either positive or negative accordingly to the subsequent price movement after each tweet post and testing its accuracy on the validation set.

3.7.2. Artificial Neural Network (ANN) architecture

ANNs are computer systems that learn to perform tasks from the inputted data, without defining any specific task. Deep neural networks are simply ANN with higher degrees of complexity - multiple layers - hence the term *"deep"*. They are inspired in biological neural structure that constitutes the brain. A collection of interconnected nodes (neurons) transmits information from one to another, similar to "*brain synapses"*. The strength of each signal depends on the weight that the previous node has on the transmission for the subsequent node and so forth, where each signal is either passed on or staled. This strength is captured, using an activation function such as *Sigmoid, Tanh or ReLu* functions. Figure 4 schematizes the structure of a simple ANN.

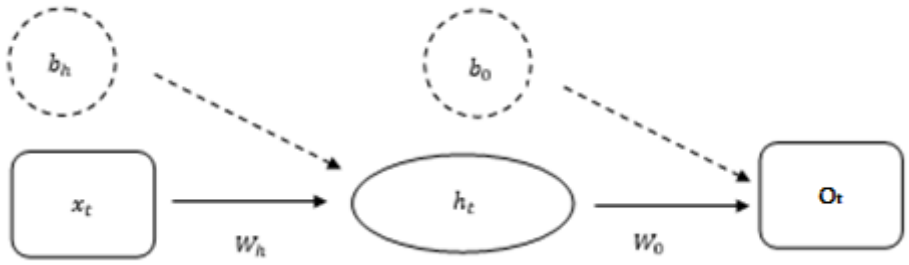

Figure 4: Simple ANN structure with one input, hidden and output layer and a bias term at each node. x_t stands for the input layer; h_t stands for the hidden layer; O_t stands for the output layer. b_h and b_0 stand for the bias term introduced at each layer.

The ANN is divided into three distinguishable layers. The input layer given by x_t represents independent variables used as input for the prediction - in our case, tweets - and can be presented as a matrix of size NxD where D is the number of features per sample and N the total number of samples. The hidden layer given by h_t of size NxM, where M is the number of nodes of the subsequent layer. W_h represents the intermediary weights between the input and hidden layer of size DxM and are the responsible for the updating of the network upon training. h_t is given by equation 4 and is the product between the transpose of W_h and x_t passed through an activation function which captures the non-linear transformation of the input allowing it to learn complex data. A bias term can be added given by b_h to shift the activation function for better fitting.

$$(4)\ h_t = \sigma\left(W_h{}^T x_t + b_h\right)$$

The output layer is given by O_t of size NxK, where K is defined as the number of outputs classification and set to 1 for binary classifications. O_t is given in equation 5.

$$(5)\ O_t = \sigma\left(W_0{}^T x_t + b_0\right)$$

3.7.3. RNN architecture

RNNs are transformed versions of traditional neural networks for dealing with sequential data such as text data (Goodfellow et al., 2016). Contrarily to the classical ANNs, each node is now dependent of both a new input and the previous node state information, referred as the hidden state as shown in Figure 5.

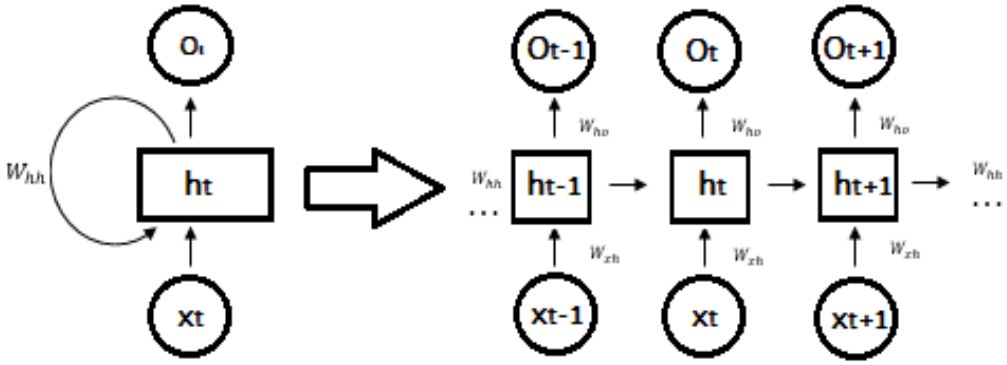

Figure 5: Folded and Unfolded RNN structure.
x_t stands for the input layer; h_t stands for the hidden layer; O_t stands for the output layer. W_{xh} , W_{xh} ,W_{h0} correspond to the intermediary weights between input-to-hidden, hidden-to-hidden and hidden-to-output layers, respectively.

Now the hidden layer depends not only on the input, but also on the inputs used in the previous hidden state (h_t) given by x_{t-1} of size NxT where T substitutes D and stands for the length of the padded sequence (fixed length per tweet). W_{xh} (DxK*)* are the input-to-hidden weights and $W_{hh}$the hidden-to-hidden (TxT*)* weights. h_t is given by equation 6, while the output layer is similar to equation 5.

$$(6)\ h_t = \sigma\left({W_{xh}}^T x_t + {W_{hh}}^T x_{t-1} + b_h\right)$$

The connections in the RNN form a direct cycle, which allows for the passing of information from one word to the next permitting the RNN to implicitly learn context-sensitive features (Kraus et al., 2017).

3.7.4. Modern RNN Units

One problem with RNN's is that they cannot memorize long term dependencies of the previous states and end up "*forgetting"* the information embedded in previous output layers. This is a consequence of the vanishing gradient descent problem which often limits its application to real-world problems (Bengio et al., 1994). During backpropagation (explained in chapter 3.7.6), the weights update is dependent of the

partial derivative of the error function at each iteration. As layers and nodes (complexity) are added to the network, the gradient can become vanishingly smaller making the network unable to learn. The farther back an input x_t is, the more its gradient vanishes, and the network forgets the early y_t values, which means that RNNs have problems in learning long term dependencies. In our case this implies that the RNN can forget the initial words from longer tweets, reducing its reliability.

To overcome this problem, we employ a modern RNN Units, the LSTM firstly proposed by Hochreiter et.al. (1997) capable of learning long term dependencies and more appropriate than other modern units such as Gated Recurrent Units (Britz et al., 2017 and Weiss et al., 2018).

3.7.5. LSTM Units

LSTM Units are modified versions of RNN's capable of storing long sequences in its weights by enforcing constant error flow within the unit, solving both the vanishing, and exploding gradient problems (Hochreiter et al., 1997).

RNN with LSTM units employs hierarchical structures including large number of hidden layers, to automatically extract features from ordered sequences of words and capture non-linear relationships or context-dependent meanings of words (Souma et al., 2019). Consequently, LSTM's have become widely used in many fields of research (Goodfellow et al., 2016), including the application in the financial literature and has been proven useful in improving decision support based on financial news as suggested by Kraus et al. (2017) which report higher accuracy when compared with traditional machine learning techniques, and Souma et al. (2019) showing that the network on average is able to predict both positive and negative sentiment from news with an accuracy of 76 %. The LSTM unit structure is illustrated in Figure 6, retrieved from Yuan et al. (2019).

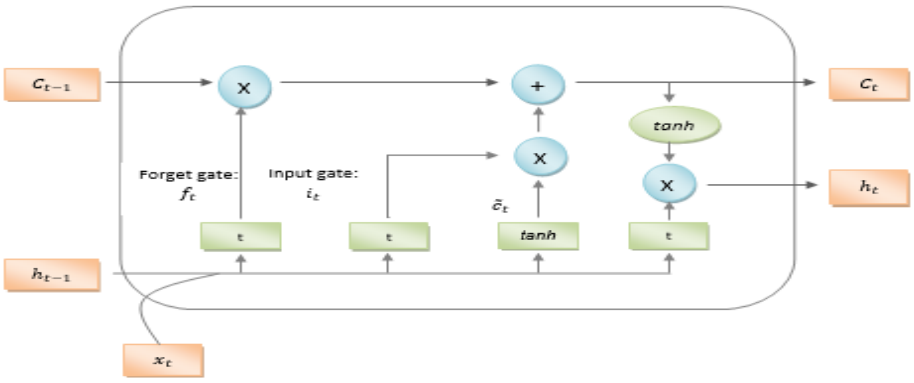

Figure 6: Structure of an LSTM Unit.
f_t is the forget gate. c_t is the cell state. i_t is the input gate. h_t is the hidden layer. *Tanh* is an activation function. x_t is the input layer.
(Source: Yuan et al., 2019)

A cell state given by c_t runs through the entire network. The LSTM Unit can remove or add information to the cell state through the use of gates. The forget gate given by f_t which learn to reset units and thus filtering which information to be discarded from the unit and takes $h_{t-1}$and x_t as inputs, given by equation 7.

$$(7)\ f_t = \sigma\left({W_{fx}}^T x_t + {W_{fh}}^T h_{t-1} + b_f\right)$$

Within the input gate we have a sigmoid layer deciding which values will be updated and is given by i_t in equation 8, while the tahn layer given by $\tilde{c}_t$ which is used to update the cell state presented in equation 9.

$$(8)\ i_t = \sigma\left({W_{ix}}^T x_t + {W_{ih}}^T h_{t-1} + b_i\right)$$

$$(9)\ \tilde{c}_t = tahn\left({W_{cx}}^T x_t + {W_{ch}}^T h_{t-1} + b_c\right)$$

The cell state is updated by the pointwise multiplication of the previous cell state C_{t-1} by f_t , forgetting irrelevant information and by adding the pointwise multiplication of i_t with $\tilde{c}_t$, which selects relevant information, and it is given by equation 10.

$$(10)\ c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t$$

Finally, the output gate given by O_t decides which parts of the cell state are relevant, given by equation 11. The former is then multiplied by the new modified cell state (C_t) passed through a tahn function to determine what information the new hidden state should carry, given by h_t in equation 12.

$$(11)\ O_t = \sigma\left({W_{Ox}}^T x_t + {W_{Oh}}^T h_{t-1} + b_O\right)$$

$$(12)\ h_t = O_t \odot tahn(c_t)$$

3.7.6. Training of the network

Constructing the network starts by randomly initializing the weights (W) at each layer leading to a prediction that differs from the target values (real values). The difference between the targets and the predicted values is refereed as the loss and can be defined by a loss function. Mean-squared error is used when dealing with regression, but since the output of our model is a binary classifier, a binary cross entropy loss function is used - given by equation 13 - where y_i represents the class and p_i represents the probability of a given class (0,1).

$$(13)\ J(W) = Binary\ Cross\ Entropy = -\frac{1}{N}\sum_{i=1}^{N} -\left[(y_i * \log(p_i) + (1-y_i) * \log(1-p_i)\right]$$

Backpropagation refers to the training of the network by minimizing the loss function. The process starts by taking the partial derivative and computing the gradient (the slope of the loss function for W_h, b_h, W_0, b_0 inputs shown in Figure 6) with respect to the weights at each layer, starting from the last layer and backpropagating to the first (Cilimkovic, 2015).

Gradient descent minimizes the loss function, by updating the parameters in the opposite direction of the loss function. It is an iterative process where each step can be defined by ∇W. A learning rate (η) determines the size for each step taken (Ruder, 2016). Weights are updated at each iteration and the learning rates defines the size of the update.

$$(14)\ W = W - \eta * \nabla W * J(W)$$

Summarizing, the network iteratively updates the weights until approaching the local minimum. The learning rate will define the smoothness of the process by determining how fast the network reaches the minimum. If set too small, the convergence can be painfully slow, while if set too large convergence can be hindered making the training oscillate and even deviate from the local minimum (Ruder, 2016).

3.7.7. The Model

In the following chapter we present the construction of our text classifier using RNN with LSTM units. Ideally we would create a text classifier for all available tweets regarding BTC for a more generalized model, however, it is plausible to assume that since majority of the Twitter users are retail investors, opinion about prices may be either misinformed or irrelevant (such as spam tweets), even if bearing sentiment. It is more plausible to assume that the opinion of influent users has a more significant impact on prices than average users as suggested in Sprenger et al. (2014).

Elon Musk has most recently drawn attention among cryptocurrency investors and as suggested in recent research, Musk tweets appear to have a significant effect on cryptocurrency markets (Ante, 2021).

The RNN with LSTM used for the model creation is based on the methodological approach used in Kraus et al. (2017) and Souma et al. (2019). Both authors objectively classify text from financial news accordingly to the subsequent price movement after the news release. However, they differ on the classification task, where the former uses a binary classification to predict price direction after the release of the news and thus using an activation function in the last layer of the model, while the latter use a regression in the last layer to predict the price return. Nevertheless, both authors implicitly assume a causal relation between the news and the price movements.

Our model in turn, implicitly assumes an existent influence that Elon Musk tweets exert on the cryptocurrency market and classify tweets using a binary classification for bullish tweets (Y = 1) and bearish tweets (Y = 0), accordingly to the subsequent price movement for 1-minute, 30-minute and 1-day event window after their release.

Firstly, the data is prepared by splitting the data into training and validation set. Tweets are converted into vectors of size NxT, where N represents the number of samples while T the vector size. T is padded into equal length of size 202 defined by the maximum number of words the network is allowed to learn. Increasing this amount would be necessary if the network were to train on a larger length text data.

Secondly, we initialize the model by passing the tweets into an embedding layer. Word embedding allows the network to extract the semantic and syntactic representation of words as low dimensional continuous vectors (Zhang et al., 2019). This is done using Keras[8] "Tokenizer" from Tensorflow[9] function which converts tweets into sequence of integers, where each integer is key index of a token in a dictionary. The dictionary values are the latent factors that capture the semantic relationship of words. The process is illustrated in Table 6.

Word	Integer	Latent factors (size 6)		
I	1	0.37	0.00	0.25
Bitcoin	2	0.22	0.32	0.30
💔	3	0.25	0.22	0.20

Table 6: Tokenizer function example.

The selected embedding layer consists of a maximum vocabulary is also of size 202, consistent with the size chosen in the vectorization of words, and a latent factor of size 8. These embedded vectors are then passed into the first and unique LSTM layer. Each unit is responsible for either storing or forgetting the relevant information from previous word(s) (integer(s)). The output from each unit is passed through a sigmoid layer

[8] Keras is a deep learning API created in Python running on top of Tensorflow
[9] Tensorflow is an end-to-end open-source platform used for machine learning

classifying the text as either bullish or bearish. The architecture of the network is shown in Figure 7.

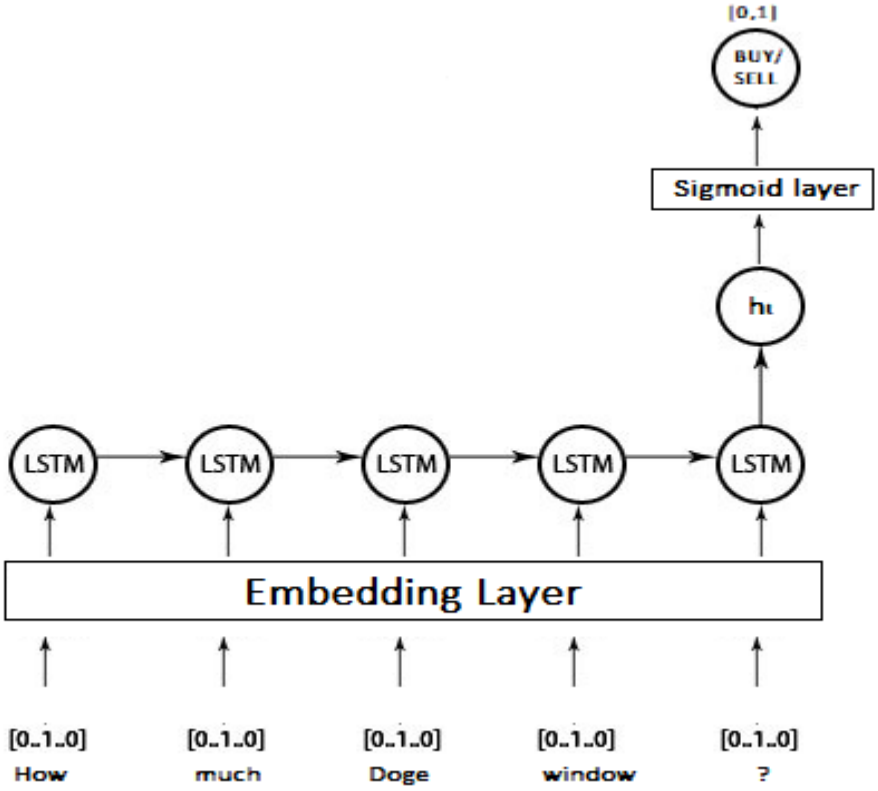

Figure 7: Elon Musk text classifier using RNN with LSTM units

Finally, the model is compiled using binary cross entropy loss function. At each iteration, the weights are updated dependent of the chosen hyperparameters. These include the number of LSTM units, the learning rate, the batch size[10] and epoch number[11] There are procedures to optimize these hyperparameters for better tunning (Nakisa et al., 2018), however, they are beyond the scope of this research and thus are arbitrarily selected by trial and error, keeping those who minimize the loss and maximize the accuracy of the model.

The model is created using Python and its implementation is illustrated in Appendix 6.5.

[10] Number of samples the model runs through at each iteration update
[11] Number of times each batch is trained.

4. Results

4.1. Granger Causality Test

In this chapter we present the results for testing the hypothesis on whether sentiment derived from Twitter correlates with changes in Bitcoin prices. We use Granger Causality Test as described in chapter 3.6 and test whether changes in sentiment systematically occur before returns. We firstly run an ADF test over all the independent variables in analysis, including the daily returns to test the stationarity of the time series in order to proceed to the model creation by using equation 4. Table 7 summarizes the testing results.

Dickey-Fuller test for unit root							
Variable	***Returns***	***Comp***	***Comp(SPAM)***	***Pos***	***Pos(SPAM)***	***Neg***	***Neg(SPAM)***
Test Statistic - Z(t)	-27.8	-28.7	-42.0	-22.2	-42.6	-25.9	-38.4
1 % Critical Value	-3.4						
5 % Critical Value	-2.9						
10 % Critical Value	-2.6						

Table 7: Augmented Dickey Fuller Tests for all 7 time series variables.
Pos represents tweets with a daily average positive polarity. *Neg* represents tweets with a daily average negative polarity. *Compound* represents the daily average combined polarity from *Pos*, *Neg* and neutral tweets. The SPAM label is to distinct tweets containing spam from those not containing it.

All test statistics are below the critical the 1 % critical value, thus we reject the null hypothesis of a unit root equal to zero, suggesting that the data is stationary for all 7 variables in analysis. We perform the Granger Causality test by testing the statistical significance of the added explanatory variables (sentiment), defined in equation 2 and equation 3, for a total of five lags, suggested as an efficient choice by Kraaijeveld et al. (2020). The p-value results of all six sentiment variables for each lag are presented in Table 8 while Table 9 presents the estimated parameters from the VAR model.

Lag	*Compound*	*Compound(SPAM)*	*Pos*	*Pos(SPAM)*	*Neg*	*Neg(SPAM)*
1-Day	0.319	0.05	0.356	0.179	0.527	0.192
2-Day	0.149	0.171	0.175	0.528	0.074	0.185
3-Day	**0.004****	0.836	**0.002****	0.678	0.216	0.475
4-Day	0.939	0.379	0.291	0.46	0.77	0.846
5-Day	0.999	0.619	0.869	0.629	0.313	0.647

* $p<0.05$; ** $p<0.01$

Table 8: Statistical significance (p-values) of bivariate Granger-causality correlation between all 6 sentiment variables and BTC returns for the period of January 2017, to April 2019.
Pos represents tweets with a daily average positive polarity. *Neg* represents tweets with a daily average negative polarity. *Compound* represents the daily average combined polarity from *Pos*, *Neg* and neutral tweets. The SPAM label is to distinct tweets containing spam from those not containing it.

Lags	*Compound*	*Pos*	*Neg*	*Compound (SPAM)*	*Pos (SPAM)*	*Neg (SPAM)*
1-Day (Rt)	-0.000	0.001	-0.003	-0.015	-0.020	-0.019
	(0.00)	(0.03)	(0.09)	(0.44)	(0.59)	(0.56)
2-Day (Rt)	0.024	0.025	0.029	0.035	0.030	0.028
	(0.70)	(0.72)	(0.85)	(1.03)	(0.87)	(0.82)
3-Day (Rt)	0.055	0.052	0.053	0.033	0.027	0.030
	(1.59)	(1.50)	(1.54)	(0.95)	(0.78)	(0.88)
4-Day (Rt)	-0.067	-0.064	-0.064	-0.059	-0.065	-0.061
	(1.95)*	(1.86)*	(1.87)*	(1.72)*	(1.90)*	(1.80)*
5-Day (Rt)	0.047	0.046	0.049	0.050	0.043	0.046
	(1.36)	(1.34)	(1.42)	(1.45)	(1.27)	(1.34)
1-Day (St)	0.009	0.026	-0.021	0.114	0.164	-0.237
	(1.00)	(0.92)	(0.63)	(1.97)*	(1.34)	(1.31)
2-Day (St)	-0.013	-0.039	0.060	-0.087	-0.083	0.265
	(1.44)	(1.36)	(1.79)	(1.37)	(0.63)	(1.33)
3-Day (St)	0.027	0.091	-0.042	-0.013	0.055	0.142
	(2.85)**	(3.13)**	(1.24)	(0.21)	(0.41)	(0.71)
4-Day (St)	0.001	-0.031	-0.010	0.056	0.098	0.039
	(0.08)	(1.06)	(0.29)	(0.88)	(0.74)	(0.19)
5-Day (St)	-0.000	0.005	0.034	-0.029	-0.059	0.083
	(0.00)	(0.17)	(1.01)	(0.50)	(0.48)	(0.46)
Constant	-0.003	-0.005	0.001	0.000	-0.006	-0.003
	(0.55)	(0.77)	(0.38)	(0.04)	(1.51)	(1.09)
N	821	821	821	842	842	842

* $p<0.05$; ** $p<0.01$

Table 9: Coefficient estimators for the VAR model used in Granger Causality Test.
Rt stands for n-day return for every n^{th} lag. St stands for n-day sentiment for every n^{th} lag. The sample period differs between spam and non-spam because of the removal neutral polarity scores equal to 1 used in the spam filtering.

Based on the results, only *Compound* and *Pos* variables have a high statistical significance with *p-values* < 0.01, both for 3-day lag similar to the significant lag found in Kraaijeveld et al. (2020) for the *Compound* variable. We find a positive effect for both variables of 9.1 % for *Pos* and 2.7 % for *Compound*, suggesting that positive sentiment and normalized sentiment of the last 3 days prior to the price movement positively affects BTC returns. However, contrarily to the authors we do not find further predictive power in any of the other lags. Furthermore, the results suggest that the 4th lagged returns is significant in explaining BTC returns as indicated by the negative coefficient at 4-Day lag with *p-values* < 0.05.

Finally, the global statistical significance tests of the models, used in the Granger Causality test are presented in Table 10.

Dependent variable	Independent variable	Chi-Square	df	p-value
Returns	*Compound*	11.375	5	**0.044***
Returns	*Compound(SPAM)*	5.55	5	0.352
Returns	*Pos*	12.23	5	**0.032***
Returns	*Pos(SPAM)*	8.05	5	0.154
Returns	*Neg*	5.6159	5	0.345
Returns	*Neg(SPAM)*	9.93	5	0.077

* p<0.05; ** p<0.01

Table 10: Granger Causality Wald Test - Global statistical significance test for each of the 6 sentiment variables over daily returns.

Accordingly with the results we reject the null hypothesis that sentiment does not predict BTC returns at a *p-value* < 0.05 for both *Compound* and *Pos* variables. The results are consistent with Kraaijeveld et al. (2020) which also found statistically significant predictive power of Twitter sentiment on bitcoin returns for the *Compound* variable and Kim et al. (2016) for the positive tweets, however, contrarily to the authors, we fail to identify a significant relation between negative tweets and BTC returns. Furthermore, spam does not seem to have predictive power over BTC returns as suggested by the lack of statistical significance for all three variables containing tweets with spam.

4.2. RNN with LSTM units

The following chapter presents the results from the model defined in sub-chapter 3.7.7. Following Souma et al. (2019) methodology, the model trains on an event window by classifying text accordingly to the subsequent price movement after the post of Elon Musk tweets for 1-minute, 30-minute and 1-day time windows. We construct a RNN with 50 LSTM units and split the validation and training set into 33% and 67%, respectively. The network is trained using 50 epochs over 20 batches to a total of 1000 iterations and the weights are updated at a 0.01% learning rate. Given the small data size the small learning rate allows the model to learn the features progressively rather than instantly at early epochs. For similar reason, the small batch size of 1 is selected to reduce the risk of underfitting the data and compromise its ability to generalize the results (Keskar et al., 2019). After trained, the performance of the neural network is measured by analysing the loss and accuracy evolution per each epoch trained, for both the training and validation sets. Figure 8 plots the loss per iteration while Figure 9 plots the accuracy per iteration for the two sets and Table 11 summarizes the metric values for the last trained epoch (50).

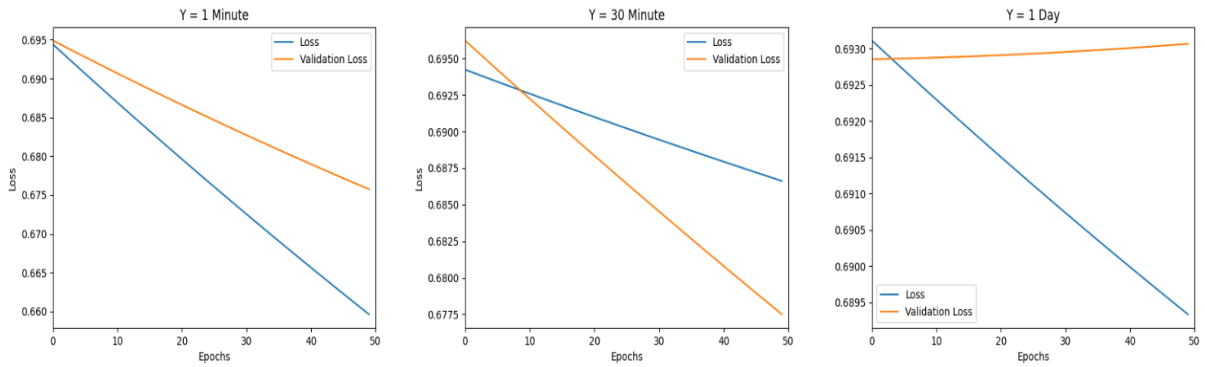

Figure 8: Training Loss and Validation Loss evolution for three event window classification limited at 50 epochs.

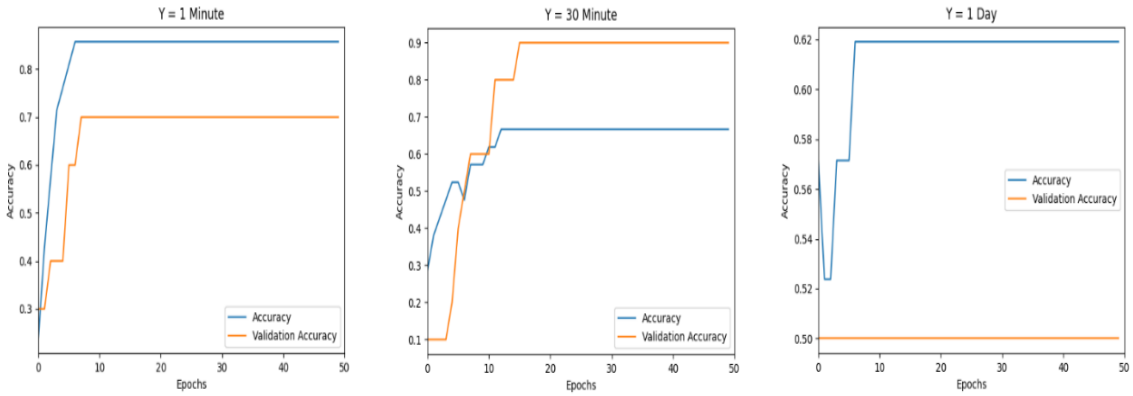

Figure 9: Training Accuracy and Validation Accuracy evolution for three event window classification.

Metrics at 50 epochs	Model		
	1-Minute	30-Minute	1-Day
Training Loss	0.66	0.69	0.69
Validation Loss	0.67	0.68	0.69
Training Accuracy	0.85	0.69	0.69
Validation Accuracy	0.70	0.90	0.50

Table 11: Performance results summarized for a total of 50 epochs trained.

4.2.1. Model Improvements

There are several factors that constrain the model's practicability for real case scenario (e.g.: Live Data), mostly related with the small data size and the lack of normalization of the labels.

Firstly, as show in sub-chapter 3.5.2, the data is mainly composed of positive price movements for all timeframes, especially for the short horizon. This means that the model tends to classify words more positively (bullish) and neglect negative words. This positive biasness is not an error, but a feature, since it captures the average positive sentiment that Elon Musk has on cryptocurrencies. However, this makes the model unfit to predict bearish sentiment. One possible solution would be to normalize the data into 50:50 positive-negative price movements labels, but that would greatly reduce the trained vocabulary. Another would be a conversion of the model to a multiclass classification

capturing sensitivity within tweets and putting less weight on individual vocabulary that has a positive association to it such as the word “Doge”.

Secondly, the tweets size and the total number of vocabularies trained are very small (202). If the model is to be put in practice it cannot be constrained to such small training set otherwise new vocabulary written by Elon Musk is not yet featured in the model and thus has no predictive power.

One solution to both problems is to simply increase the dataset, without necessarily label the data according to the price movement, but by manual tagging. The model at its current state, objectively classifies text according to a subsequent price movement, but this price movement itself is made by people perception on the tweets semantic orientation – if it is bullish or bearish. Thus, the model can be further trained by manual tagging vocabulary according to its semantic orientation. Tweets can be arbitrarily classified as positive or negative and then passed through the network because certain vocabulary is widely known to be associated with the classification intent of the network such as “buy” or “sell”. Alternatively, instead of manual tagging, the network can implement features from existent sentiment classifiers such as those defined in sub-chapter 2.3 for further enhance training.

5. Conclusions

Market efficiency asserts on the assumption that human rationality is solely able to explain asset prices, however, behaviour finance challenges this view by emphasizing the importance of human emotion in the investment process. Sentiment analysis can be useful in explaining the behaviour of financial markets. Traditionally, the capturing of sentiment was done through surveys, however, with the recent rise of large-scale web data, new sources of sentiment emerged for further research.

In this thesis we study the impact that sentiment from Twitter has on cryptocurrency markets. We select this market segment, because of its wide acceptance among retail investors, which previous literature suggests, are more prone to emotional biases and bounded rationality and thus more likely to act on market sentiment.

In the first part we analyse the broader impact that sentiment derived from Twitter can have on BTC returns, between 1 January 2017 until 29 April 2019.We test the hypothesis on whether sentiment is able to predict changes in BTC returns. Sentiment from Twitter is captured using VADER lexicon classifier, classifying them into positive, negative, neutral, and compound classes for a total of 529 375 tweets. We filter out for spam derived from Twitter bots, accordingly to suggestions in NLP literature, compromising to nearly 80 % of the collected sample greatly differing from the expected 10-14 % range as suggested, and test its significance by splitting into spam and non-spam tweets finding that spam plays no significant effect in predicting BTC returns. We perform a Granger Causality to test the hypothesis, a common approach used in research of this field such as in Bollen et al. (2011); Mao et al. (2011); Kim et al. (2016) and Kraaijeveld et al. (2020). We reject the null hypothesis, suggesting that sentiment has predictive power over BTC returns. More precisely, we find that negative tweets do not granger cause returns while positive tweets do, consistent with Kim et al. (2016) findings on the greater significant effect of positive tweets in comparison with negative ones on cryptocurrency returns. Furthermore, we find a stronger correlation between sentiment three days before the actual BTC price movement with positive coefficient values of 0.09 and 0.03 for positive tweets and the compound, respectively, at p-values < 0.01. The

results suggest the refute of EMH hypothesis. As a suggestion for future research, a larger set of cryptocurrencies and different periods of observation could be used for testing.

However, this approach raises two concerns. Firstly, because it depends on a pre-build lexicon classifier, some vocabulary may not be included in the classifier, ignoring important text features that define market mood. Secondly, it assumes that average market sentiment is a reliable predictor for price movements ignoring factors such as user influence in the analysis.

In the second part of this research, we propose the creation and training of an RNN with LSTM units to account for the two raised concerns above mentioned. This network works as a text classifier that classifies vocabulary accordingly to the subsequent price movement for 1-minute, 30-minute and 1-day timeframes of Bitcoin and Dogecoin after a tweet is posted. We include user influence by only selecting Elon Musk tweets and test the classifier accuracy on a validation set of tweets, achieving an 81 % accuracy rate for smaller time frames (1-minute and 30-minute) and 40 % for larger time frames (1-day). This is consistent with the noise trading behaviour hypothesis advanced by De Long et al. (1990) and further suggested in related studies such as Karabulut (2013) and Siganos et al. (2014), since the network is only able to accurately predict price movement on very short timeframes and not for longer timeframes implying price reversal between the periods.

However, the scarcity of vocabulary in the data makes the network an unreliable text classifier for price movement prediction, by putting more weight on isolated vocabulary when in reality has no semantic meaning (e.g.: such as the repetition of the word Doge in the tweets) and by failing to account the vast existent text features available in the English lexicon. Since the training vocabulary is scarce, the network ability to generalize classifications is limited. As suggestion for future work, it would be interesting to overcome this limitation by introducing a higher variety of vocabulary as explained in chapter 4.2. Indeed, the main feature of the proposed neural network is that more lexical features can be trained to improve the model performance accordingly to the researcher's intent. Furthermore, textual information may not be enough in predicting sentiment and other variables may play an important role for such task. For instance, relating financial sentiment derived from social media, specifically Twitter, such as tweet volume as

suggested by Mao et al. (2011); user influence as suggested by Sprenger et al. (2014) and even market data through the use of technical indicators as suggested by Vargas et al. (2017) can be proven useful in predicting price movements. It is possible to add these non-lexical features into the network, however, such applications are beyond the scope of this thesis. Nevertheless, such inclusion may prove useful in further improving predictability of BTC price movements performance, a potential topic for future research.

Finally, it would be interesting to test the profitability of the trained network in short term trading strategies applicable to 1-minute and 30-minute timeframes, however, as stated above, the model would most likely be inadequate for such task since the trained vocabulary is scarce and positively biased.

6. Appendix – Code Implementation

6.1. Filtering of Characters and stop words removal

```
import re
import nltk
from bs4 import BeautifulSoup
from nltk.tokenize import word_tokenize
from nltk.tokenize import WordPunctTokenizer ## This is used to eliminate the double space created from the cleaning of the data
from nltk.corpus import stopwords

pat1 = r'@[A-Za-z0-9./]+'
pat2 = r'https?://[A-Za-z0-9./]+'
combined_pat = r'|'.join((pat1, pat2))

def tweet_cleaner(text):
        soup = BeautifulSoup(text, 'lxml')
        souped = soup.get_text()
        stripped = re.sub(combined_pat, '', souped) ## HTML and @ cleaning


        try:
                clean = stripped.decode("utf-8-sig").replace(u"\ufffd", "?")
        except:
                clean = stripped
        letters_only = re.sub("[^a-zA-Z_, ?!:()]"," ", clean) # Character filtering

        stop_words = set(stopwords.words('english')) - set(["not","no"])
        words = nltk.tokenize.word_tokenize(letters_only)
        words = [t for t in words if not t in stop_words] # Removal of stopwords
        return (" ".join(words).strip())
```

6.2. Contraction's handling using Contractions

```
import contractions

df["clean_text"] =df["clean_text"].astype(str)
df["no_contract"] = df["clean_text"].apply(lambda x: [contractions.fix(word) for word in x.split()])
df['No_contract_STR'] = [' '.join(map(str, l)) for l in df['no_contract']]
```

6.3. Removal of Non-English tweets using Fasttext

```
import fasttext

pretrained_model = "lid.176.bin"  ## Pretrained model from Facebook
model = fasttext.load_model(pretrained_model)
langs = []
for sent in df['clean_text']:
  lang = model.predict(sent)[0]
  langs.append(str(lang)[11:13])
df['language'] = langs

english = df["language"] == "en"
df = df[english]
```

6.4. Tokenization of words into sequence of vectors using Keras

```
import tensorflow as tf
from tensorflow.keras.preprocessing.text import Tokenizer
from tensorflow.keras.preprocessing.sequence import pad_sequences


max_voc =len(word_list) # Total number of words in the data
tokenizer = Tokenizer(num_words = max_voc)
tokenizer.fit_on_texts(text_train)
sequences_train = tokenizer.texts_to_sequences(text_train)
sequences_test = tokenizer.texts_to_sequences(text_test)
data_train = sequence.pad_sequences(sequences_train, maxlen=max_voc)
data_test = sequence.pad_sequences(sequences_test, maxlen=max_voc)
```

6.5. RNN with LSTM model implementation using Keras.

```
from tensorflow.keras.layers import Input, Dense,LSTM, SimpleRNN,
Flatten,GRU,GlobalMaxPool1D, Embedding
from tensorflow.keras.models import Model, Sequential
from tensorflow.keras.optimizers import SGD, Adam
from keras.layers import Dense
from keras.optimizers import SGD

embedding_vector_lenght =6
```

```
model = Sequential()
model.add(Embedding(max_voc,embedding_vector_lenght))
model.add(LSTM(50))
model.add(Dense(1,activation = "sigmoid"))
sgd = SGD(lr=0.0001)
model.compile(loss = "binary_crossentropy",optimizer = sgd, metrics = ["accuracy"])
print(model.summary())
r = model.fit(data_train, YTrain,validation_data =(data_test,YTest), epochs=50,
batch_size=1,shuffle=False)
```

References

- Annett, M., & Kondrak, G. (2008). “A comparison of sentiment analysis techniques: Polarizing movie blogs”. *Conference of the Canadian Society for Computational Studies of Intelligence* (pp. 25-35). Springer, Berlin, Heidelberg.
- Ante, L. (2021). “How Elon Musk's Twitter Activity Moves Cryptocurrency Markets”. *Available at SSRN 3778844*.
- Barberis, N., Shleifer, A., & Vishny, R. (1998). “A model of investor sentiment”. *Journal of financial economics*, *49*(3), 307-343.
- Batrinca, B., & Treleaven, P. C. (2015). “Social media analytics: a survey of techniques, tools and platforms”. *Ai & Society*, 30(1), 89-116.
- Bengio, Y., Simard, P., & Frasconi, P. (1994). “Learning long-term dependencies with gradient descent is difficult”. *IEEE transactions on neural networks*, *5*(2), 157-166.
- Blockware (2020). “Bitcoin Analysis - Institutional vs Retail Volume Comparison”
- Bollen, J., H. N. Mao and X. J. Zeng (2011). "Twitter mood predicts the stock market”. *Journal of Computational Science* **2**(1): 1-8.
- Britz, D., Goldie, A., Luong, M. T., & Le, Q. (2017). “Massive exploration of neural machine translation architectures “., *arXiv preprint arXiv:1703.03906*.
- Bukovina, J. (2016). "Social media big data and capital markets-An overview.” *Journal of Behavioral and Experimental Finance* **11**: 18-26.
- Cheah, E. T., & Fry, J. (2015). “Speculative bubbles in Bitcoin markets? An empirical investigation into the fundamental value of Bitcoin”., *Economics letters*, *130*, 32-36.
- Cheong, M., & Lee, V. (2010). “A study on detecting patterns in Twitter intra-topic user and message clustering”. *2010 20th International Conference on Pattern Recognition* (pp. 3125-3128). IEEE.
- Cilimkovic, M. (2015). “Neural networks and back propagation algorithm”. *Institute of Technology Blanchardstown, Blanchardstown Road North Dublin*, 15, 1-12.

- Connel, Daniel J. (2015) “Institutional Investing: How Social Media Informs and Shapes the Investing Process”. Greenwich.
- Cresci, S., Lillo, F., Regoli, D., Tardelli, S., & Tesconi, M. (2019). “Cashtag piggybacking: Uncovering spam and bot activity in stock microblogs on Twitter”. *ACM Transactions on the Web* (TWEB), 13(2), 1-27.
- Da, Z., Engelberg, J., & Gao, P. (2011). “In search of attention”. *The Journal of Finance*, 66(5), 1461-1499.
- Daniel, H. Subrahmanyam, 1998 Daniel, K., Hirschleifer, D., & Subrahmanyam, A.,(1998, December). “Investor psychology and security market under-and overreactions”. *The Journal of Finance*, *53*(6), 1839-1885.
- De Roo, G., & Lueck, E. (2020). “The effect of Elon Musk's tweets on Tesla stock price.”
- Fama, E. F. (1965). "The Behavior of Stock-Market Prices”. *Journal of Business* **38**(1): 34-105.
- Fang, X., & Zhan, J. (2015). “Sentiment analysis using product review data”. *Journal of Big Data*, 2(1), 5.
- Ferrara, E., Varol, O., Davis, C., Menczer, F., & Flammini, A. (2016). “The rise of social bots”., *Communications of the ACM*, 59(7), 96-104.
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). “*Deep learning*”. MIT press.
- Granger, C. W. (1969). Investigating causal relations by econometric models and cross-spectral methods. *Econometrica: journal of the Econometric Society*, 424-438.
- Hatzivassiloglou, V., & Wiebe, J. (2000). Effects of adjective orientation and gradability on sentence subjectivity. In *COLING 2000 Volume 1: The 18th International Conference on Computational Linguistics*.
- Henry, E. (2008). “Are investors influenced by how earnings press releases are written?”. *The Journal of Business Communication* (1973), 45(4), 363-407.
- Hochreiter, S., & Schmidhuber, J. (1997). “Long short-term memory”. *Neural computation*, *9*(8), 1735-1780.
- Hougan, M. (2018). “What Gold’s History teaches us about bitcoin as a store of value”.

- Hutto, C. J., Yardi, S., & Gilbert, E. (2013). “A longitudinal study of follow predictors on twitter”. *Proceedings of the sigchi conference on human factors in computing systems* (pp. 821-830).
- Hutto, Clayton, and Eric Gilbert. "Vader: A parsimonious rule-based model for sentiment analysis of social media text”. *Proceedings of the International AAAI Conference on Web and Social Media*. Vol. 8. No. 1. 2014.
- Jungherr, A., P. Jurgens and H. Schoen (2012). "Why the Pirate Party Won the German Election of 2009 or The Trouble With Predictions: A Response to Tumasjan, A., Sprenger, T. O., Sander, P. G., & Welpe, I. M. "Predicting Elections With Twitter: What 140 Characters Reveal About Political Sentiment" ". *Social Science Computer Review* **30**(2): 229-234.
- Karabulut, Y., 2013: "Can Facebook predict stock market activity?".
- Kalra, S., & Prasad, J. S. (2019, February). “Efficacy of news sentiment for stock market prediction”. In *2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon)* (pp. 491-496). IEEE.
- Kennedy, A., & Inkpen, D. (2006). “Sentiment classification of movie reviews using contextual valence shifters”. *Computational intelligence*, *22*(2), 110-125.
- Keskar, N. S., McCann, B., Varshney, L. R., Xiong, C., & Socher, R. (2019). Ctrl: A conditional transformer language model for controllable generation. *arXiv preprint arXiv:1909.05858*.
- Kim, Y. B., Kim, J. G., Kim, W., Im, J. H., Kim, T. H., Kang, S. J., & Kim, C. H. (2016). “Predicting fluctuations in cryptocurrency transactions based on user comments and replies”. *PloS one*, *11*(8), e0161197.
- Kirlić, A., Orhan, Z., Hasovic, A., & Kevser-Gokgol, M. (2018). “Stock market prediction using Twitter sentiment analysis”. *Invention Journal of Research Technology in Engineering & Management (IJRTEM)*, *2*(1), 01-04.
- Kraaijeveld, O., & De Smedt, J. (2020). “The predictive power of public Twitter sentiment for forecasting cryptocurrency prices”. *Journal of International Financial Markets stitutions and Money*, *65*, 101188.
- Kralj Novak, P., Smailović, J., Sluban, B., & Mozetič, I. (2015). “Sentiment of emojis”. *PloS one*, *10*(12), e0144296.

- Khan, Wahab, et al. "A survey on the state-of-the-art machine learning models in the context of NLP". *Kuwait journal of Science* 43.4 (2016).
- Kietzmann, J. H., Hermkens, K., McCarthy, I. P., & Silvestre, B. S. (2011). "Social media? Get serious! Understanding the functional building blocks of social media". *Business horizons*, 54(3), 241-251.
- Kraus, M., & Feuerriegel, S. (2017). "Decision support from financial disclosures with deep neural networks and transfer learning". *Decision Support Systems*, *104*, 38-48.
- Kwak, H., Lee, C., Park, H., & Moon, S. (2010). "What is Twitter, a social network or a news media?". *Proceedings of the 19th international conference on World wide web* (pp. 591-600).
- Lapedes, A., & Farber, R. (1987). "Nonlinear signal processing using neural networks: Prediction and system modelling". No. LA-UR-87-2662; CONF-8706130-4.
- Lee, Pei En. "The empirical study of investor sentiment on stock return prediction". *International Journal of Economics and Financial Issues* 9.2 (2019): 119.
- Levallois, C. (2013). "Umigon: sentiment analysis for tweets based on lexicons and heuristics".
- Li, X., Xie, H., Chen, L., Wang, J., & Deng, X. (2014). "News impact on stock price return via sentiment analysis". *Knowledge-Based Systems*, *69*, 14-23
- Liu, Bing, and Lei Zhang (2012). "A survey of opinion mining and sentiment analysis". *Mining text data*. Springer, Boston, MA, 415-463
- Liu, Y., & Tsyvinski, A. (2021). "Risks and returns of cryptocurrency". *The Review of Financial Studies*, *34*(6), 2689-2727.
- Loughran, T., & McDonald, B. (2011). "When is a liability not a liability? Textual analysis, dictionaries, and 10-Ks". *The Journal of Finance*, 66(1), 35-65.
- Mao, H., Counts, S., & Bollen, J. (2011). "Predicting financial markets: Comparing survey, news, twitter and search engine data". arXiv preprint arXiv:1112.1051.
- Merton, R. C. (1987). "A Simple-Model of Capital-Market Equilibrium with Incomplete Information". *Journal of Finance* **42**(3): 483-510.

- Nakamoto, S. (2008). “Bitcoin whitepaper”.
- Nakisa, B., Rastgoo, M. N., Rakotonirainy, A., Maire, F., & Chandran, V. (2018). “Long short term memory hyperparameter optimization for a neural network based emotion recognition framework”. *IEEE Access*, *6*, 49325-49338.
- Nisar, T. M., & Yeung, M. (2018). “Twitter as a tool for forecasting stock market movements: A short-window event study”. *The journal of finance and data science*, 4(2), 101-119.
- Pang, Bo, and Lillian Lee. "Opinion mining and sentiment analysis”. *Comput. Linguist* 35.2 (2009): 311-312.
- Raamkumar, A. S., S. G. Tan and H. L. Wee (2020). "Measuring the Outreach Efforts of Public Health Authorities and the Public Response on Facebook During the COVID-19 Pandemic in Early 2020: Cross-Country Comparison”. *Journal of Medical Internet Research* **22**(5).
- Ruder, S. (2016). “An overview of gradient descent optimization algorithms”. arXiv preprint arXiv:1609.04747.
- Salton, G., & Buckley, C. (1988). “Term-weighting approaches in automatic text retrieval”. *Information processing & management*, *24*(5), 513-523.
- Schumaker, R. P., Zhang, Y., Huang, C. N., & Chen, H. (2012). “Evaluating sentiment in financial news articles”. *Decision Support Systems*, *53*(3), 458-464.
- Shleifer, A., & Vishny, R. W. (1997). “The limits of arbitrage”. *The Journal of finance*, *52*(1), 35-55.
- Siganos, A., E. Vagenas-Nanos and P. Verwijmeren (2014). "Facebook's daily sentiment and international stock markets”. *Journal of Economic Behavior & Organization* **107**: 730-743.
- Souma, W., I. Vodenska and H. Aoyama (2019). "Enhanced news sentiment analysis using deep learning methods”. *Journal of Computational Social Science* **2**(1): 33-46.
- Sohangir, S., Wang, D., Pomeranets, A., & Khoshgoftaar, T. M. (2018). “Big Data: Deep Learning for financial sentiment analysis”. *Journal of Big Data*, *5*(1), 1-25.

- Sprenger, T. O., Tumasjan, A., Sandner, P. G., & Welpe, I. M. (2014). “Tweets and trades: The information content of stock microblogs”. *European Financial Managemen*t, 20(5), 926-957.
- Statista (2020). “Number of monthly active Twitter users worldwide from 1st quarter 2010 to 1st quarter 2019”.
- Tetlock, P. C., Saar-Tsechansky, M., & Macskassy, S. (2008). “More than words: Quantifying language to measure firms' fundamentals”. *The Journal of Finance*, 63(3), 1437-1467.
- Tversky, A. and D. Kahneman (1981). "The Framing of Decisions and the Psychology of Choice”. *Science* **211**(4481): 453-458.
- Vargas, M. R., De Lima, B. S., & Evsukoff, A. G. (2017, June). “Deep learning for stock market prediction from financial news articles”. In *2017 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA)* (pp. 60-65). IEEE.
- Wang, Y. and J. E. Taylor (2018). "Coupling sentiment and human mobility in natural disasters: a Twitter-based study of the 2014 South Napa Earthquake”. In *Natural Hazards* **92**(2): 907-925.
- Weiss, G., Goldberg, Y., & Yahav, E. (2018). “On the practical computational power of finite precision RNNs for language recognition”. *arXiv preprint arXiv:1805.04908*.
- Wu, T., Liu, S., Zhang, J., & Xiang, Y. (2017). “Twitter spam detection based on deep learning”. *Proceedings of the australasian computer science week multiconference* (pp. 1-8).
- Xing, F. Z., E. Cambria and R. E. Welsch (2018). "Natural language based financial forecasting: a survey ”. *Artificial Intelligence Review* **50**(1): 49-73.
- Yuan, X., Li, L., & Wang, Y. (2019). “Nonlinear dynamic soft sensor modeling with supervised long short-term memory network”. *IEEE transactions on industrial informatics*, *16*(5), 3168-3176.
- Zhang, J., Chen, Y., Cheung, B., & Olshausen, B. A. (2019). “Word Embedding Visualization Via Dictionary Learning”. *arXiv preprint arXiv:1910.03833*.

- Zhou, P., Qi, Z., Zheng, S., Xu, J., Bao, H., & Xu, B. (2016). “Text classification improved by integrating bidirectional LSTM with two-dimensional max pooling”. *arXiv preprint arXiv:1611.06639.*