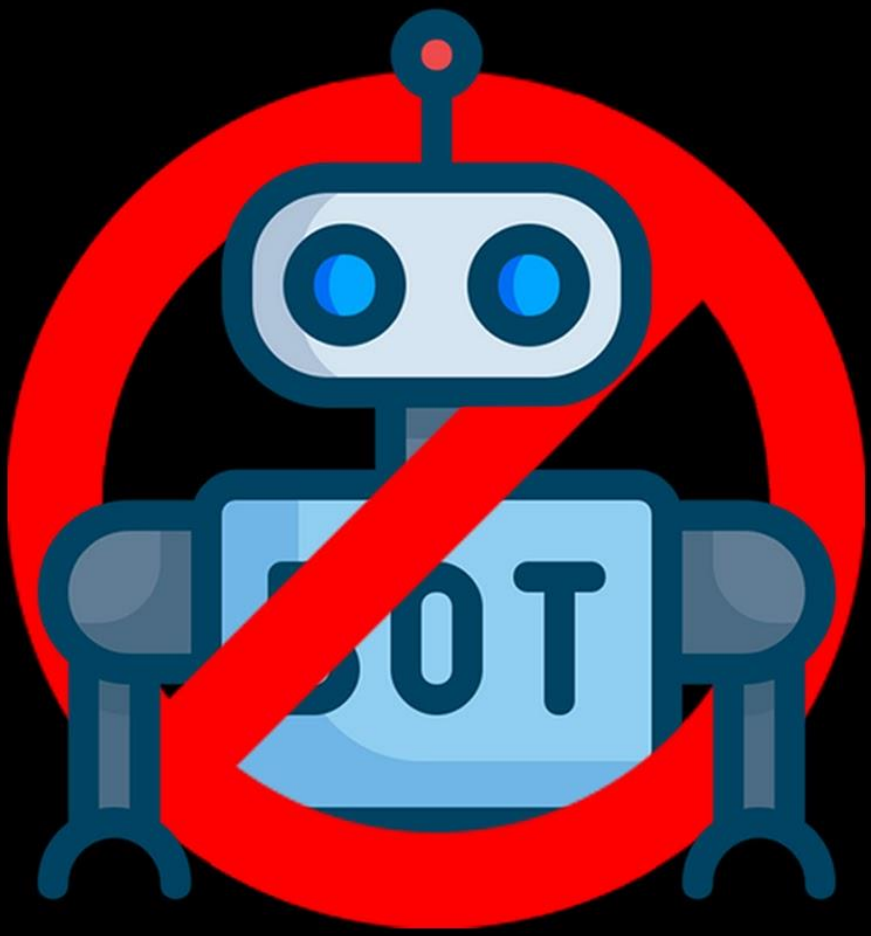
41004 AI/ANALYTICS CAPSTONE PROJECT - AUTUMN 2021

ASSIGNMENT 2: MID-PROJECT UPDATE & PRESENTATION



GROUP 35:

ASHLEY NGUYEN (13389465) DENZEL MOK (13229671)

MITCHELL VALENTINUS (13643352) PATRICK SONGCO (10420139)

BOT BUSTERS

PROJECT 24: EXPLAINABLE BOT ACCOUNTS DETECTION IN SOCIAL NETWORKS WITH GRAPH MINING TECHNIQUES

SUPERVISOR: DR. HONGXU CHEN

Table of Contents

Table of Figures

1. The Business Problem

The goal of this project is to analyse current progress in social bot detections, visualise it, and study the possible application of graph-based algorithms to the detection procedure, and finally, attempt to propose a novel methodology in explainable social bot detections.

For our research background, we studied why social media is important for cryptocurrency. We found that Reddit and Twitter are the most popular platforms for cryptocurrency. As of April 2021, the subreddit Bitcoin has 2.7 million members and Cryptocurrency has 2.1 million. On the other hand, Twitter, the platform that we chose for our research project, has 210K tweets daily with the #Bitcoin (BitInfoCharts 2021). Twitter also becomes the industry's newsfeed, where the important debates that define the industry happens, such as Bitcoin vs Gold, where the cryptocurrency reputation is made, and where the industry leaders are, like the top traders, journalists, researchers, and even Elon Musk. Twitter also recognizes that tweets about Bitcoin have been surging throughout the years simultaneously with the rise of Bitcoin price (Lielacher & Pickering 2020), the graph is presented in Figure 1.

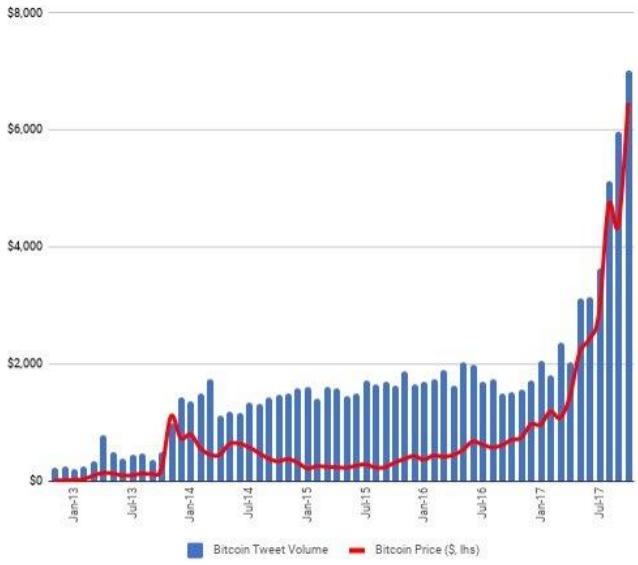


Figure 1: Bitcoin Tweet & Price Graph

As we know how important social medias are to cryptocurrency, bots can be a real threat to the ecosystem. Bots themselves are fake or inauthentic accounts that are made to provide misinformation, increasing fake engagements, and even making scams easier. In our research, we found that there are 2 types of bots, the regular one that only serves one purpose, such as retweeting, and the fake accounts that are able to mimic the actions of real humans in social media, they can have locations enabled, post random pictures, and even tweet just like humans. This makes it harder to detect than the regular bot accounts.

Bots have been around for a long time, in 2018. Facebook disclosed that they have 1.3 billion fake accounts and their monthly users, 5% of them are fake. For Twitter, they estimated that 40% of their users are not real. These high numbers surely are not healthy for the social ecosystem (Lielacher & Pickering 2020). We found some bitcoin examples that experienced sudden huge upticks on user interests, Satoshi Vision Bitcoin ($BSV) and Ripple Bitcoin ($XRP).

This happened through bots that spam contents about them over and over again to gain real human's interest (Lielacher & Pickering 2020).

To approach this problem, we extracted several interesting user features from our own scrapped Twitter API dataset such as ‘age of account’, ‘number of followers’, and many else, as we believe these kinds of features can help us navigate and detect what kind of pattern do bots usually have. We used the Gephi Twitter Streaming Importer Plugin to find the edges between users, and we also used the Botometer API as the bot labeller for each user. We believe these kinds of applications and programs can help our research thoroughly and provide a strong foundation for further research.

2. Data Exploration

2.2 Data Gathering Problem

Due to the relative novelty of graph-based approaches in detecting malicious bots, we found it difficult to find existing datasets that contained both user information and network information (edge connections). As such, we utilized publicly available APIs such as the Gephi Twitter Streaming Importer Plugin, Twitter API and Botometer API to create our own dataset for analysis. The APIs, data mining process, and collected dataset will be discussed further in the following sections.

2.3 Data Mining Process

The data mining process involved three steps. First, we needed to scrape both user and network information using the Gephi Twitter Streaming Importer Plugin. Next, we extracted more information for each user using the Twitter API. Finally, we used the Botometer API to determine the likelihood that a user was a bot or human.

2.3.1 Gephi Twitter Streaming Importer Plugin

The Twitter Steaming Importer Plugin on Gephi was used to scrape 5001 Twitter users with 14573 connections between them. These connections represent hashtags present in Tweets between users. The hashtags used were based on the cryptocurrency topic. These included: #crypto #cryptocurrency #blockchain #bitcoin #btc #ethereum #eth #dogecoin #coinbase.

The output of this step was two tables. A Nodes Table containing all the users and an Edges Table containing all connections between users.

2.3.2 Twitter API

The "get_user" endpoint was used on the Twitter API to extract more meaningful information for each user. This endpoint takes input in the form of a Twitter screen name and outputs a JSON file (Figure 1).


```
'{"id": 14254757, "id_str": "14254757", "name": "CK Nomad", "screen_name": "DrumKitt87", "location": "Atlanta, Ga", "profile_location": n
ull, "description": "Former professional drummer for the likes of Arrested Development turned tech entrepreneur, programmer, crypto enthu
siast, and world traveler! #BTC #ETH $TRAC", "url": null, "entities": {"description": {"urls": []}}, "protected": false, "followers_coun
t": 337, "friends_count": 652, "listed_count": 8, "created_at": "Sun Mar 30 01:40:05 +0000 2008", "favourites_count": 30943, "utc_offse
t": null, "time_zone": null, "geo_enabled": false, "verified": false, "statuses_count": 3154, "lang": null, "status": {"created_at": "Mon
Apr 12 18:28:28 +0000 2021", "id": 1381675602713792513, "id_str": "1381675602713792513", "text": "RT @DrevZiga: That moment when @branara
kic brings up the slide below at a meeting... @origin_trail Decentralized Knowledge Graph enabled da\\u2026", "truncated": false, "entiti
es": {"hashtags": [], "symbols": [], "user_m...'
```


Figure 2: Example user JSON file

Specific features in the JSON file were extracted for the purpose of this analysis (Table 1). These features were selected based on the Twitter policy on spam profile detection.

Extracted/Derived User Features	Assumptions
Age of Account (Years)	Bots generally have new accounts
Number of Followers	Bots have low number of followers
Number of Followings	Bots tend to follow a large number of users
Follower/Followings Ratio	This ratio is low for Bots
URL in User Description	Bots typically share links to malicious sites
Verified Account	Bot accounts aren't verified
Number of Tweets (including Retweets)	Bots post tweets more frequently
Number of Tweets User has Liked	Bots like tweets more frequently
Number of Public Lists User is a member of	Bots are members of many public lists

Figure 3: Selected Features Table

During the data collection process in this step, some users were either deleted or suspended by Twitter since the initial collection of users in step one (Figure 2). Therefore, these users were deleted in the data pre-processing phase (Section 2.4)

```
[{'code': 63, 'message': 'User has been suspended.'}]
Account @horanghaeey
100
[{'code': 63, 'message': 'User has been suspended.'}]
Account @btswifeuuuuuuuu
[{'code': 63, 'message': 'User has been suspended.'}]
Account @candymincyypal
200
[{'code': 50, 'message': 'User not found.'}]
Account @danielsdozie
300
[{'code': 63, 'message': 'User has been suspended.'}]
Account @walefinance
400
```

Figure 4: Sample error codes using Twitter API

2.3.3 Botometer API

The final step was to establish a ground truth label for each user. This was accomplished by thresholding the score given by the Botometer API. The Botometer API is a publicly available web application developed by Yang *et al*. (2019) that assigns a score between 0-5 of bot-like behaviour to Twitter users; 5 being a high likelihood that the account is a bot. As such, we arbitrarily chose to label all users with a score of 4 and above as a bot, human otherwise.

2.4 Data Pre-Processing

The raw Nodes Table and Edges Table were cleaned and pre-processed using Python in a Google Colab notebook.

2.4.1 Nodes Table

This table contains the list of all users and their features. The various alterations are outlined below.

- 77 users were deleted because additional features in step two were unable to be extracted.
- Attributes in Table 1 were added as columns for extra features.
- During the collection of Botometer scores in this step three, some users were assigned a blank score. For the purpose of this analysis, we have assumed that these users were either deleted or suspended by Twitter for being a bot/spam account. These 89 users were given a score of 6. We chose a score of 6 instead of 5 so that we can omit them from analysis if needed.
- Human/Bot Label column was added. This was calculated by thresholding the Botometer score. A score of 4 or over was labelled 1, 0 otherwise.
- Screen names for each user were converted to lower case for uniformity

2.4.2 Edges Table

This table contains the list of all users-user interactions and hashtag present. The various alterations are outlined below.

- 323 rows were deleted as they contained interactions between users that were deleted in the Nodes Table.
- Unwanted columns were removed.
- The @ and # symbols were removed from the beginning of screen names and hashtags respectively.
- Screen names for each user were converted to lower case for uniformity

2.5 Final Cleaned and Pre-Processed Dataset

The statistics of the final dataset we will be using for the modelling stage is shown in Table 2 below. Figure 3 and 4 display the first 5 rows of the Nodes and Edges Tables respectively.

	Raw Data	Pre-Processed Data
Number of Users (Nodes)	5001	**4924**
Number of Connections (Edges)	14573	**14250**
Number of Attributes	3	**12**

Figure 5: Dataset Statistics Table

	Screen Name	Id	Age in Years	# of Followers	# of Followings	Follower/Following Ratio	URL	Verified	Number of Tweets	# of liked Tweets	# of lists	Botometer	Bot: 3.5 Threshold	Bot: 4.0 Threshold
0	rumkitt87	14254757	13	321	649	0.49	0	0	3009	28401	6	0.7	0	0
1	ryptomichnl	146008010	11	181300	536	338.25	1	0	57312	43979	2367	0.9	0	0
2	atihsk87	2665227374	7	73248	409	179.09	0	0	11859	7887	1882	0.3	0	0
3	p889900	1065823520314140000	3	129	444	0.29	0	0	474	1438	0	1.6	0	0
4	ewardiqa	1200106100491720000	2	79551	90	883.90	1	0	45	98	14	1.6	0	0

Figure 6: Nodes Dataset Head

	Source	Target	hashtag
0	drumkitt87	cryptomichnl	ethereum
1	drumkitt87	fatihsk87	ethereum
2	cryptomichnl	fatihsk87	ethereum
3	sp889900	rewardiqa	binance
4	sp889900	rewardiqa	rew

Figure 7: Edges Dataset Head

3. Initial Findings

3.1 Correlation Matrix

The initial analytics results have provided important insights to the dataset, from which we were able to facilitate the following steps and make adjustments to the project plan accordingly. Each finding is presented with a visualisation and a corresponding observation.

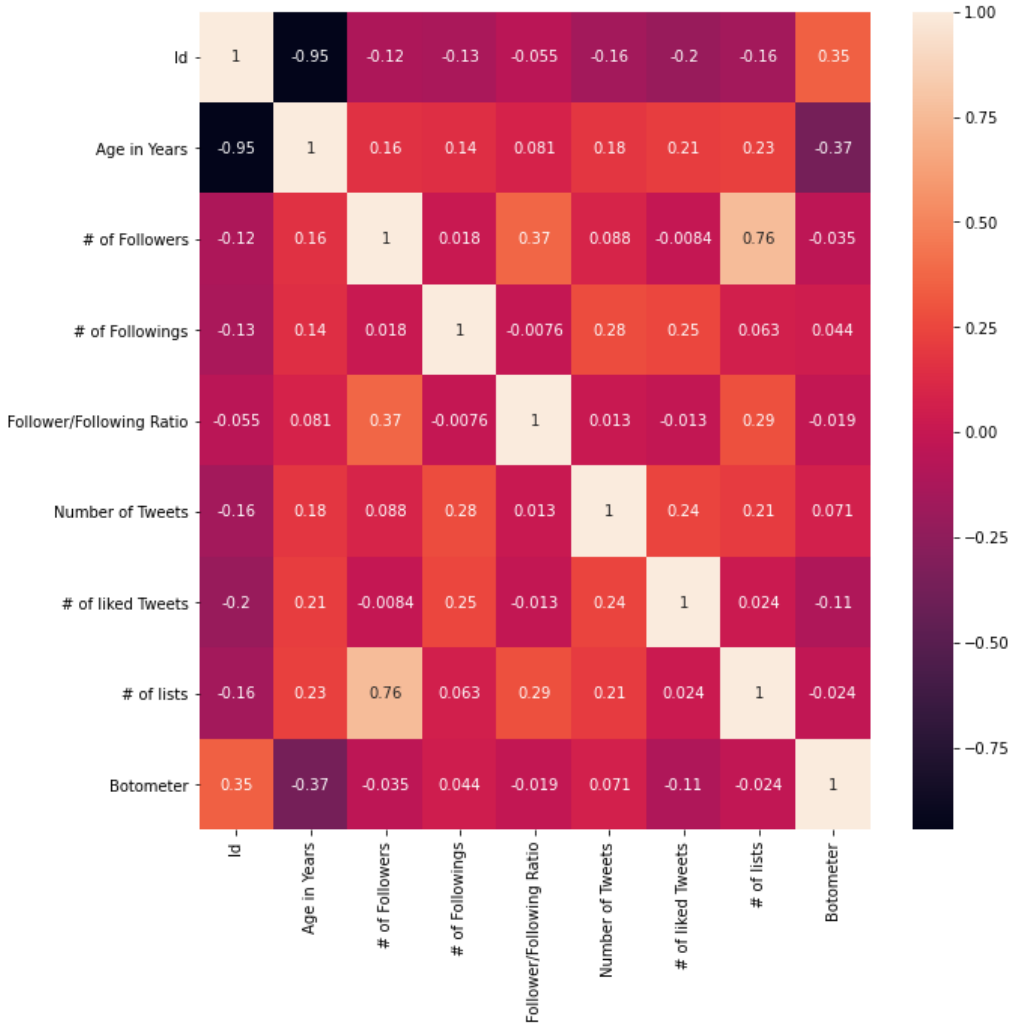


Figure 8: Data Features Correlation Matrix

The Correlation Matrix for our Twitter dataset is created using existing Python library. Each attribute of the data table is listed on every row and column. The value in a cell indicates the correlation between the horizontal row and the corresponding vertical row properties in the chart (note that cells on the diagonal line top left to bottom right are always 1.0 indicating a perfect correlation between an attribute and itself):

- The value is closer to 0, indicating the low correlation between the two properties.
- The value is closer to 1, representing two properties with a proportional relationship.
- The value is closer to -1, representing two properties with an inverse relationship.

From the correlation matrix results, these attributes seem to portray a low correlation to one another (approximately 0.0) as the information they convey is independent of other attributes. However, there are some visible correlation relationships, naming Age in Years versus ID and Number of Followers versus Number of Lists.

- ID and Age in Years share an inverse relationship as the newer the account, the higher the ID number is.
- The interesting relationship here is Number of Followers versus Number of List (scored 0.76 on the correlation matrix) as they share a proportional relationship.

Although some properties display a high positive or negative correlation value, we still need to consider their practical use, whether they really make sense in practice or statistical coincidence.

3.2. Age of Account Distribution

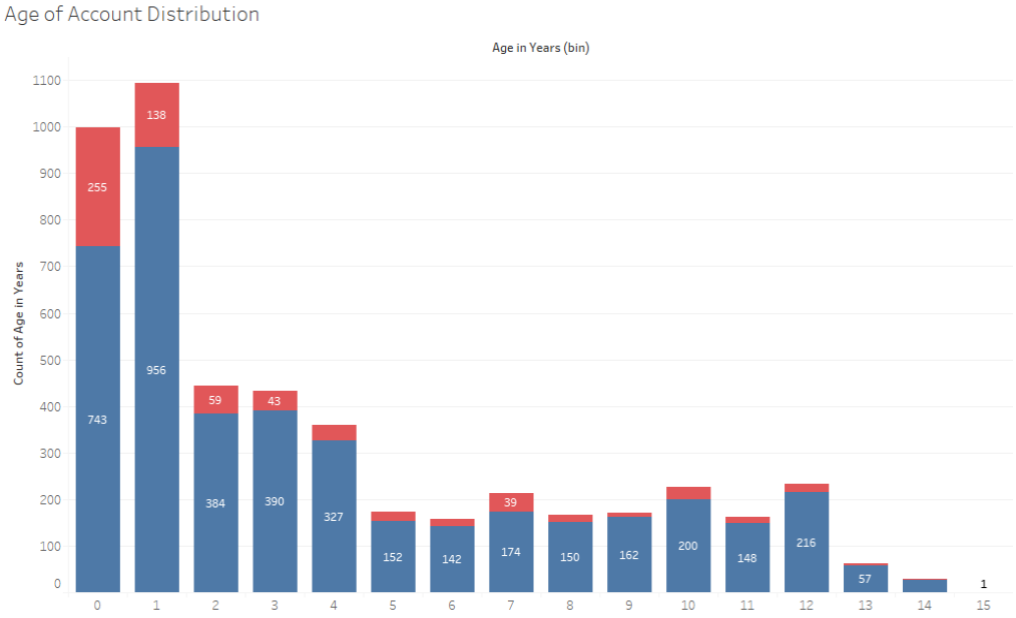


Figure 9: Age of Account Distribution

The Age of Account distribution reveals that more than 70% active bots tweeting about cryptocurrency are “fresh”, meaning that they have only been created around a year or less. We could assume that this might be partially due to the fact that discussions on cryptocurrency trading have only been around for the last 3 years.

3.3. Verification Check

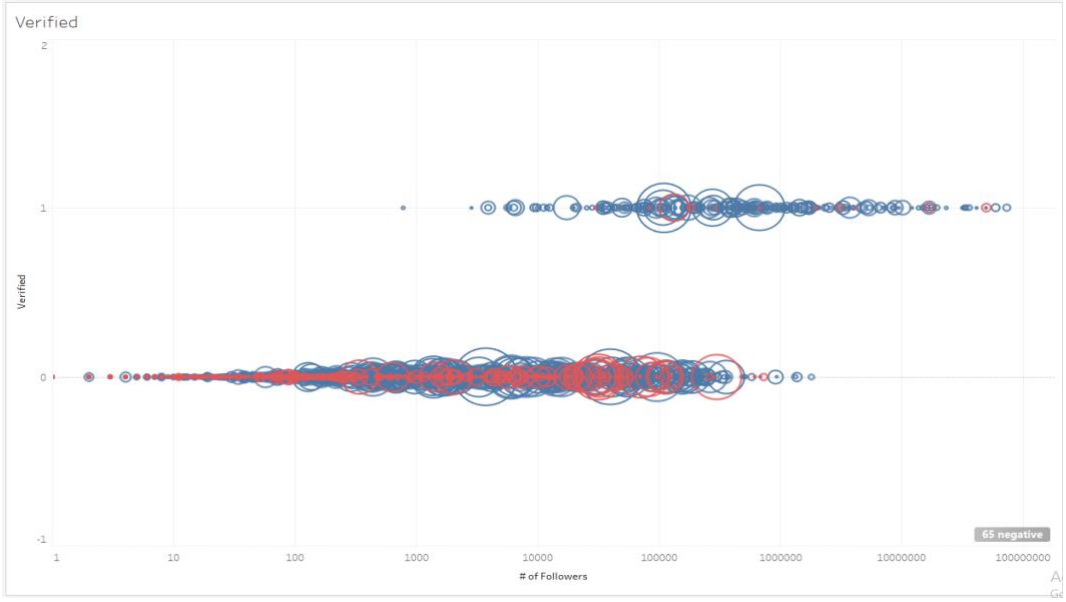


Figure 10: Verified Check versus Number of liked Tweets

Another observation we've noticed is that almost all bot accounts are not verified. When plotting it against the number of liked Tweets (indicating the users' interactive activities on Twitter), it is visible that bot accounts almost cannot be verified despite the effort put into making it interactive and human-like.

3.4. Follower/Following Ratio

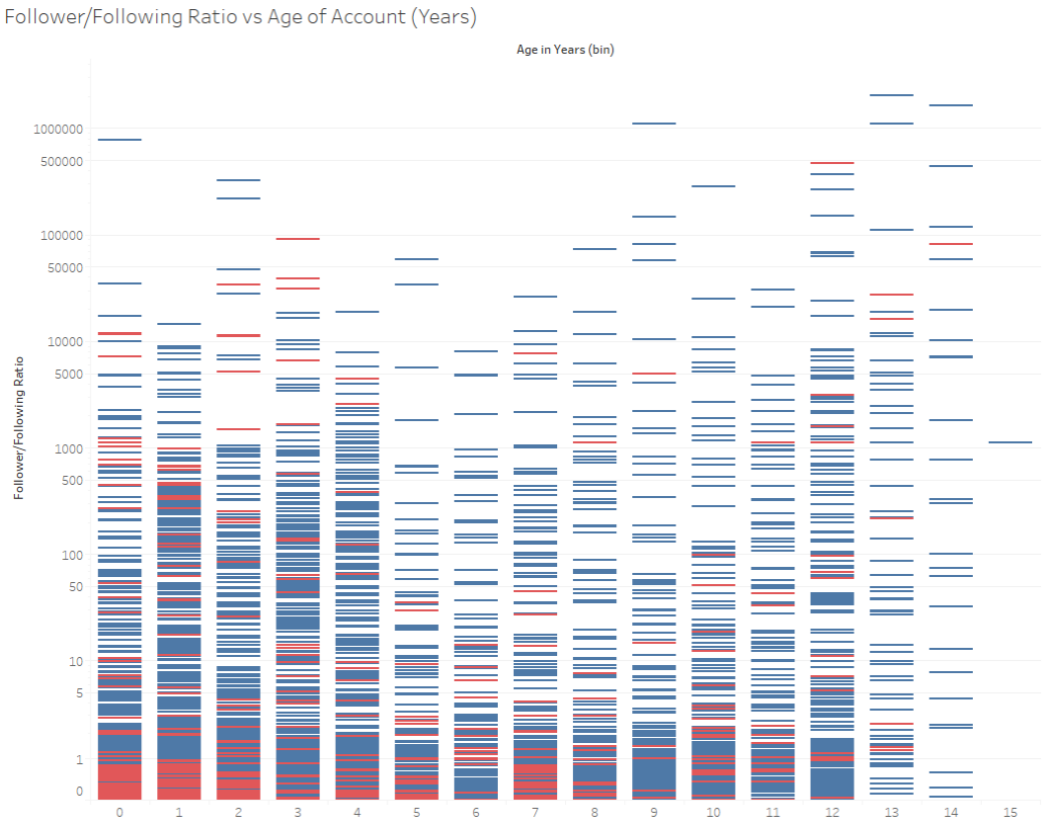


Figure 10: Follower/Following Ratio versus Age of Accounts (Years)

When plotting accounts' Follower/Following Ratio throughout the Age of the Accounts (in Years), we noticed that there were a large number of red-coloured accounts (bots) portrayed in the bottom left corner of the graph. This contradicts with Chu et al (2012) and previous studies that a high follower/following ratio could indicate a spamming behaviour. In fact, recent bots tend to have a lower follower/following ratio (closer to 1). As Twitter have imposed a limit on the follower/following ratio for bot concerns, we could explain this as an attempt to imitate human accounts.

3.5. Number of Liked Tweets versus Follower/Following Counts

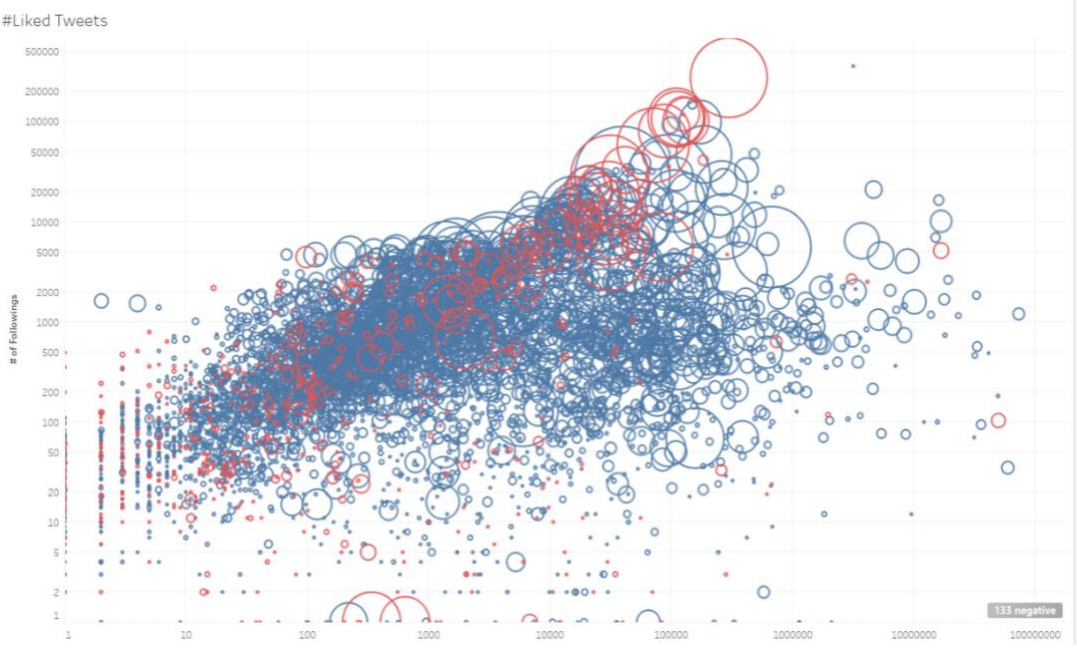


Figure 11: Number of liked Tweets versus Follower/Following counts

Similarly, bot accounts with higher number of both followers and followings (low follower/following ratio) tend to have more Tweets liked as well. This seems to be an attempt to imitate a high interaction on Twitter to avoid being "bot-busted".

3.6. Outlier Detection

Outlier detection was performed on some attributes (Figure 12). We find that many users, both human and bot users fall outside of the 1.5 Interquartile range and therefore can be deemed as outliers in their respective distributions. These users will be taken into account when performing our analysis and modelling.

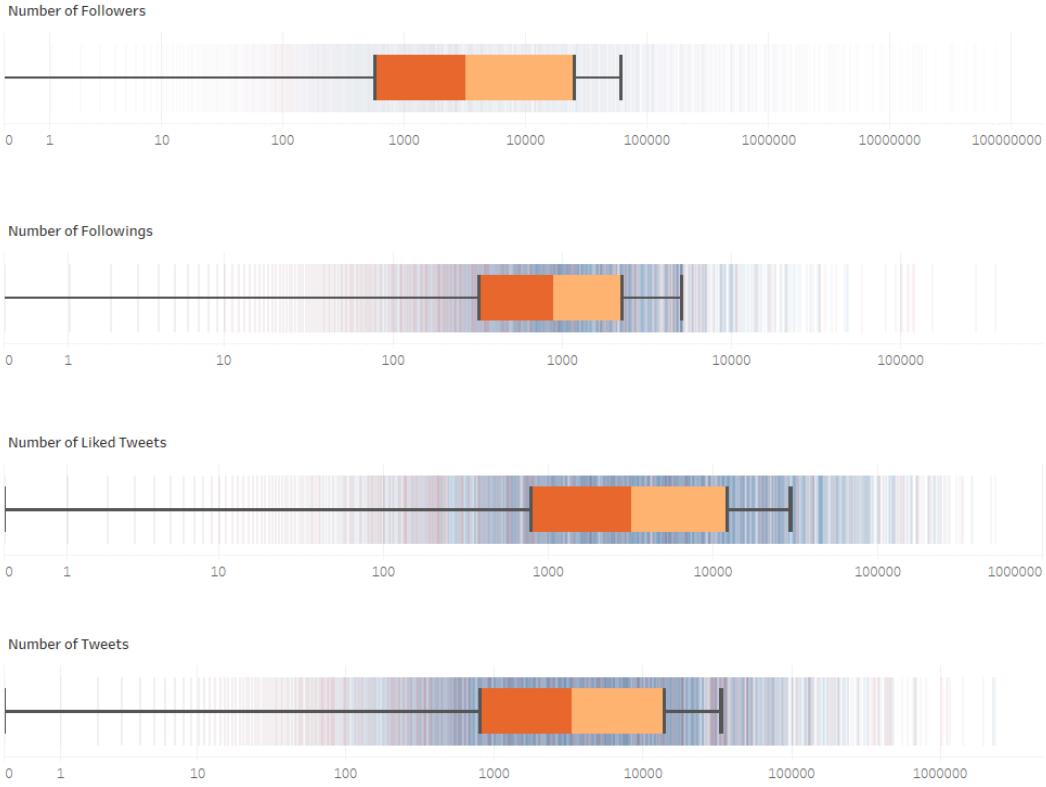


Figure 12: Boxplot Analysis of chosen attributes

3.7. Cluster Detection

Cluster detection was performed on Gephi to help visualise the communities present in the network (Figure 13). The modularity of the dataset was calculated using a community detection algorithm. It can be seen that multiple communities/clusters are present in the dataset and verify the use of graph-based algorithms for analysis. This is due to the strong network structure made available by connecting users via hashtags present in Tweets. Additionally, an average clustering coefficient of 0.482 was calculated meaning that several of the users in the network are connected to many other users in the network. Again, displaying the strength of analysing the users as a network as opposed to individually.

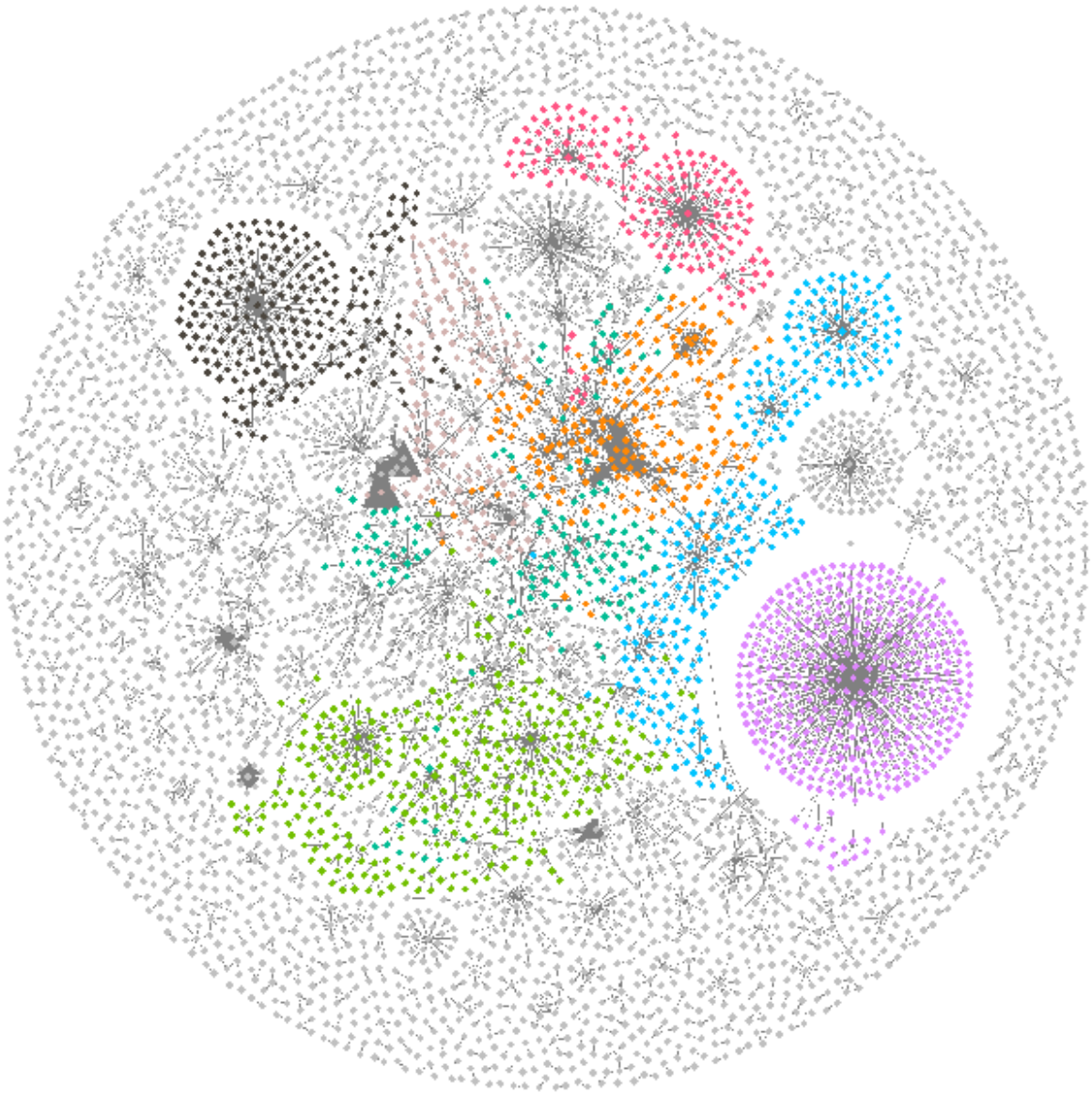


Figure 13: Clustering Visualisation of Dataset

3.8. Network Visualisation

Our last finding is the inherent network structure between clusters (Figure 14). This gives us an intuition that clusters of users can provide a graph-based model extra features to classify whether a user is a human or a bot.

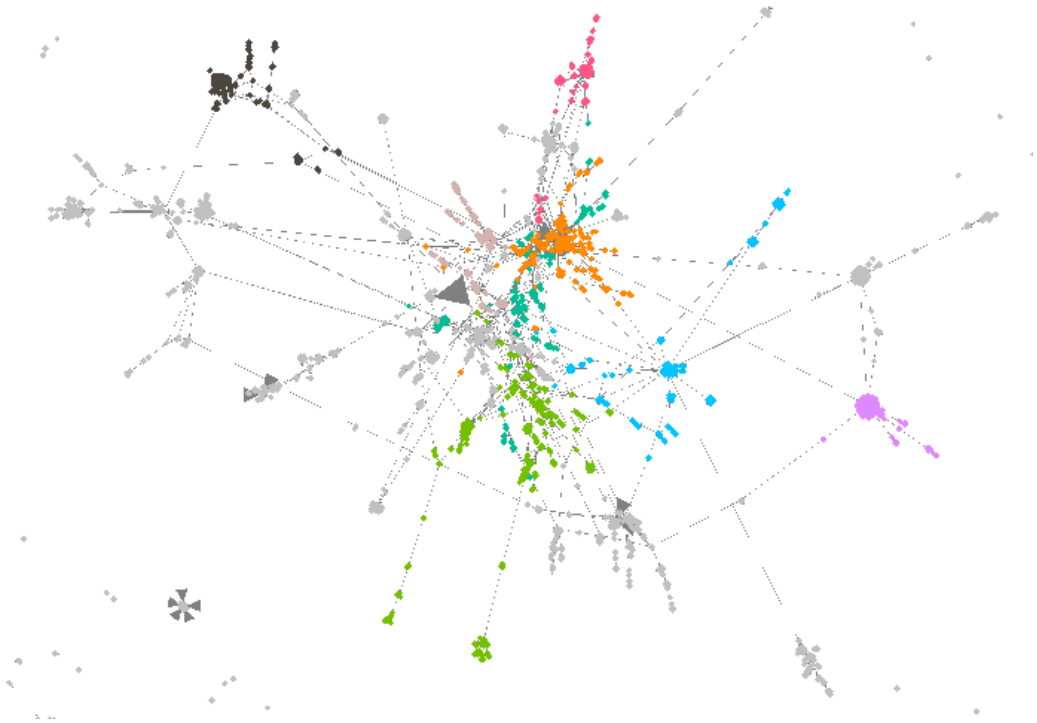


Figure 14: Network Visualisation of Dataset

4. Difficulties Encountered

Over the course of the project so far, we have encountered many issues that resulted in unexpected delays and changes to our project plans. In this section, we will identify some of the major difficulties experienced in our project, as well as the risks we have identified that could negatively impact the project.

4.1 Difficulties

The problems we have encountered have been listed below with the corresponding solution or response we have developed to overcome this issue:

- Unable to find an appropriate dataset that is relevant to our business problem
    - We generated our own custom dataset to get attributes and relationships that were of interest and highly relevant to what we wanted to find

- Our data collection generated an imbalanced dataset with more human labels than bot labels, creating complications in the modelling process
    - We plan to use sampling techniques or cross validation to handle the imbalanced data to minimize the effects of an imbalanced dataset

- Long and complicated process to apply for developer access to twitter's API which was used to obtain attributes for our dataset
    - We started and planned for the application process well in-advance of deliverables dates and had several members apply for developer access to increase our chances of obtaining approval

- Selecting appropriate Twitter tags to collect user account data that captured and encompassed a good representation of our problem
    - We based our choice on trending hashtags using publicly available websites that provided information on popular tags

- Rate limit on the number of Botometer scores that can be obtained per day using a publicly available Bot scoring API
    - To get around this issue, we used multiple accounts across several days to speed up the collection, however if we wanted any new data or make changes to our collection methods, additional time is required which may result in delays to our progress

- Twitter user accounts were deleted after our initial data collection process, resulting in problems with obtaining additional attributes and missing scores with the Bot scoring API
    - We cleaned and pre-processed the data to remove any user accounts that were removed from Twitter
    - For the Bot scoring API, we assumed that the accounts which did not return a score was removed due to their classification as a bot account, and therefore presented them with a bot label

4.2 Risk and Mitigation

During our project, we identified the risks to our project as well as the mitigation strategies to ensure minimal impact and delays. The risks have been provided below, each accompanied by their mitigation strategy:

- Missing project deadlines because of poor scheduling
    - Split project into smaller tasks among team members to balance the workload
    - Weekly meetings among team members to discuss project requirements and progress
    - Use of a Gantt chart to help plan and manage our progress

- Misunderstanding the requirements and objectives of the project
    - Frequent meetings with the project supervisor to ensure that the project is on the right track
    - Clear communication and assistance among team members through different contact channels such as email or a discussion group

- Project delays from unexpected events (illness, injury, etc.)
    - Prepare backup or contingency plans in case anything goes wrong
    - Add extra time or a safety cushion to deadlines to make sure that we have enough time to complete our project if an unexpected event occurs

- Legal and privacy concerns that may implicate our project
    - Research and acquire any necessary approvals to comply with the law
    - Hide and remove any sensitive or identifying information from our analysis and findings

5. Project Plan

5.1 Updated Project Scope

The Business Understanding, Data Understanding and Data Preparation phases of the CRISM-DM methodology have been completed. Three of the four major deliverables have been completed and we anticipate that the project will be completed on time.

With guidance from our supervisor, we have added scope to the Modelling phase. The added task is to implement traditional machine learning methods on the created dataset to verify the feasibility of the dataset in detecting malicious bots. This will not affect the overall timeline as we have assigned different team members to implement different models in parallel. The additional models that we will be implementing are Logistic Regression and Random Forest models.

The final five weeks of the project the team will focus on:

1. Generating test designs for both bot detection and explainability
2. Training/Testing:
    I. Logistic Regression
    II. Random Forest
    III. GNN/GCN
3. Results Evaluation
4. Final Reporting

5.3 Updated Gantt Chart

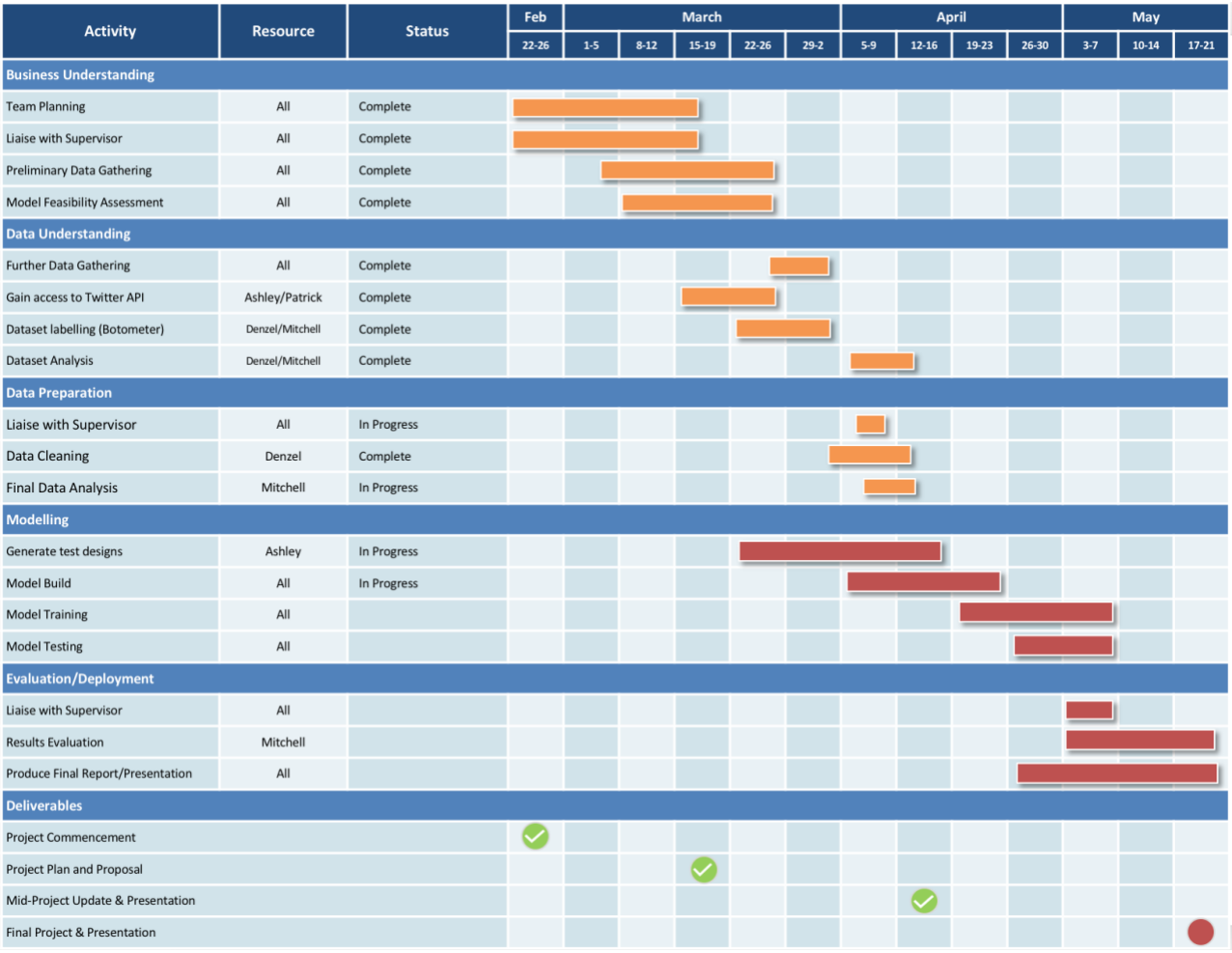


Figure 15: Updated Project plan Gantt chart

6. References

BitInfoCharts (2021). *Bitcoin Tweets historical charts*. https://bitinfocharts.com/comparison/bitcoin-tweets.html

Lielacher, A., and Pickering, A. (2020). *Fake views: How social media bots distort the crypto narrative*. Brave New Coin. https://bravenewcoin.com/insights/fake-views-how-social-media-bots-are-distorting-the-crypto-narrative

K.-C.Yang, O.Varol, C.A.Davis, E.Ferrara, A.Flammini and F.Menczer (2019) "Arming the public with artificial intelligence to counter social bots" Human Behaviour and Emerging Technologies.

Zi Chu, Gianvecchio, S., Haining Wang, & Jajodia, S. (2012). Detecting Automation of Twitter Accounts: Are You a Human, Bot, or Cyborg? *IEEE Transactions on Dependable and Secure Computing*, *9*(6), 811–824. https://doi.org/10.1109/TDSC.2012.75