Analysis of Bitcoin and Ethereum Tweets

Prakhar Sinha - 41985391

Krish Kohli - 22379440

Utkarsh Misra - 89748669

Contents

Introduction

As globalization and the rise of the internet during the 90s brought the world closer, it also lead to a vast increase in the exchange of goods and services. With the advent of the 90s tech boom, there were several attempts at creating a decentralized, digital currency to make global transactions efficient, fluid and secure. However, the lack of technological advancements became an unavoidable hindrance to its progress for nearly two decades. However, in 2009, a consortium of programmers under the alias of Satoshi Nakamoto introduced Bitcoin, which has since opened the floodgates for a wealth of cryptocurrencies entering the market in the last ten years.

Simultaneously, the rise of social media has led to a profound impact in global forces that shape the world of both technology and global economy. Whether it be stock prices or voter turnout, an influential tweet or a viral video can shift the tide of public sentiment towards all types of issues, serious or innocuous.

With the recent market uptrend in prices of cryptocurrencies in 2017 followed by the downturn in 2018, conversations about cryptocurrencies have always grabbed headlines and especially developed a large ecosystem of users on social media platforms like Twitter, Reddit and Facebook. The rising popularity of cryptocurrencies along with the pervading impact of social media has brought up important concerns and raised pertinent questions in the online community. For instance, there has been much speculation in the past year about the influence of social media on the impact of fluctuation cryptocurrency prices. The correlation (if one exists) between price and online influence is an important aspect to consider for crypto enthusiasts, one that we will attempt to delve into. In addition, due to the quick, decentralized method by which these currencies are exchanged, online platforms such as twitter have become hotbeds for buyers, sellers and scammers. Through text and sentiment analysis, we will attempt to uncover the proportion of these groups within the larger twitter community.

For our group project we have chosen key words “Bitcoin” and “Ethereum” to collect our 10,000 tweets and performed analysis on them. The timeline for our tweet collection is 10hrs over October 5th and 6th for both bitcoin and Ethereum.

Bitcoin

Bitcoin, the first out of the cryptocurrencies, was born back in 2008. Bitcoin offers the promise of lower transaction fees than traditional online payment mechanisms and is operated by a decentralized authority, unlike government issued currencies. There are no physical bitcoins, only a ledger associated with public and private keys. It is the largest out of the cryptocurrencies.

Ethereum

Ethereum is an open source decentralized system blockchain-based distributed computing platform and operating system featuring smart contract functionality. Ether is a cryptocurrency whose blockchain is generated by the Ethereum platform. The project was publicly announced in January 2014, with the core team consisting of Vitalik Buterin, Mihai Alisie, Anthony Di Iorio, Charles Hoskinson, Joe Lubin and Gavin Wood.

Description of Code:

Using Twython Streamer, we obtained real time tweets using our two keywords, after which we stored them in separate JSON files. After that, we conducted preliminary analysis by going over each tweet, and analysing the text by using a number of functions to evaluate different aspects about the collection of tweets. For instance, we used functions to clean text (remove punctuation and numbers), get the most popular tweet, the most popular twitter user, most popular hashtags and so on. A snapshot of some of our functions are provided below:

```
def most_popular_without_stopwords(tweets):
    'Returns the Ten Most Popular keywords from a list of Tweets minus the stopwords'
    excude_punctuation = set(string.punctuation)
    exclude_digits = set(string.digits)
    tweet_words_set = [ ]
    for tweet in tweets:
        tweet_text = tweet['text']
        clean_tweet = clean_text(tweet_text)
        clean_tweet = re.sub(' +', ' ', clean_tweet)
        stopwords = nltk.corpus.stopwords.words('english')
        stopwords.extend(["RT", "co", "https", "We",  "https", "We", "via", "'", "’", "…", "I"])
        #print(stopwords)
        tweet_words = word_tokenize(clean_tweet)
        for word in tweet_words:
            if word not in stopwords:
                tweet_words_set.append(word)
    counter = Counter(tweet_words_set)
    return counter.most_common(10)
```

```
def most_popular_tweet(tweets):
    'Returns the most popular tweet from a list of tweets'
    popular_tweet_set = { }
    for tweet in tweets:
        tweet_id = tweet['id']
        quote_count = tweet['quote_count']
        retweet_count = tweet['retweet_count']
        reply_count = tweet['reply_count']
        favourite_count = tweet['favorite_count']
        try:
            quote_count += tweet['retweeted_status']['quote_count']
        except:
            quote_count = tweet['quote_count']
        try:
            retweet_count += tweet['retweeted_status']['retweet_count']
        except:
            retweet_count = tweet['retweet_count']
        try:
            reply_count += tweet['retweeted_status']['reply_count']
        except:
            reply_count = tweet['reply_count']
        try:
            favourite_count += tweet['retweeted_status']['favorite_count']
        except:
            favourite_count = tweet['favorite_count']
        tweet_score = quote_count + retweet_count + reply_count + favourite_count
        #print(tweet_score)
        popular_tweet_set[tweet_id] = tweet_score
    return max(popular_tweet_set.items(), key = operator.itemgetter(1))[0]

with open(os.path.join('downloaded_tweets/tweet_stream_ethereum_10000.json')) as file:
    data = json.load(file)
```

```
def extract_locations(tweets, keyword):
    'Opens up a WordCloud showing the origins/location of the tweets with larger font size meaning higher frequency'
    location_stopwords = ['Earth', 'Blockchain', 'Assist', 'assist', 'blockchain', 'Worldwide', 'City', 'city']
    locationset = ''
    for tweet in tweets:
        location = tweet['user']['location']
        if location is not None and location not in location_stopwords:
            locationset += ' {}'.format(location)
    wordcloud = WordCloud(max_font_size=40).generate(locationset)
    plt.figure()
    plt.imshow(wordcloud)
    plt.axis('off')
    plt.savefig(os.path.join('insights/{}_location.pdf'.format(keyword)))
    plt.show()

with open(os.path.join('downloaded_tweets/tweet_stream_bitcoin_10000.json'), 'r') as file:
    tweets = json.load(file)
    extract_locations(tweets, 'bitcoin')

with open(os.path.join('downloaded_tweets/tweet_stream_ethereum_10000.json'), 'r') as file:
    tweets = json.load(file)
    extract_locations(tweets, 'ethereum')
```

We then used the results from our functions to visualize some of our preliminary highlights (given under the 'Preliminary Analysis' section).

Preliminary analysis

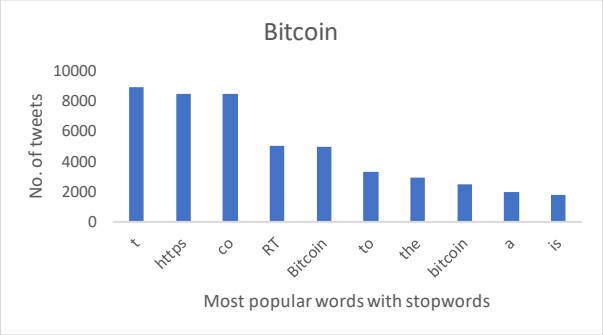
2a)

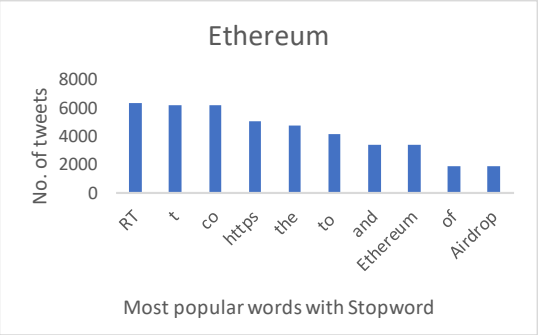
The data from the preliminary analysis shows that RT, t, co and to feature among the most popular keywords in both Bitcoin and Ethereum tweets. For this part we imported the nltk stopwords library and went through all the tweets and stored the words in a list with/without stopwords. After which, we tokenized the tweets and returned the top 10 from each set.

10 most popular words with stopwords Bitcoin vs Ethereum

Stopwords for Bitcoin	No.of tweets
t	8928
https	8470
co	8470
RT	5040
Bitcoin	4926
to	3267
the	2921
bitcoin	2446
a	1978
is	1797

Stopwords for Ethereum	No.of tweets
RT	6311
t	6174
co	6159
https	5040
the	4744
to	4181
and	3406
Ethereum	3399
of	1900
Airdrop	1859



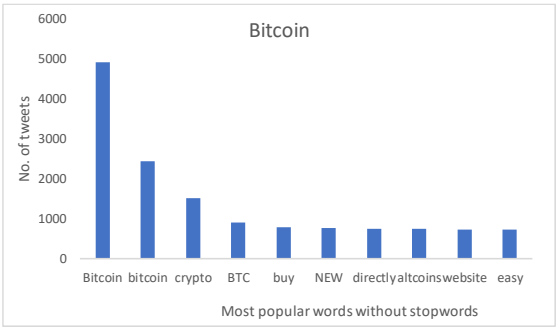


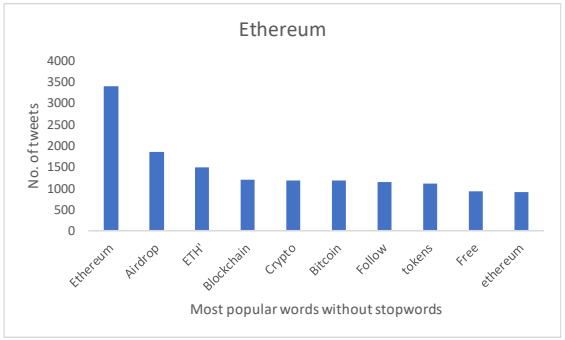
After inserting a list of stopwords to remove we found the most popular words without stopwords mentioned in Bitcoin and Ethereum tweets. The top mentioned word tends to be the cryptocurrency itself, along with words like crypto appearing in both tables.

10 most popular words without stopwords Bitcoin vs Ethereum

Without stopwords for Bitcoin	No.of tweets
Bitcoin	4926
bitcoin	2446
crypto	1507
BTC	893
buy	785
NEW	759
directly	743
altcoins	741
website	734
easy	727

Without stopwords for Ethereum	No.of tweets
Ethereum	3399
Airdrop	1859
ETH'	1491
Blockchain	1196
Crypto	1190
Bitcoin	1179
Follow	1141
tokens	1104
Free	934
ethereum	902





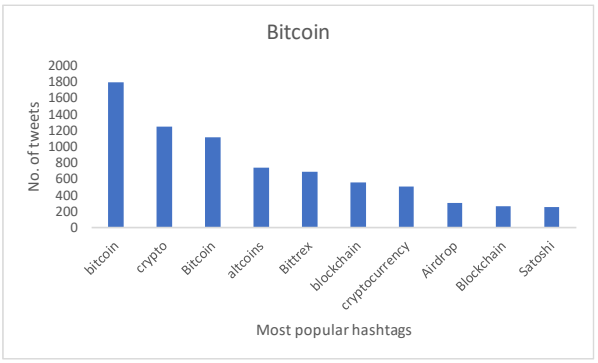
2b)

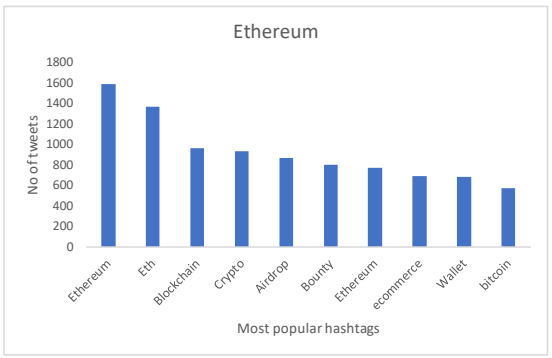
10 most popular hashtags Bitcoin vs Ethereum

Here we iterated through the tweets and extracted the data from the index tweet['entities'] ['hashtags']. The most popular hashtags for Bitcoin and Ethereum tweets in our set of 10k tweets are cryptocurrencies themselves, followed by crypto and blockchain.

Most popular hashtages for Bitcoin	No.of tweets
bitcoin	1790
crypto	1244
Bitcoin	1118
altcoins	736
Bittrex	688
blockchain	553
cryptocurrency	504
Airdrop	303
Blockchain	262
Satoshi	248

Most popular hashtags for Ethereum	No.of tweets
Ethereum	1582
Eth	1364
Blockchain	957
Crypto	932
Airdrop	861
Bounty	801
Ethereum	771
ecommerce	690
Wallet	682
bitcoin	572





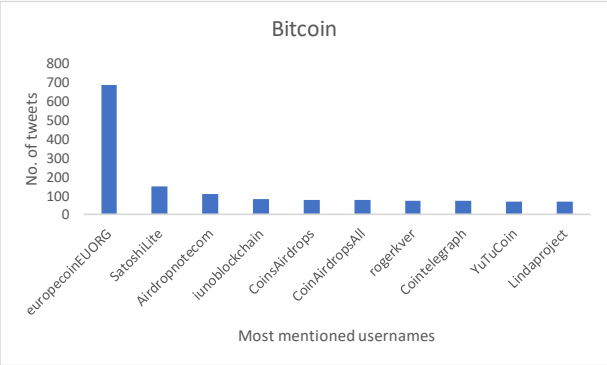
2c)

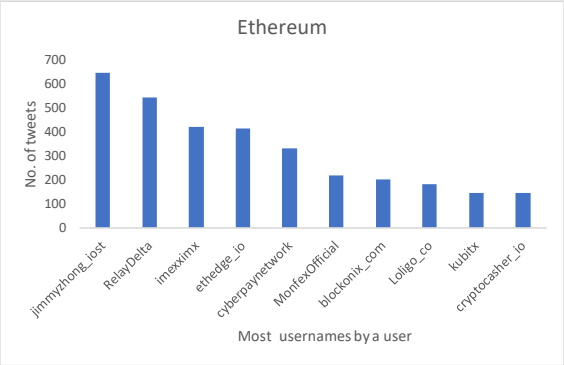
10 most popular usernames mentioned for Bitcoin vs Ethereum

Here we iterated through the tweets and extracted the data from the index tweet['entities']['user_mentions']. The most usernames mentioned for Bitcoin and Ethereum tweets in our set of 10k tweets are the handles europecoin, EUORG and jimmyzhong_iost.

Most popular usernames mentioned for Bitcoin	No.of tweets
europecoinEUORG	687
SatoshiLite	150
Airdropnotecom	110
iunoblockchain	81
CoinsAirdrops	77
CoinAirdropsAll	76
rogerkver	74
Cointelegraph	73
YuTuCoin	71
Lindaproject	68

Most popular usernames mentioned for Ethereum	No.of tweets
jimmyzhong_iost	646
RelayDelta	544
imexximx	422
ethedge_io	415
cyberpaynetwork	332
MonfexOfficial	219
blockonix_com	201
Loligo_co	182
kubitx	146
cryptocasher_io	145





2d) To calculate the most frequently tweeting person we iterated through the tweets and added the users in a list to get the user having the highest count in the list.

The most frequently tweeting person (tweet author) about Bitcoin is the handle Airdropnotecom with 110 tweets.

The most frequently tweeting person (tweet author) about Ethereum is the handle workwithfintech with 64 tweets.

2e) To calculate the influence score we iterated through the tweets, and then went to retweeted info dictionary. We then added up the three values of reply count, favorite count and quote count.

The most influential tweet for Bitcoin is @coinbundlecomfrom with the text

“Before we ask, \u201cWhen moon?\u201d perhaps we should be asking, \u201cWhen Bermuda?\u201d \n\nThese global locations have some of the highe\u2026" and tweet ID no. 1048736972368310272. The tweet was from Indonesia and had the influential count of 6496.

The most popular tweet for Ethereum is from edge capital a hedge fund in Zurich promising Ethereum payouts. The influential count for this tweet is 10,632 with the tweet ID no. 1048513921970135047. The text was “@ethedge_io: If you own an Ethereum address with over $100 in Ether and / or tokens, you can receive our hedge fund payouts of up to $28\u2026"

Wordcloud:

To get the Wordcloud for tweets of Bitcoin and Ethereum, we initially collected the tweets, cleaned it up by removing punctuation and numbers, performed stemming and lemmitization, and then removed the stop words. Finally, we then used the 'wordcloud' module to generate a WordCloud for this list of words which depicted the location of the 10,000 tweets with larger font size implying higher frequency (Refer to Figure).

```
def clean_text(unclean_text):
    'Takes string as input and returns it without numbers or punctuation'
    table_p = str.maketrans(p, len(p) * " ")
    table_d = str.maketrans(d, len(d) * " ")
    text_without_punctuation = unclean_text.translate(table_p)
    text_without_punctuation_numbers = text_without_punctuation.translate(table_d)
    return text_without_punctuation_numbers

stopwords = nltk.corpus.stopwords.words('english')
stopwords.append('might')
with open(os.path.join('downloaded_tweets/tweet_stream_{}_{}.json'.format(sys.argv[1], sys.argv[2])), 'r') as file:
    tweets = json.load(file)

excude_punctuation = set(string.punctuation)
exclude_digits = set(string.digits)
tweet_words_set = [ ]
for tweet in tweets:
    tweet_text = tweet['text']
    clean_tweet = clean_text(tweet_text)
    clean_tweet = re.sub(' +', ' ', clean_tweet)
    stopwords = nltk.corpus.stopwords.words('english')
    stopwords.append("RT")
    #print(stopwords)
    tweet_words = word_tokenize(clean_tweet)

text2 = ''
for word in tweet_words:
    if len(word) == 1 or word in stopwords:
        continue
    text2 += ' {}'.format(word)
wordcloud2 = WordCloud(max_font_size=40).generate(text2)
plt.figure()
plt.imshow(wordcloud2)
plt.axis('off')
plt.show()
```

Bitcoin Wordcloud

Ethereum Wordcloud

Sentiment Analysis

In order to conduct our sentiment analysis, we used TextBlob and matplotlib to get the subjectivity and polarity score and visualize it by frequency on a histogram. We first extracted the tweet text using functions and then appended their subjectivity and polarity scores in a list which we then plotted as a histogram using the Matplotlib module. The subjectivity score measures how subjective or objective a particular word/sentence is while the polarity score measures how 'positive or negative' a word/sentence sounds.

```
for w in words: #Words
    ws = bl(w)
    words_subj.append(ws.sentiment.subjectivity) #Word Subjectivity list
    words_pol.append(ws.sentiment.polarity) #Word Polarity lists
sentences_subj = []
sentences_pol = []

for s in sentences: #Sentences
    ss = bl(s)
    sentences_subj.append(ss.sentiment.subjectivity) #Sentencce Subjectivity List
    sentences_pol.append(ss.sentiment.polarity) #Sentence Polarity List

subjectivity_avg = sum(sentences_subj)/len(sentences_subj)
polarity_avg = sum(sentences_pol)/len(sentences_pol)
print('Average subjectivity score is: {}'.format(subjectivity_avg))
print('Average polarity score is: {}'.format(polarity_avg))
plt.hist(sentences_subj, bins=10)
plt.xlabel('subjectivity score')
plt.ylabel('tweet count')
plt.grid(True)
#plt.savefig('subjectivity.pdf')
plt.show()
plt.hist(sentences_pol, bins=10)
plt.xlabel('polarity score')
plt.ylabel('tweet count')
plt.grid(True)
#plt.savefig('polarity.pdf')
plt.show()
```

Specifically, for our analysis, we targeted two groups and one subgroup to conduct a more focussed analysis on the use of Twitter as a platform for buyers, sellers, and scammers.

Group I: Traders

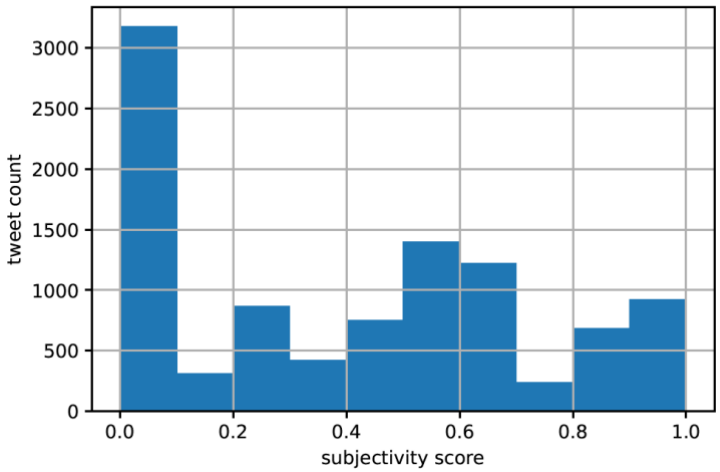
For the first we group, we filtered our list of tweets according the following keywords: [buy, sell, trade, finance, bear, buy, ledger, chart, cryptotrading, cryptotrader, daytrading]. From this smaller pool of tweets, we then conducted sentiment analysis on the tweets to get subjectivity and the polarity spread on a histogram. By doing this, we hope to get a more in-depth look at the extent of emotional language and fact-based wording used by traders on the twitter 'marketplace.' We also measured the proportion of 'traders' out of the total twitter userbase, as per our self-described criterion. This way, we hope to get a rough idea of how many people use twitter as a hub for trading on a daily basis.

Finally, we also broke up the trader group into two different subgroups: buyers and sellers, and then compared the number of tweets for each subgroup. This was done to get an approximation of the buying-to-selling ratio on twitter.

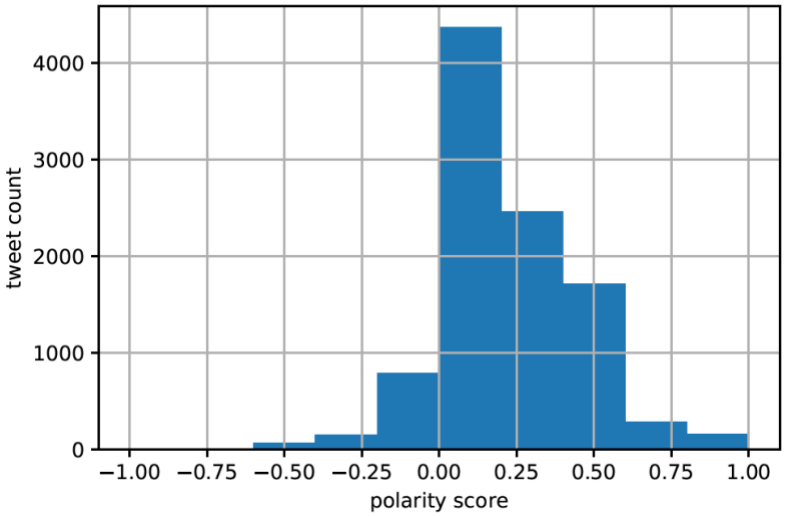
Group II: Scammers

It is estimated that 30% of the twitter userbase are either human scammers or bots which are created specifically for scamming (or other nefarious) purposes. We aim to get our own estimate of this speculative number by filtering out the total tweets based on the following keywords (known to be used in popular scamming ploys): [giveaway, airdrop, free, quick, earn, fast, limited, payment, opportunity]. Again, we applied sentiment analysis on this smaller batch of tweets to better gauge the tricks and tactics used by scammers to dupe people into sending bitcoin (or Ethereum). For instance, do scammers often use flowery, positive language to create a sense of well-being to lure their victims? Do they often stay away from fact-based arguments and strategies to arouse people's attention? If so, we aim to find out whether we will be able to discover them through text and sentiment analysis. Finally, we performed a calculation to get an approximation of the scammer group as a percentage of the total.

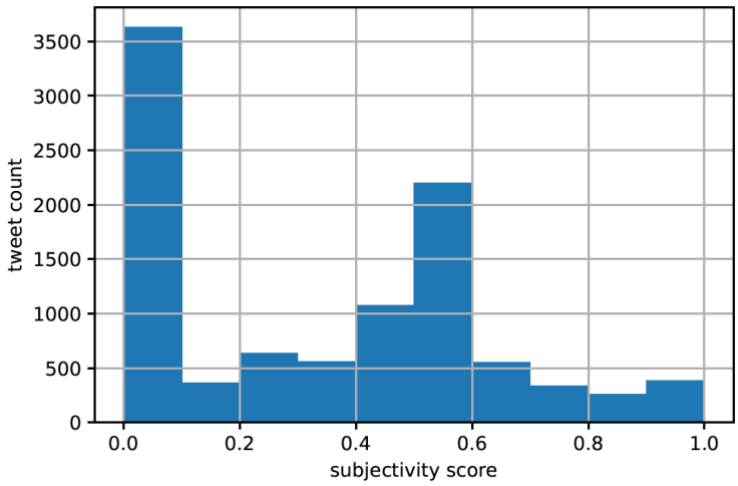
Ethereum Subjectivity



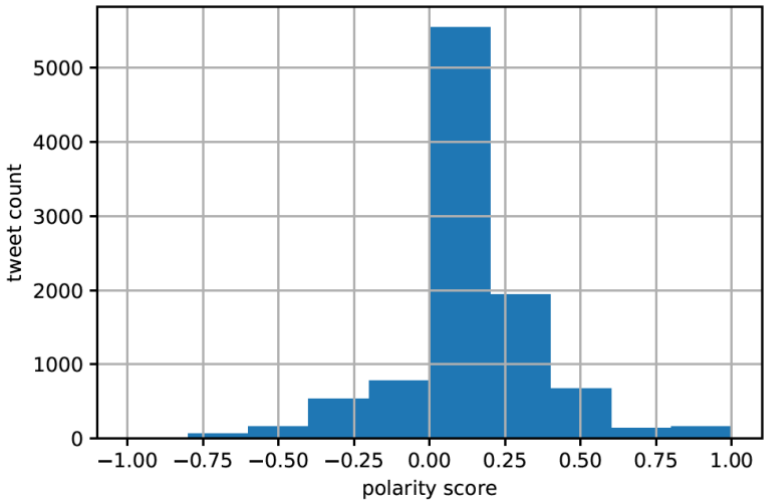
Ethereum Polarity



Bitcoin Subjectivity



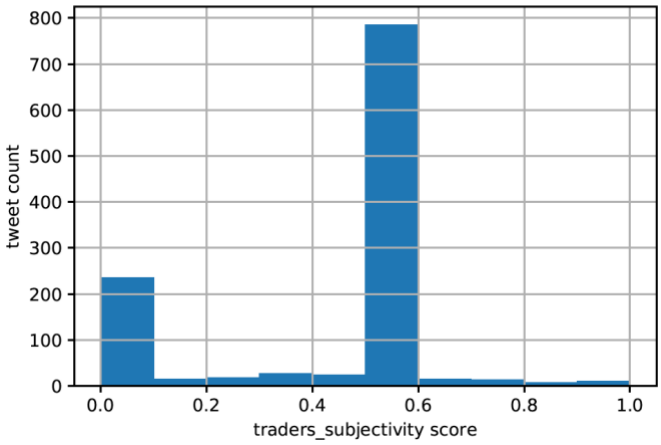
Bitcoin Polarity



Insights and Conclusions

Group I: Traders

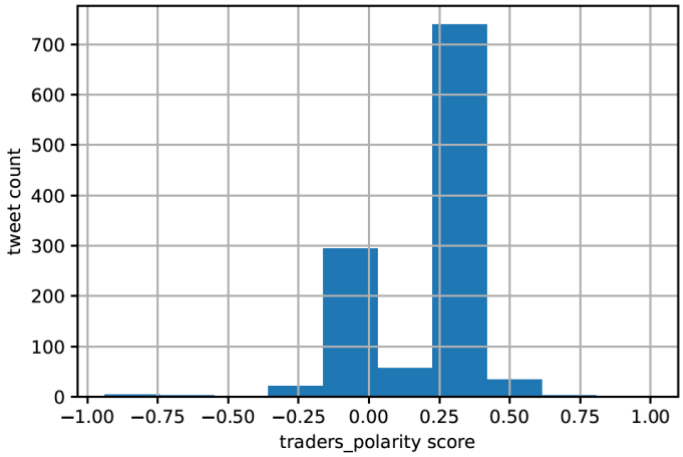
The graphs below show us the polarity and subjectivity histogram for the trader group. From the subjectivity graph, we can see that a majority of the tweets lie within two subgroups: one centring around 0 subjectivity (or 100% objectivity) and one around 0.5 subjectivity. From this result, we can assume that the first group usually sticks to facts-based verbiage and technically sound terms when conducting their trades online. From a brief glance over some of these tweets, we noticed that at least a few of these were reporting findings in the current market (e.g. present Bitcoin Price Index) or sticking to more objective terminology such as numbers and figures. While this finding is not surprising, the second one is more inclined towards what we would expect from traders who wish to present a positive light when advertising (or selling) their asset. Embellishing facts and figures to sell your asset is surely an age-old sales tactic, one that we see in this second group within the trader



Bitcoin Traders Subjectivity

As for the polarity, we saw two subgroups for the traders. As expected, we saw that the larger chunk of the tweets drifted towards more positive (around 0.3) tweets. This is expected from traders, who have infused their sentences with positivity and optimism to sell their goods.

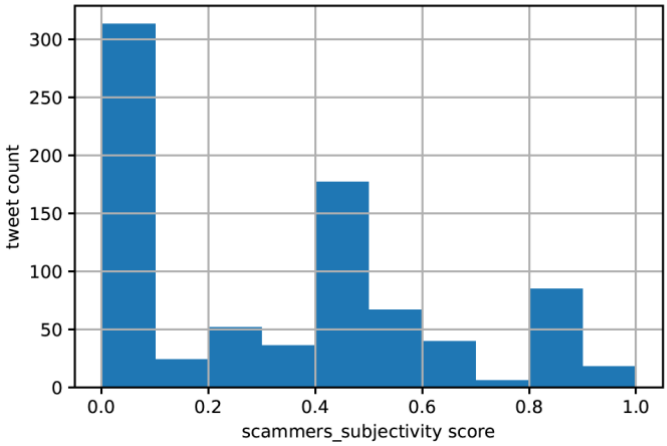
Bitcoin Traders Polarity



Group I: Scammers

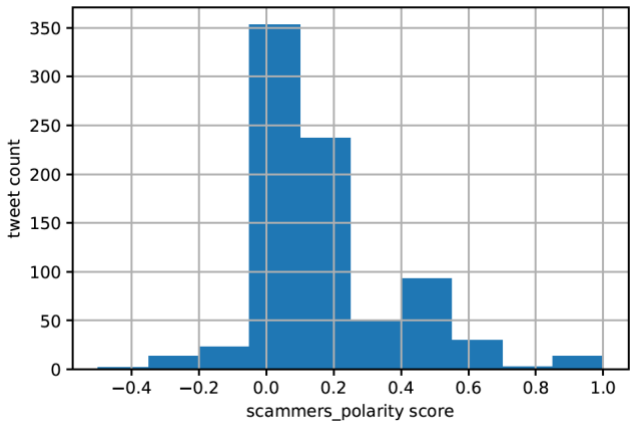
The graph below shows us the subjectivity scores for scammers. As we can see, there's quite a spread on it, which implies that their choice of words and sentence construction is steering far away from fact-based ideologies and arguments. Particularly, within the three subgroups in this graph, we see one concentrated at one end, making it highly subjective. This is in line with our expectation that many of these scammers lie, trick and dupe people into believing their stories so that they can trick them into 'donating' bitcoins. The contrast between the subjectivity spread between genuine traders and scammers is stark and differentiable just by visually glancing over it. This is a promising result, because it reaffirms the that analytics/machine learning models can be implemented to identify bots or scammers if sentimental analysis shows us such a difference just by looking.

Bitcoin Scammers Subjectivity



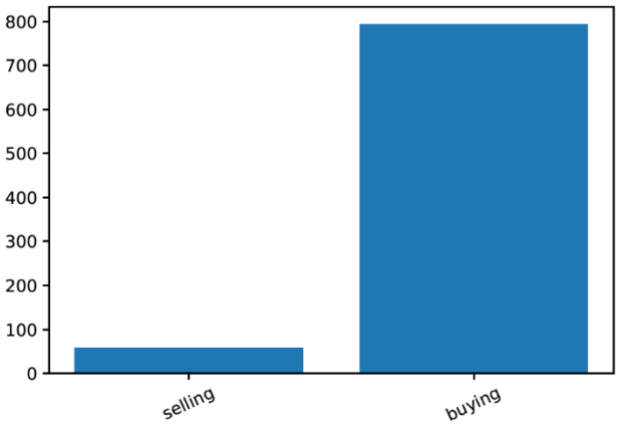
Similar to the polarity spread in Group I, we found that two distinct subgroups popped up when looking at the scammer polarity histogram. Particularly, one occupying the '0' score and one slightly more positive, maxing at around 0.3. In fact, on comparing the same graph from Group I, we see a lower count for the positive tweets for scammers. While this is counterintuitive at first, because we assume that scammers will likely inject their sentences with false promises and flowery language to lure viewers in, this result is by far the most interesting. The fact that a majority are 'neutral' almost looks like the scammers 'are trying too hard' to sound neutral. All in all, this result is definitely one that has the most scope for exploration and research.

Bitcoin Scammers Polarity



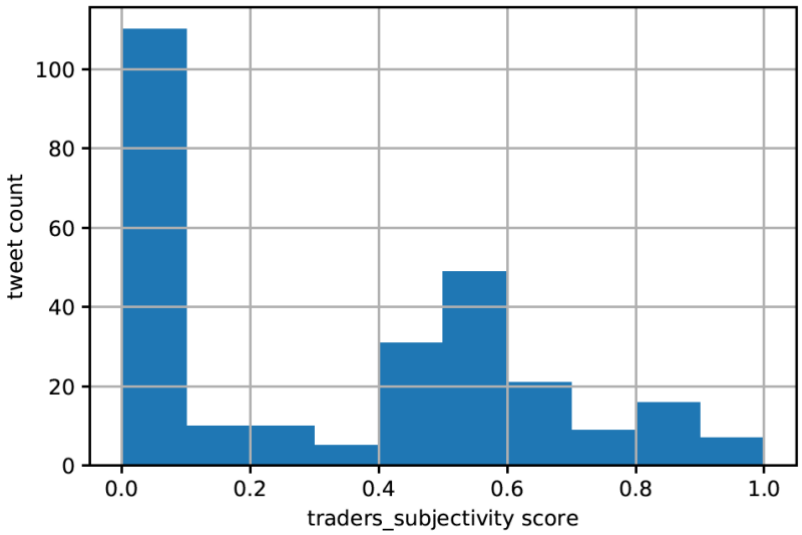
Finally, we observed the following numbers for the buyer versus seller subgroup, giving us a final ratio of 16:1 in favour of buyers.

Seller-to-Buyer Tweet comparison

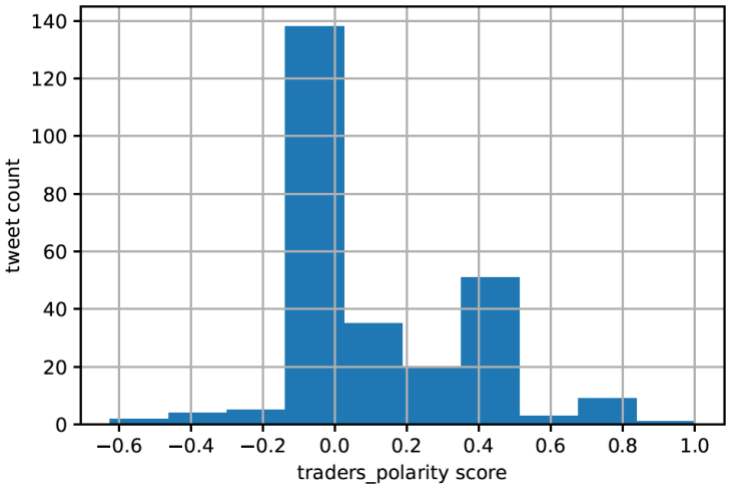


Given below are the graphs for Group I and II for Ethereum:

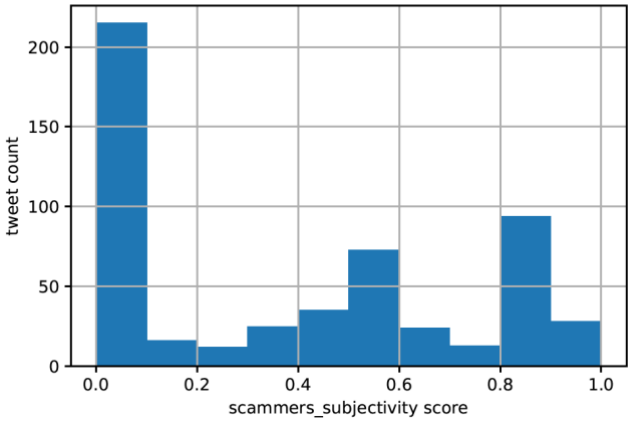
Ethereum Traders Subjectivity



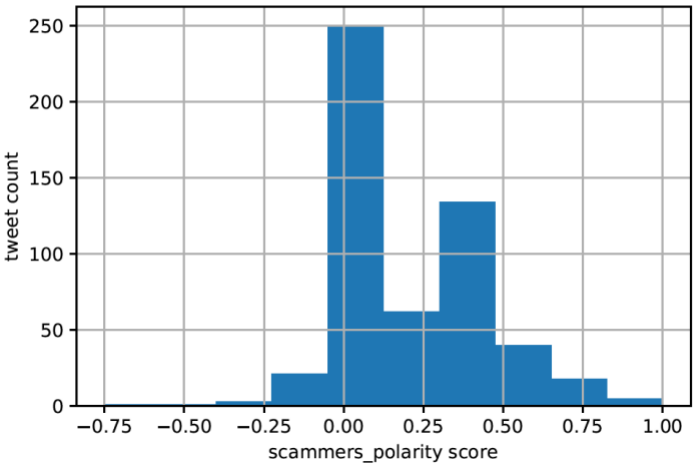
Ethereum Trades Polarity



Ethereum Scammers Subjectivity



Ethereum Scammers Polarity



Location WordCloud

We also formed a WordCloud for the twitter location to get insights from where are the most tweets about our keywords are coming(given below). For example, for both of our keywords, you can see that United Kingdom and United States were amongst some of the most popular origins for those tweeting about Bitcoin and Ethereum.

Bitcoin wordcloud

Ethereum wordcloud

Sources and References

- BAIT 508; J_Ch20_Text Analytics (TextBlob, WordCloud by Professor Gene Moo Lee
- BAIT 508; JG_Ch09_Get Data (Api, Web Scraping, Reading Files, Sentiment Analysis) by Professor Gene Moo Lee
- https://hackernoon.com/text-processing-and-sentiment-analysis-of-twitter-data-22ff5e51e14c
- https://topdogsocialmedia.com/social-media-cryptocurrency-and-blockchain/
- https://medium.com/ethos-io/cryptocurrency-sentiment-analysis-9009b4eaa422
- https://ambcrypto.com/ethereum-eth-sentiment-analysis-april-25/
- https://www.proofpoint.com/us/threat-insight/post/money-nothing-cryptocurrency-giveaways-net-thousands-scammers-0
- https://www.cryptoglobe.com/latest/2018/07/crypto-sentiment-analysis-twitter-indicator-bullish-at-5-month-high/
- https://cointelegraph.com/bitcoin-for-beginners/what-are-cryptocurrencies#history
- https://stackoverflow.com/questions/9343929/how-to-stem-words-in-python-list
- https://stackoverflow.com/questions/8270092/remove-all-whitespace-in-a-string-in-python