JOURNAL OF CYBERSECURITY

Journal of Cybersecurity, 2023, 1–13
https://doi.org/10.1093/cybsec/tyad014
Research paper

Research paper

Digital fingerprinting for identifying malicious collusive groups on Twitter

Ruth Ikwu[1,*], Luca Giommoni[1,*], Amir Javed[2], Pete Burnap[2] and Matthew Williams[1]

[1]School of Social Sciences, Cardiff University, Cardiff CF10 3WT, UK and [2]School of Computer Science and Informatics, Cardiff University, CF24 4AG, UK

*Corresponding author. Cardiff School of Social Sciences, Cardiff University King Edward VII AVE CF10 3WT, Wales, United Kingdom; E-mail: ruth.ikwu@outlook.com

Received 31 October 2022; revised 1 May 2023; accepted 20 June 2023

Abstract

Propagation of malicious code on online social networks (OSNs) is often a coordinated effort by collusive groups of malicious actors hiding behind multiple online identities (or digital personas). Increased interaction in OSN has made them reliable for the efficient orchestration of cyberattacks such as phishing click bait and drive-by downloads. URL shortening enables obfuscation of such links to malicious websites and massive interaction with such embedded malicious links in OSN guarantees maximum reach. These malicious links lure users to malicious endpoints where attackers can exploit system vulnerabilities. Identifying the organized groups colluding to spread malware is non-trivial owing to the fluidity and anonymity of criminal digital personas on OSN. This paper proposes a methodology for identifying such organized groups of criminal actors working together to spread malicious links on OSN. Our approach focuses on understanding malicious users as 'digital criminal personas' and characteristics of their online existence. We first identify those users engaged in propagating malicious links on OSN platforms, and further develop a methodology to create a digital fingerprint for each malicious OSN account/digital persona. We create similarity clusters of malicious actors based on these unique digital fingerprints to establish 'collusive' behaviour. We evaluate the ability of a cluster-based approach on OSN digital fingerprinting to identify collusive behaviour in OSN by estimating within-cluster similarity measures and testing it on a ground-truth dataset of five known colluding groups on Twitter. Our results show that our digital fingerprints can identify 90% of cyber personas engaged in collusive behaviour and 75% of collusion in a given sample set.

Key words: malware propagation, cyber persona fingerprinting, online social networks

Introduction

Organized crime groups, including hackers, misinformation campaigners, and even actors of illicit trafficking rings, have used Twitter to spread malicious information and resources [1]. Interactions on online social networks (OSNs) can now eliminate trade-offs between anonymity and efficiency in distributing malicious codes online. This luxury afforded to malicious actors makes it difficult for analysts to pinpoint perpetrators of various cybercrimes on OSN platforms. Interaction on most social network platforms means a user can follow other users, like, mention, share, comment on, and click on posts by other users. Microblogging sites such as Twitter are therefore efficient for hackers to spread malicious codes with higher chances of users clicking on these malicious links [2].

Digital personas (represented by online accounts) play a critical role in how criminals leverage OSNs to maximize the effects of their malicious efforts. Malicious actors can operate multiple accounts simultaneously, i.e. there can be a single account operated by multiple physical persons or one physical person operating multiple online

accounts. As there are no constraints on the number of digital personas attributed to one person, one criminal may control multiple cyber personas, or a single cyber persona may be shared by a criminal group actively engaging with potential victims [3]. This anonymity also makes it easier for malicious actors to build online numerous online fake accounts to automate criminal activities [4]. For example, research demonstrates how one malicious user can create armies of sockpuppets or 'ego-networks' to boost deceptive behaviour in online discussion communities [5]. These sockpuppets deceive ordinary users with personalized messages into engaging and spreading their malicious content. Similarly, criminal entities can also disguise themselves as legitimate accounts masquerading as 'trusted sources' while spreading malicious content [6].

Before the popularity of OSNs, it was harder to get malware to unsuspecting victims as hackers relied on traditional social networks such as spam/phishing emails [7] or comment sections of newsgroups [8]. These methods were limited due to minimum reach while demanding a lot of effort to get users to download malicious codes onto their systems. The distinguishing feature of OSNs is the ability of users to freely share and interact with other users' content. This feature creates an efficient self-propagating platform for malware distribution that requires minimum effort but guarantees maximum reach. Before the explosion of OSNs, researchers had identified the importance and nature of criminally motivated networks and how information is propagated within them [9]. In 1993, e.g. the reconstruction of malicious social communication networks involved in price-fixing the electrical equipment industry concludes that criminal networks are set up to maximize concealment rather than efficiency [9]. However, in the case of modern social networks, Sanzgiri, Hughes, and Upadhyaya debunks the myth of the need for digital secrecy but rather emphasizes the goal of malicious actors as physical anonymity [6].

In OSNs, the distribution of malicious links is largely enabled by URL shortening [10]. Perpetrators can distribute links to malicious sites or software by embedding them in shared posts. However, microblogging sites like Twitter have limitations on the number of characters that can fit in a single post; therefore, users use short URLs in posts rather than standard URLs. Obfuscated or short URLs are standard URLs that have been encoded into URLs with fewer characters primarily to fit into the character limits of social media posts using URL encoding services such as bit.ly and TinyURL [11], and Twitter's shortening service. URL encoding is a form of obfuscating information that tricks unsuspecting users to download/spread malicious software without their knowledge. Malicious actors working together will either propagate the same obfuscated URL or use the same obfuscation tool/service to encode their malicious links. Identifying these criminal actors and groups in OSNs is challenging as they are often hidden underneath layers of information and are usually portrayed as 'legitimate' accounts. Thus, in this study, we present a novel approach for identifying collusive groups of malicious actors working together to spread malware. Our approach exploits the digital fingerprints of a user's online presence using an unsupervised learning approach to group similar online digital fingerprints. To the best of our knowledge, this is the first study that fingerprints each malicious user and identifies collusion between malicious users.

We aim to build a generalizable framework that can effectively identify groups of social media accounts working together to propagate malicious links. We use Twitter as the OSN, although our results can be applied to microblogging sites with similar social structures to something like Facebook. Our novel methodology provides the following inherent contributions to the body of work: For known malicious accounts, we define and create a digital fingerprint for each associated digital persona based on the following criteria:

- **A URL fingerprint:** to identify accounts that use similar tools in creating their attack vector.
- **An account fingerprint:** to identify accounts created with similar account characteristics.
- **A post-content/language fingerprint:** to identify accounts with the same set of people writing posted content or posting similar content.
- **An activity fingerprint:** to identify the account with similar interaction patterns between them.

We apply an unsupervised machine learning algorithm to these digital fingerprints to discover latent similarities between digital personas. Our unsupervised learning creates groups of colluding actors based on similarities in their digital personas. To evaluate our digital fingerprint, we estimate within-cluster similarity measures of users in a cluster. Furthermore, using five groups of known colluding accounts as a baseline, we compare the extent to which the application of an unsupervised learning model on our digital fingerprints can identify collusion amongst these accounts.

The rest of the paper is organized as follows: The 'Methodology' section explains each step of the methodology namely, how we developed a fingerprint for digital personas and how we developed an unsupervised learning model to group these digital personas. The 'Creating digital fingerprints for malicious actors' subsection of the 'Methodology' section particularly explains the process of establishing collusion, which is critical as evidence of organized criminal activities. This subsection explains the similarity of social interaction parameters as evidence of collusion. Finally, we evaluate and compare our results in the 'Results' section with current literature and conclude in the 'Conclusion' section.

Background

In the past, traditional malware propagation techniques such as spam emails and comment sections of newsgroups were used to entice users to download and share various malicious codes. The advent of 'flexible' digital personas in highly interactive modern social networks such as Twitter, and the use of advanced URL obfuscation techniques, has enabled hackers to automate the dissemination of malicious links.

When a person creates an online account on a digital platform, they create a digital representation of themselves called a cyber or digital persona. Prominent studies in the field provide the rationale for a digital persona as a model for representing an individual's identity in cyberspace [12]. These digital personas are developed through collecting, storing, and analysing an individual's digital footprint (or online activities) [13]. Other researchers further demonstrate how the activities of online accounts leave digital footprints that are unique to a person's existence in cyberspace, i.e. a digital fingerprint [3]. Social interconnectedness establishes support for creating online communities of digital personas characterized by active social presence, social participation, and collaboration.

Digital personas are attributed directly to an actual person or persons with certain cyber characteristics, e.g. email address, IP address, social accounts, etc. The components of the digital persona define online identities and characteristics of people as they interact within an online network [14].

For profiling digital personas in underground networks, researchers have developed simple solutions to tracking deviance in such networks. Some researchers show how to create digital profiles for physical actors in major underground forums [15]. Their methodology exploits simple Skype communication protocols to expose location data, network behaviour, and work habits from online Skype handles of malicious underground actors. Others cluster

user accounts from various underground forums to predict if a user would engage in high-profile criminal behaviour in the future [16]. Flores, Garcia, and Cortez use structural profiles of Twitter network users to create network embeddings and establish similar groups [17]. However, for modern social media networks such as Facebook, Twitter, and Instagram, there are a greater proportion of ordinary, goodwilled users not involved in malicious activities. The challenge, therefore, is identifying malicious intent in such massive social networks.

Such social network platforms offer key features that allow users to share content and interact with posted content. Twitter is one of many social media platforms that allows the broadcasting of 280 characters limited 'tweets'. A person creates a Twitter account, thereby creating a digital representation of themselves on the platform. A registered user creates a 'tweet' (a 280-character limited post) and shares it on his/her 'timeline'. A timeline is a historical feed of a user's activity on Twitter. A tweet can contain any one of the following types of content: text, videos, images, or links to internal or external content. Interaction on Twitter is supported by connections between multiple users. A connection is created between two Twitter users when a user 'follows' another user or a user is 'followed by another user'. This user–follower entity model is a representation of the structure of information dissemination in OSN platforms. In addition to direct connections, interacting relationships are also crucial to the structure of information dissemination on the platform. Two users A and B have an interacting relationship each time:

- User A retweets a post from User B or User B retweets a post from User A.
- User A replies to a tweet by User B or User B replies to a tweet by User A.
- User A likes a post from User B or User B likes a post from User A.
- While using the '@username', User A mentions User B in a tweet or User B mentions User A in a tweet.

It is important to note that relationships on Twitter can be asymmetrical, i.e. follower–following relationships, but are not automatically so. A collection of tweets can represent information on a specific topic. Users can indicate topic categorization tweets by appending '#' to the topic, e.g. #crypto. These are called hashtags and are used to decide trending topics of interest on the platform. Hackers often leverage this OSN feature to reach many users by appending a 'trending' hashtag to the post with content containing their malicious links. Within this ecosystem, malicious actors can form groups and work together to achieve malicious intent.

Mapping online collusive networks is important for understanding the operational structure or 'modus operandi' in social networks. Multiple studies claim that the key to unravelling collusive networks is understanding various measures of influence and identifying key actors in the network based on 'centrality' measures [18]. However, some research suggests that online collusive networks often face a trade-off between secrecy and efficiency as such, and measures of 'popularity' in overt social networks may not apply to identifying key actors in a covert criminal network incorporating secrecy [19].

Some researchers allege that 'secrecy/covertness' in malware propagation networks has a different meaning due to the nature of operation of covert networks on online social platforms [6]. Covert groups spreading malicious information on social networks are 'hidden' in their ability to disconnect their physical identities from their online identities. As opposed to real-world covert networks that use online social platforms as an enabler to recruit and share resources among already indoctrinated members, malware propagation networks intend to optimize information dissemination to a maximum number of unsuspecting victims. This assertion, therefore, emphasizes identifying 'key player accounts' (collaborating accounts) with influence and the means to spread information quickly to the most amount of people [6].

In addition to staying physically hidden while spreading malicious codes, collusive groups must have similarities in their digital personas. Although there is sufficient research covering the detection of automatically generated content on Twitter, a solution to establishing collusion between a set of accounts is still limited [20]. Therefore, it is possible to detect Twitter accounts generating and spreading automated content, but difficult to identify which of these accounts are collaborating to spread the content.

After extensive research, we have summarized the task of establishing 'collusion' in OSNs into identifying two phases: (a) synchronization in account activity [21] and (b) identifying similarities in digital personas [22]. Synchronized behaviour among independent random users is highly unlikely and therefore such suspicious activities are considered anomalous behaviour [23]. Consequently, current methods of detecting cooperating or colluding accounts on Twitter depend on supervised training of features observed over a certain period. The goal is to establish similarities between account activity features such as frequency of posting new content, consistent time of posting [23], and frequency of retweets [24]. These methods rely heavily on a long duration of time-based activities; however, true collusion is truly characterized by a short period of similar activities between accounts [23].

To address this gap, some researchers propose using overall interactions of potentially spam or automated accounts on Twitter as a stepping stone to identifying colluding accounts [21]. Unfortunately, these methods assume that collusion is only possible with automatically generated content. It is important to note that collusion is not restricted to 'fake' or 'spam' accounts and real-world users can actively participate in spreading malicious content online, e.g. in cases of cyber hacktivism [3].

Detecting collusion involves observing a consistent pattern of similarities in the activities of Twitter users, e.g. synchronized retweet activities between a group of accounts [20]. For example, by identifying similarities in account features such as interaction activities and retweet similarity, researchers demonstrate how to detect black-market-driven collusion [25]. One solution proposes a simple cross-correlation of activities as an absolute indicator of collusion between accounts. Simple cross-correlation of accounts' activities conveniently is not restricted to the presence of time-dependent activities between tweets, rather analyses account information relative to each other rather than independently [23]. Cross-correlating accounts assume that the activities and characteristics of collusive accounts are somewhat similar, and by quantifying the change and frequency of activities, it is possible to observe correlated user accounts acting similarly. Similarly, in OSNs, interaction mechanisms such as 'like' and 'retweets' have become social currency to boost the reach of malicious posts from malicious actors [2]. Sometimes, collusion takes the form of false boosting of malicious content with no future engagement. Unlike bot-based malicious ecosystems, these collusive networks are made up of real human users. For example, one study creates a set of heuristics for detecting such human-like false boosters of online content based on account activities and account follower–following networks [26]. Their research currently records the highest accuracy for a classification-based approach in the research body of work for detecting collusive actors on Twitter. However, meth-

ods used for identifying malicious actors in their research are less grounded than those provided in our work.

Methodology

Data collection

We collected data for 1 week on one popular event, the COVID-19 2020 pandemic. The COVID-19 related tweets were collected between 11 March 2020 and 21 March 2020. For this study, we collect tweets containing hashtags and phrases related to the COVID pandemic using Twitter's Streaming API. We used the hashtags such as #covid, #covid19, #lockdown, #StayHomeStaySafe, #StayHome, #QuarantineandChill, #corona, #coronavirus, and #pandemic to filter tweets from Twitter's Streaming API. Political, cultural, and social events provide increased interaction on social media platforms therefore increasing the probability of a malicious link being retweeted or shared. We selected a short-time frame to ensure we capture the immediate activities of any colluding actors within a specific period. For activity data, we performed data collection everyday of the data collection period to capture all user activity within the data collection period. We finally filter these to unique tweets excluding all retweets. We capture 1 255 178 COVID-19 related tweets collected in March at the beginning of the implementation of national lockdown rules. We filter tweets to only those with embedded external URLs. Finally, since Twitter automatically shortens all embedded links with Twitter's URL shortening service, we perform an initial GET request for each URL in our dataset. Using the python's 'redirect' parameter from it's GET request library [27], we captured the redirected long-form destination URL.

Labelling tweets

Each tweet was labelled as Benign or Malicious by parsing the long-form embedded URL through VirusTotal. VirusTotal is a free cloud-based antivirus engine aggregator that provides an integrated interface to several antivirus scan engines [28]. In addition to AntiVirus Engines, VirusTotal uses website scanners to detect malicious content available in URLs. VirusTotal performs antivirus scans in parallel across multiple antivirus engines. Each URL passed through VirusTotal is passed through at least 80 different antivirus search engines.

A tweet is labelled as 'Malicious' if at least one antivirus engine classifies its embedded URL as 'malicious' or 'suspicious'. A tweet is labelled as 'Benign' if no antivirus engine classifies its embedded URL as 'malicious' or 'suspicious'. Tweets with ambiguous results such as 'Unknown' or 'Unseen' (where virus total is unable to establish the label of embedded URL) are excluded from the results. Finally, we filter our data to tweets with embedded URLs tagged as 'Malicious' and tweets from all other users directly connected to malicious tweets. We detected 153 unique user accounts actively involved in spreading malicious links at the first week of the COVID-19 2020 pandemic. Of these malicious 153 account, 44 accounts were invalid due to account no longer existing at the time of analysis or invalid data fields from Twitter accounts, leaving 109 unique malicious users in our sample.

Creating digital fingerprints for malicious actors

A digital fingerprint aims to identify patterns representative of a person's online presence and its interaction with other digital personas. Various techniques to correctly characterize the persona combine multiple aspects of its existence in cyberspace such as how the speak (or write), identity characteristics, and online activity. We extend this method to create characteristics of malicious accounts that represent their existence in OSNs. We assume that each malicious account is controlled by at least one physical person. One cyber persona might be controlled by multiple physical people, or many cyber personas may be controlled by one physical person. Our aim is to create a representative set of digital persona features that can expose latent similarities between digital personas to identify groups of accounts that may be working together to propagate malicious codes. We use a set of 75 features, as shown in Tables 1–4 below, to characterize the different ways in which digital personas can exhibit latent similarities online based on literature. We create four sets of features to characterize a malicious actor's digital footprint:

- ***URL fingerprint:*** This examines in-depth characteristics of a malicious URL to identify actors who use similar propagation tools. For example, using the same obfuscation method, using the same host IP with varying paths and parameters in the URL.
- ***Account fingerprint:*** This examines the characteristics of a persona's online existence to identify accounts with similar markers as possibly created by the same set of people. For example, difference in account creation time, account location, account age, etc.
- ***Language fingerprint:*** This analyses similarities in language of posted content to identify accounts who have similar authors or are posting similar content. For example, accounts automating posting with similar post templates.
- ***Activity fingerprint:*** This examines similarities in the pattern of activity among users. This helps identify accounts who act in a similar way, e.g. bots programmed to post specific amount of content daily or programmed to like and retweet posts from other specific accounts.

Explanations for these features are detailed in the 'Attributing actors by URL characteristics DF_1 (characterizing or identifying similar propagation tools)' subsection through 'Attributing actors by activity characteristics (DF_4) (characterizing or identifying accounts where actors engage with content in a similar way)'.

Attributing actors by URL characteristics DF_1 (characterizing or identifying similar propagation tools)

The first category of our digital fingerprint are URL-based features, which focuses on identifying groups of cyber personas who use similar propagation tools. In attempting to identify covert groups of malicious actors on social networks working together, it is reasonable to assume that they use similar tools to create and distribute their malicious codes [29]. Although there is a wide acceptance of open-source software in these communities, hackers who purchase a Top-Level Domain for malicious intent are known to use the same TLD with multiple sub-domains and path strings to create new malicious links [30].

Furthermore, a common research objective in cyber threat mitigation is differentiating malicious URLs (URLs pointing to websites with malicious codes) from benign URLs (URLs pointing to clean websites). Some of these techniques for separating malicious and benign URLs provides solutions for analysing the metadata of URLs [30]. Other techniques identifies a set of lexical and host-based features that can be extracted from URL strings to separate malicious and benign URLs [31]. Lexical features refer to properties of the URL name such as the length of URL string or the number of digits present in URL string while host-based features relate to metadata of the host such as WHOIS information, IP Address properties, location, domain registration, and connection speed [30]. Table 1 below outlines features extracted from embedded URL strings.

Table 1: URL fingerprint features

SN	Criteria	Description	Source
1	URL length	Total number of characters in URL string	[30, 31]
2	URL digit count	Total number of numeric digits [0–9] in URL string	[31]
3	URL schema	Server access protocol	[31]
4	URL hostname	Hostname from URL query string	[31]
5	URL path	Section of webpage	[31]
6	URL query	Query string parameters	[31]
7	URL parameters		[31]
8	URL age	Number of days since URL domain was created	[32]
9	Number of subdomains	Number of sub domains associated with URL	[32]
10	Obfuscation tool	Obfuscation service used to shorten URL	[32]
11	URL life	Number of days to expiration	[32]
12	URL network	First 3 octets of URL's IP address	[32]
13	URL country	IP address server country	[32]
14	URL city	IP address server city	[32]
15	Connection speed	Connection speed	[30]
16	Days since updated	Number of days since website was updated	[30, 31]
17	Named servers	Named servers	[30, 31]
18	Open ports	Services running on website	[30, 31]
19	ISP	Whois Internet service provider if available	[30, 31]
20	Organization	Whois registered organization	[30, 31]
21	Sitemap hash	If site has hash map embedded	[30, 31]

Table 2: Tweet account characteristics

SN	Criteria	Description	Source
1	Followers	Total Twitter users following user account	[22, 35, 36]
2	Following	Total Twitter users the user account is following	[22, 35, 36]
3	Location	The physical geo-location of account as stated on account	[22, 35, 36]
4	Protected	Protected accounts have their accounts private where only approved followers of the account can see account tweets	[22, 35, 36]
5	Verified	This indicates that an account of public interest has been authenticated by Twitter	[22, 35, 36]
6	Listed count	This indicates that one or more other Twitter users have added the account to a preferred Twitter list	[22, 35, 36]
7	Likes/ favourites	Total number of likes across all tweets	[22, 35, 36]
8	Static count		[22, 35, 36]
9	Created at	Timestamp representing the date the Twitter account was opened	[22, 35, 36]
10	Geo enabled		[22, 35, 36]
11	Contributors enabled	This account allows other Twitter users to post content on their behalf	[22, 35, 36]
12	Language set	Language set	[22, 35, 36]

Finding similarities across these sets of features will reveal if malicious URL links created by the same or similar actor(s) have similar characteristics embed in their metadata. Our fingerprint includes features related to tools by which URL's host, domain, query path length, and string were created.

For example, benign links embedded in Twitter posts will usually contain links to news articles, albums, marketing ads/products, etc. However, long-form of embedded malicious links will have missing path queries or have path queries that are not English-readable or automatically generated. Therefore, on the average malicious paths in URL strings will have shorter lengths than those in benign URL strings as seen in Fig. 1.

Attributing actors by social network account characteristics DF_2

In addition to similar propagation tools, the characteristics of social accounts listed in Table 2 below, are useful in identifying actors of covert cyber-criminal networks in OSNs. It is common for one criminal to hide behind multiple digital personas or multiple digital personas to control a single account interacting with multiple victims [3]. This category of features will therefore identify multiple accounts operated by a single cyber persona or an account operated by multiple criminal entities. The fluid nature of a person's online identity (cyber persona) makes it easier for criminal elements to disguise themselves as legitimate users to exploit unknowing victims. For example, in 2010, masquerading an elite financial consultant, the hacker group Lulzec created a 1 million capacity botnet made up of Facebook users by distributing links to malicious websites luring their victims to book consultation sessions [33]. These bookings were click-baits to further inject users' computers with malicious codes that gives hackers control of users' systems. Similarly, misinformation bots are known to exist on social network platforms such as Twitter, for the sole purpose of spreading propaganda and influencing public opinion. In 2018, Twitter reportedly found 50 000 fake accounts dedicated to actively sharing election-related materials and other automated election-related activities in the USA. Attackers can manage the execution of such elaborate attack by au-

Table 3: Tweet text language characteristics

SN	Criteria	Description	Source
1	Emoji count	Number of ideograms in Twitter post	[50]
2	Punctuation count	Number of special characters or punctuations in Twitter post	[11]
3	Emoticon ratio	Ratio of ideograms to letters (characters) in a tweet	[50]
4	Punctuation ratio	Ratio of punctuation to letters (characters) in a tweet	[11]
5	Capitalization	Number of characters in tweet written in upper case	[11]
6	Alphanumeric count	Number of numerical digits [0–9] in Twitter post	[11]
7	Exclamation count	Number of exclamation marks '!' in Twitter post	[50]
9	Sentences count	Number of sentences in Twitter post as separated by a period (.)	[11]
10	Token count	Number of words in in Twitter post	[11]
11	Token ratio	Ratio of the total number of unique words to the total number of words in the Twitter post	[11]
12	Hashtag count	Number of words in Twitter post beginning with a '#'	[50]
13	Adjectives	Number of Adjectives POS in Twitter post	[42]
14	Conjunctions	Number of Conjunctions POS in Twitter post	[42]
15	Adverbs	Number of Adverbs POS in Twitter post	[42]
16	Delimiters	Number of Delimiters POS in Twitter post	[42]
17	Nouns	Number of Nouns POS in Twitter post	[42]
19	Pronouns	Number of Pronouns POS in Twitter post	[42]
20	Verbs	Number of Verbs POS in Twitter post	[42]
22	Adposition	Number of Adposition POS in Twitter post	[42]
23	Anger	Amount of antagonism present in Twitter post	[51]
24	Anticipation	Quantification for anticipation detected in Twitter post	[51]
25	Disgust	Quantification for disgust detected in Twitter post	[51]
26	Fear	Quantification for fear detected in Twitter post	[51]
27	Joy	Quantification for joy detected in Twitter post	[51]
28	Sadness	Quantification for sadness detected in Twitter post	[51]
29	Trust	Quantification for trust detected in Twitter post	[51]
30	Surprise	Quantification for surprise detected in Twitter post	[51]
31	Negative	Quantification for negative detected in Twitter post	[47]
32	Positive	Quantification for positive detected in Twitter post	[47]

Table 4: Account activity features

SN	Criteria	Description	Source
1	Average user daily posts	Average number of posts from user per day	[21, 24]
2	Average user daily retweets	Average number of retweets from user per day	[21]
3	Average daily retweeted	Average number of time the user is retweeted by others daily	[21]
4	Number of unique retweets	Average number of unique users retweeted by user daily	[21]
5	Number of daily user likes	Average number of daily likes on user's post	[21]
6	Average user daily mentions	Average number of unique users mentioned per post in user's posts	[21]
7	Average posting interval	Average time difference (in minutes) between posts	[21]
8	Average retweet interval	Average time difference (in minutes) between retweets	[21]
9	Average number of hashtag per post	Average number of hashtags used across posts	[21]
10	Average daily replies	Average number of replies User A gets on original content daily	[21]
11	Average daily replied	Average number of tweets User A replied to	[21]
12	Unique number of user likes	Unique number of users who have liked User A's statuses	[21]
12	Unique number of user likes	Unique number of users who have liked User A's statuses	[21]

tomating the creation of malicious accounts using the same account templates [34].

Multiple researchers suggest that hidden characteristics of account attributes can identify accounts that exhibit similar behaviour of having similar underlying characteristics [37]. Another study shows that it is possible to identify similar spam accounts by feeding these features to a fine-tuned *k*-means clustering algorithm similarities in profile information [22]. Similarly, Dudorov et al. uses a condensed set of *k*-principal components to characterize groups of spammers on Twitter based on similarities in account features [38]. Finally, Arshi et al. cluster the account details of Facebook users in order to identify groups of people with similar political, cultural, and social views for conducting targeted decentralized mini campaigns [39]. These authors determine that it is possible to create friendship, communication, and affinity networks based on subtle political and cultural affiliations. The rationale for including profile details is that similar accounts will look similar. For example, consider a group of bots following each other and re-posting each other's posts in a coordinated manner.

Attributing actors by language style DF_3 (characterizing or identifying accounts with the same set people writing posted content)

Every author has unique writing pattern and a representation of an author's use of language within conversations highlights characteristics such as hidden in the structure and syntax of sentences. In our third category of features, we assume that criminal entities automate

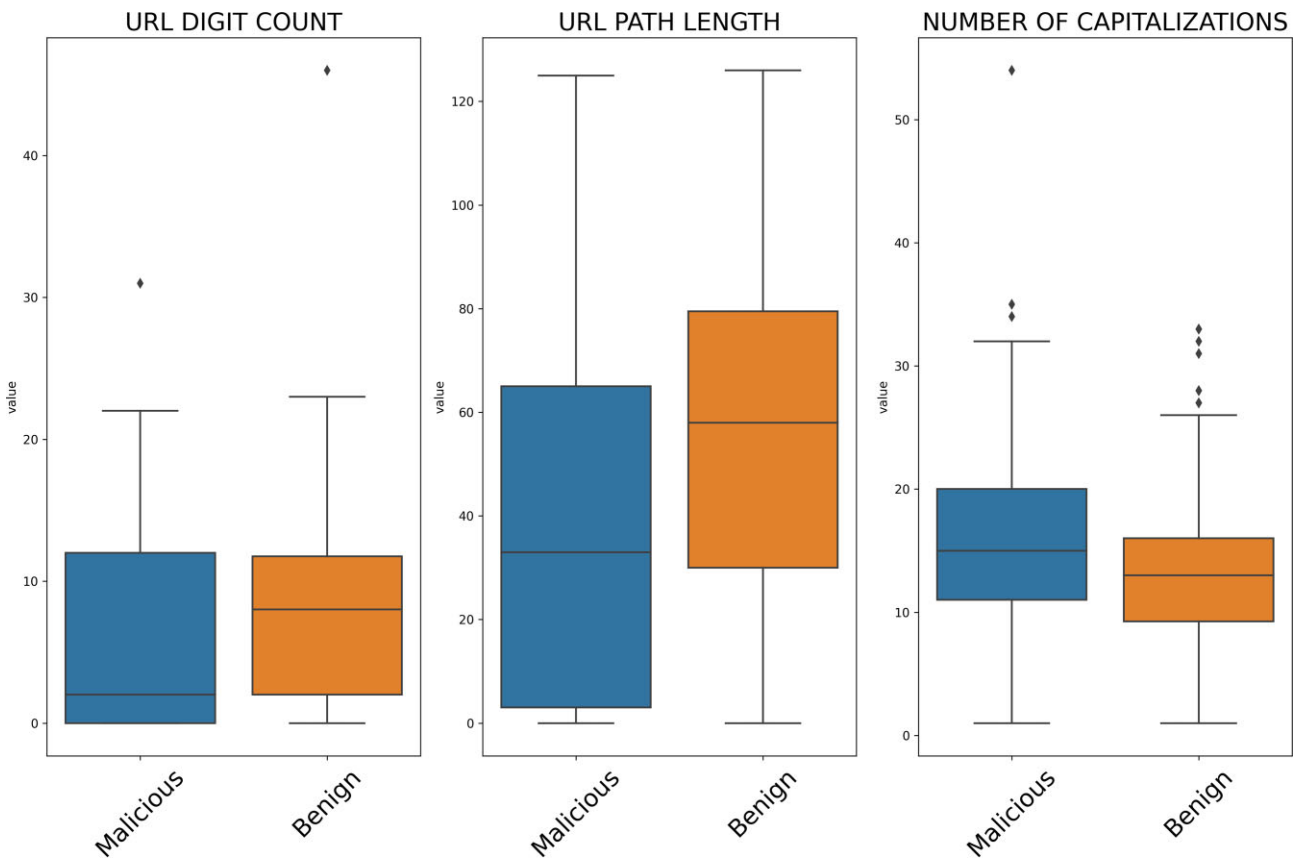

Figure 1: URL features.

the creation of malicious posts. A language fingerprint represents a model of a persona's language within conversations [40]. Therefore, posts from elements within the same criminal network would have similar language characteristics. We expect similarities in language to be less varied amongst colluding malicious users than amongst random independent users. Chen et al. also extend their experiment to demonstrate how attackers create multiple posts from a single template using multiple cyber personas to propagate these malicious posts in a coordinated manner [34].

The study of 'measuring writing style' is called stylometry [41] and its applications are used mainly in Authorship Attribution [11]. Authorship Attribution, is the science of inferring the author of a given piece of text based on embedded language characteristics. In the context of social networks, this practice becomes important in creating clusters of cyber personas based on representative characteristics of language that can reveal hidden groups of accounts with similar language patterns.

In order to extract text similarity features listed in Table 3 below, we combine techniques from simple natural language processing and authorship attribution such as word frequency pattern identification [42], sentiment analysis [43], parts of speech tagging [44] that offer the capability to study the patterns of individual use of language. Combining these two techniques leverages both simple statistical methods and complex linguistically informed computational algorithms to create a representative feature set of language traits observed within a text. These feature set can further be compared to identify groups of text that have similar patterns of writing to infer groups of social media accounts posting similar content.

Word frequency-based pattern identification techniques provides ability to study the types of keywords and their frequency of across texts including punctuations, emojis, and numerical digits. A spectral clustering approach can also highlight similarities between multiple persons online based on the frequency of common words [45]. However, most word frequency-based text-clustering solutions are limited in their ability to address semantic relationships between words, which often results in models that are void of contextual meaning [46]. The inclusion of subjective states of texts largely narrows this gap [47]. Sentiment analysis identifies and quantifies subjective states in a given text while emotion detection quantifies the amount of a specific human emotion intended in a text [48]. Robert Plutchik's wheels of emotions reduces 34 000 human emotions to 8 primary emotion dimensions: anger, fear, anticipation, surprise, joy, sadness, trust, and disgust. Colneric and Demsar successfully use extract human emotions from 73 billion tweets using a recurrent neural network classifier based on Plutchik's emotion categories [49].

Attributing actors by activity characteristics (DF_4) (characterizing or identifying accounts where actors engage with content in a similar way)

Assuming there is a repeating pattern of the same set of users consistently retweeting every new tweet from User A, we would consider this a synchronized activity. If we assume that a group of accounts on Twitter are working together to spread malware, we therefore expect their activities to be similar, synchronized, and correlated. Patterns of activities behind accounts who are controlled by similar people should exhibit similar patterns of usage and association. For exam-

ple, these include indicators from Table 4 below, such as an account retweeting content from the same sets of accounts or in the case of bots, accounts active during certain hours of the day.

Such features in Table 4 below, can be extracted from studying patterns of interaction of bots. Some experts argues that while the priorities of bots and covert hacker groups on social media may differ, their goals are inherently the same, to reach the most amount of people in the most effective way [52]. Features such as repeated behaviour, time interval between activities, similarities in accounts mentioned in original posts, ratio of similarity in followers to following are all areas of interest for identifying accounts that may be colluding [24]. In analysing groups of bots on Twitter, it should be noted that they tend to act similar and interact with similar content and the same user accounts. These set of features are included to address groups of accounts that exhibit similar activity patterns.

For each malicious user, we extract a timeline of the most recent 10 000 tweets—these includes, retweets, mentions, and likes. We create estimates for daily user activity and interaction. We first observe how often a user posts original content (in the case user who posts malicious content). We then measure daily number of retweets and the number of unique users a user retweets (who is retweeting who) [24]. In addition to retweets, we estimate unique bi-directional likes and mentions. Incorporating these features captures accounts with synchronized behaviour as possibly colluding behind the scenes.

The next section outlines the application of an unsupervised learning approach to find latent groups in these digital fingerprints.

Identifying covert groups involved in the propagation of malware on Twitter (analysis and results)

After fingerprinting each account, we transform the features extracted and parse them through an unsupervised learning algorithm to group accounts with similar latent characteristics. To ensure useful features are used for the clustering task, a data pre-processing step is required. In this subsection, we outline details of our data pre-processing and analysis methodology.

Feature transformation

Data cleaning. Data cleaning ensures that data represent the problem space to be modelled. This mainly includes removing all tweets with missing information for any observed feature. Missing data in Twitter API data usuallys result from data not collected due to user privacy restrictions or data collector's API access rights. Removing records with any missing data can potentially lead to loss of information; however, performance of AI models on reduced data have been shown to out-perform AI models built with other type of data replacement techniques [53, 54].

One-hot encoding. One-hot encoding refers to a method for transforming categorical variables—e.g. 'profile text color' and 'location' data—into formats understood by machine learning algorithm [54]. This method creates a binary vector for each unique category and assigns a 1 or 0 to each observation for each category. If an observation has the i^{th} category present, the corresponding vector component is set to 1 and 0 otherwise. After one-hot encoding, our dataset contains 135 features across the four digital fingerprinting groups.

Feature scaling. In most cases, datasets contain multiple features with varying degrees of magnitude, scale, and usefulness to the prediction task. For example, variables on different scales may create spurious bias. As machine learning algorithms are highly sensitive to these huge variations, feature scaling is required to even out these huge variations in the dataset and center feature values around their means. We use a Min–Max scaling as shown below:

$$X_{scaled} = \frac{(X - X_{min})}{(X_{max} - X_{min})}. \quad (1)$$

By using the Min–Max Scaler, all features would be transformed into a range (0 through 1), i.e the minimum value and maximum values for any feature in our dataset would be 0 and 1, respectively.

Feature selection

Feature selection is the process of subsetting initial feature set for better performance of machine learning models. The purpose of feature selection is to remove features that provide no useful information for the learning task and therefore, serve to reduce the quality of results or exponentially increase computational resources. In an unsupervised learning task, useless features can either be multicollinear features or Zero variance features.

Zero-variance features have a variance close to zero $\sigma < -0$ and do not contain any significant pattern of change. On the other hand, multicollinear features are highly correlated with one or more other features in the feature space. They contain no new useful information and may lead to decreased statistical significance or over-fitting.

To remove zero-variance features, we consider that malicious actors are rare observations and set the threshold for 'low variance' to absolute zero. Given the sparse nature of Twitter data and criteria for tweet collection outlined in the previous sections, a lot of features were similar across all users.

To identify multicollinear features, we compute a Pearson's pairwise correlation matrix of all features in our sample. The Pearson's correlation between two variables lies between (-1 for negatively correlated features and 1 for positively correlated features). Two variables are said to be collinear of their correlation coefficient is close to perfectly 1 or -1. We identify all pairs of features with correlation coefficients >0.9 or <-0.9 and finally remove one pair of correlated features. Removing absolute zero-variance variables reduced our feature set to 37 variables while removing multicollinear features removed two more variables from our data. At the end of the feature selection phase, we had 35 features for each tweet in our data.

Identifying collusion

To identify groups of colluding actors, we apply a k-means clustering algorithm to our transformed data to create groups of users with similar digital fingerprints. We assume that 'colluding' users will exhibit certain similarities in their digital fingerprints. K-means is a partition-based algorithm that regards the centre of a cluster as the data point of reference for all other data points in the same cluster. Given a set of observations with no labels, k-means partitions samples into groups based on the Euclidean distance between them.

Let $X = x_i$, $i = 1.....n$ be a set of data points for n Twitter accounts to be separated into a set of k clusters where each cluster should contain several accounts with similar latent fingerprinted features. The goal of k-means is to find several clusters that minimizes the squared error between a cluster mean and corresponding data points in the cluster. If μ_k is the cluster mean of the cluster c_k, then the squared error between μ_k and all data points in c_k is given as

$$J(c_k) = \sum_{x_i \in c_k} ||x_i - \mu_k||^2.$$

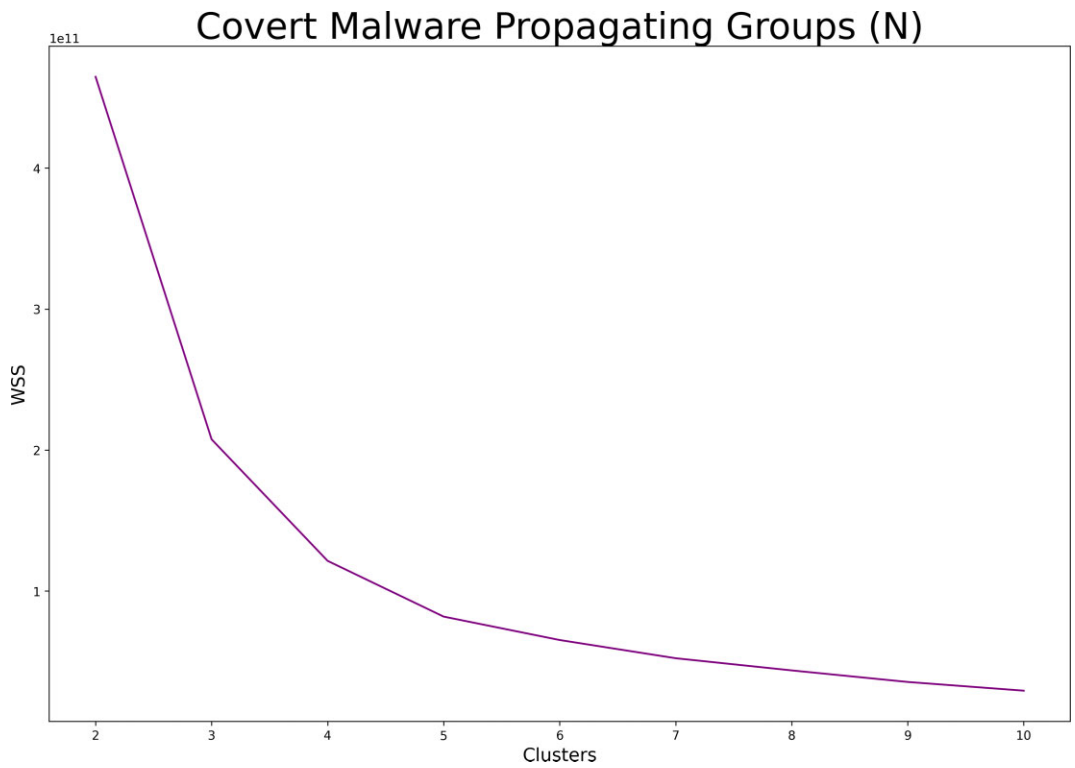

Figure 2: Determining k ($k = 5$).

Therefore, k-means minimizes the sum of squared error over all k clusters:

$$J(c_k) = \sum_{k=1}^{n} J(c_k). \tag{2}$$

K-means estimates similarity between user accounts by calculating the distance between their fingerprinted data points with a distance algorithm (Euclidean distance). For example, given two there the distance between two data points x and y is defined as

$$d(x, y) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}. \tag{3}$$

There are two ambiguous aspects to the k-means clustering algorithm. K-means a localized optimization problem that is sensitive to the selection of cluster centroids. Given we have no prior knowledge of the number of covert groups (clusters) distributing malware on Twitter, a random number k, is assigned. Consequently, choosing bad or largely dissimilar cluster centroids will result in bad clustering results. Researchers most often turn to finding the number of clusters that reduces the overall within cluster sum of squares across all clusters so that creating another cluster does not improve the total within cluster sum of square error [55]. This technique is called the 'Elbow' method [56], which shows a sharp decrease in the within cluster sum of square to form an elbow at the optimal 'k' when plotted on a graph such as in Fig. 2. The k-means algorithm is initially run for several iterations with various values for k and the within cluster sum of square for each run is recorded. The X-axis in the Fig. 1 represents the number of clusters chosen on each run and the Y-axis represents the recorded the within cluster sum of square.

Baseline evaluation setup

Hackers during the COVID-19 pandemic exposed fatal flaws in the infrastructure of social networking sites such as Twitter. We selected this event because it forced unprecedented increase in online presence and collaboration. Specifically, the COVID 'stay-at-home' restrictions from early 2020 to early 2021 created a scenario that emboldened malicious groups and therefore increasing the chances of victims clicking on malicious links. Given the lack of ground truth data, i.e., real data on collusive groups on Twitter, we develop two validation methods to prove our methodology's ability to identify collusion using a cluster-based fingerprinting approach.

The first validation method estimates within cluster malicious URL similarity [57] of users in our clustered data. The purpose of this strategy is to establish baseline similarity between users in the same cluster. For estimating similarity between URLs propagated my malicious users within clusters, we use a three-point-based approach that includes:

- Estimating similarity between two URLs with cosine text similarity approach [58]. Cosine text similarity score falls between 0 (dissimilar) and 1 (similar) and determines closeness' between two text strings. In general, these measures cover two types of similarity, surface closeness (lexical similarity) and meaning (semantic similarity). Surface similarity considers character level similarity, while semantic similarity accounts for the actual meaning behind characters or the entire phrase in context. Given two text strings A and B, the cosine similarity between A and B is estimated as

$$\frac{\sum_{i=1}^{n} A_i * B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}} \tag{4}$$

- Estimating URL host similarity, which includes manual examination for similarity of host-based features. For example, within each colluding cluster, we examine the ratio of unique number of originating countries to the total number of colluding users in the cluster. In addition, unique hosting providers and in-depth

host-based features extracted from shodan [59] are also manually inspected.

In the second method, we collected data from five groups of Twitter accounts each known to be collaborating to disseminate similar information. To collect this information, we target tweets from the 2021 UK Scottish Elections across the five major Scottish political parties: The Scottish National Party, Scottish Conservatives, Scottish Labour, Scottish Greens, and The Scottish Liberal Democrats. We selected this event because it was a major political event therefore, there is increased cooperating Twitter activity between members of the same political group in the days leading up to the elections. To simulate 'collaborating groups of tweets', we collect tweets from the party's official Twitter accounts, party's leader and their top five parliamentary MPs. We identify the top 10 active parliamentary representatives for each party by estimating the friends to follower's ratio for all parliamentary representative within each party.

We collect tweets from these accounts and group them as 'collaborating accounts' working together to share similar tweets on their timelines. Our final evaluation data contain 2552 tweets from 34 unique collaborating users, i.e. five groups, each with six or seven collaborating users as shown in Table 5. These tweets represent those filtered as containing an embedded URL to ensure evaluation data corresponds with the methodology. Although, these are not malicious URLs and these accounts are not attributed to 'maliciousness', these accounts are suited for our evaluation as they represent groups of accounts working together to disseminate similar information. We expect members of the same group to actively interact with post shared by each other during election campaigns than otherwise with members of a different political party. The results of the analysis are outlined in the following section.

Results

The graph shows three 'elbow' points, at 3-clusters ($k = 3$), 4-clusters ($k = 4$), and at 5-clusters ($k = 5$). Table 6 below shows the percentage sample population for each cluster at $k = 5$. Within our selected sample and timeframe, our unsupervised learning identifies five groups of malicious actors. Because k-means minimizes the squared errors, assigning outlier points to their own clusters gives the optimal results. These five groups of actors are assumed to be actively propagating malicious links within the first week of the COVID-19 pandemic on Twitter. We estimate the total number of unique malicious links propagated and the average activity rate for each cluster. A user activity includes posting, retweeting, liking, and commenting. Activity rate is estimated as the average number of likes, retweets, mentions, and posts by actors in a cluster.

During the start of the week of the COVID-19 pandemic, a total of 80 unique malicious links were being spread on Twitter by 109 unique users. Our model estimates that these 109 users can be represented as five groups based on similarities in propagation tools, account details, and language style. The organization behind the spread of malicious code during the COVID-19 pandemic is assumed to be represented by these groups. Figure 3 shows key variations in digital personas of malicious and benign users on Twitter. In comparison to tweets from benign users, tweets from malicious actors are less varied in key features of their digital fingerprints.

For example, although malicious tweets collectively portray higher levels of anxiety driven emotions such as fear, surprise, and anger, benign tweets are more varied on these. Activity rates such as time intervals between posting and retweeting are also observed to be more synchronized with malicious tweets. Also note that URL links used by malicious actors have shorter path lengths and alpha numeric digits in them as seen in Fig. 1. Excluding outliers, malicious actors post at least one new content between 0 and 3 days as opposed to a highly frequent posting interval of at least three original tweets per day by benign actors.

In the next section, we provide a cluster-based evaluation of the methods presented in this paper. Firstly, we manually examine similarities between the URLs propagated within each cluster in our baseline dataset and assign a URL similarity [57] score to each cluster. Clusters with cosine similarity scores closer to one are more likely to have created their malicious URLs using the same tools or have these URLs created by the same person/s or entities.

Secondly, we create an experimental setup of simulated colluding accounts of five groups of political actors actively cooperating to spread campaign information on Twitter. We fingerprint and cluster these accounts and estimate how well our digital fingerprints can capture these groups of 'colluding' actors.

To evaluate our methodology against the benchmark data, we start by assigning a label to each tweet indicating its party grouping and fingerprinting each tweet. We then run an unsupervised learning model to assign each tweet to a cluster. Table 6 shows the comparison between cluster assignments and original party groupings. We compare the extent to which digital fingerprints of these collaborating accounts are grouped together with an unsupervised learning model. The resulting Table 7 below is a confusion matrix of the total number of correctly clustered users.

Note that the assigned cluster for each group is shaded grey in Table 7. We measure the 'fitness' of our fingerprinting methodology by calculating the accuracy and sensitivity of the cluster assignments. We use the accuracy as a general measure of how well our digital fingerprints can separate groups of colluding users. In addition, we choose to evaluate the sensitivity of cluster assignments as the identification of truly colluding actors is essential, with a certain tolerance for accepting some falsely identified colluding actors. To do this, we define the following terms in relation to our cluster assignments:

- True positives: members of a group classified correctly clustered as that group, e.g. Group 5 → Cluster 5 [3].
- True negatives: non-members of a group correctly clustered as non-members of the group, e.g. Group 4 → Cluster 5[3]
- False positives: non-members of the group wrongly clustered as members of the group, e.g. (Group 3, Group 4) → Cluster 4 [1, 1].
- False negatives: members of the group wrongly clustered as non-members of the group, e.g. (Cluster 2, Cluster 3) → Group 5 [2, 1].

For clarity, we produce the following summary in Table 8:

In Table 8 above, we use standard performance metrics in evaluating how well our digital fingerprints can identify colluding users and groups of colluding users. The accuracy measures the ability of digital fingerprints to identify collusive behaviour amongst two or more users. Ninety % of the time, the model can identify groups of users engaged in similar collusive behaviour.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} = \frac{24 + 160}{24 + 160 + 9 + 10} = \frac{184}{203} = 90.6\%$$

The sensitivity here measures the ability of digital fingerprints to correctly include malicious actors in their corresponding groups and exclude them from groups to which they do not belong. Seventy %

Table 5: Evaluation setup summary

Group	Number of unique users	Average number of tweets per user	Unique number of URLs tweeted	Total tweets collected
Group 1	7	64	22	450
Group 2	7	66	21	465
Group 3	7	70	16	491
Group 4	7	84	24	591
Group 5	6	92	33	555

Table 6: Summary of malicious links propagated within clusters

Cluster	Users	Unique malicious links	Users to unique host ratio	Inter-cluster URL CS	Inter-cluster tweet CS
1	23	19	0.8	0.68	0.52
2	5	4	0.5	0.62	0.55
3	77	47	1.0	0.81	0.65
4	3	3	1.0	0.73	0.78
5	10	7	1.0	0.92	0.76

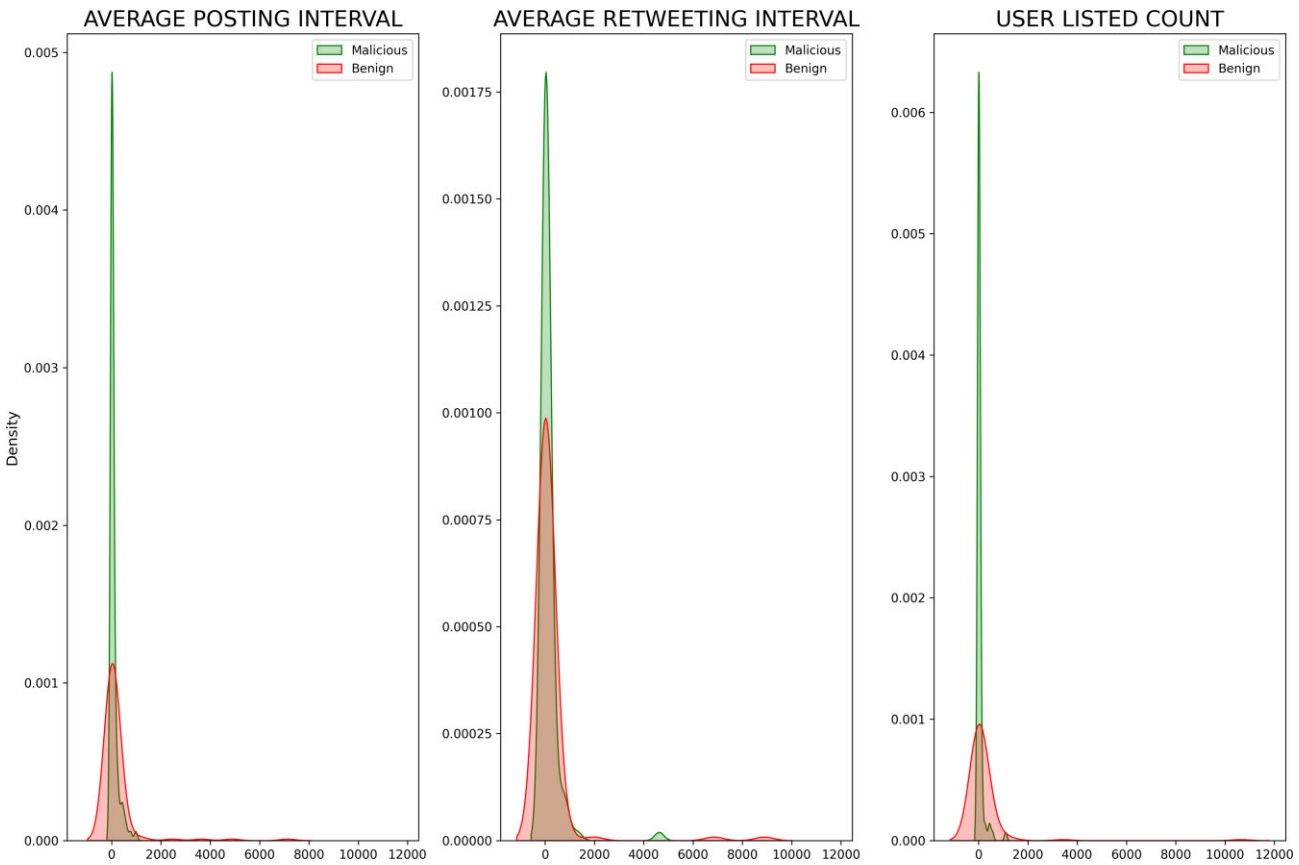

Figure 3: Tweet activity features.

Table 7: Cluster—group assignment matrix

	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5
Group 1 (slp)	7	0	0	0	0
Group 2 (scp)	0	5	1	1	0
Group 3 (sgp)	0	2	4	0	1
Group 4 (snp)	1	0	0	5	1
Group 5 (sld)	0	2	1	0	3

Table 8: Evaluation metrics for groups

	TP	TN	FN	FP
G1	7	33	0	1
G2	5	30	2	4
G3	4	32	3	2
G4	5	33	1	1
G5	3	32	3	2
Total	24	160	9	10

of the time, the model correctly places malicious users in their corresponding collusive groups while 78.8% of the time, the model correctly excludes malicious users from the wrong collusive groups. This measure, therefore,

$$Sensitivity = \frac{TP}{TP + FN} = \frac{24}{24 + 10} = \frac{184}{203} = 70.5\%$$

As these fingerprints rightly identify groups of colluding actors, it is sensitive to the inclusion of non-group members into an identified group. While, clustering the digital fingerprints can identify 'collusion', as evidenced by our evaluation setup, it falls short of identifying multiple groups of colluding actors at the same time. The results prove that the digital fingerprints are efficient in separating collusion activities for a group of users from what would be considered a normal online relationship between other groups of users. Further research may address exploring latent features of digital fingerprints that identify various types of malicious actors. Such research would enable 'family-like' grouping of malicious actors on OSNs.

Conclusion

In this paper, we addressed the problem of finding malicious actors spreading malicious links in OSNs. Our research proposed a methodology that creates a digital footprint for cyber personas in OSNs. Our digital footprint was created with four facets of a person's online presence while propagating malicious links—characteristics of the URL shared, account characteristics, activity characteristics, and characteristics of language used in shared posts. We collected data around a major event and created a digital fingerprint for each account. Using an unsupervised learning model, assuming no a prior knowledge of group affiliation, we grouped these digital fingerprints into clusters of collusion and estimated the joint probability of collusion in any one of our four facets within each cluster.

The evaluation on labelled data proves that using our digital fingerprinting methodology is efficient in identifying the 'collusion' among certain users. As we can detect collusive behaviour among a group of users in online OSNs, the immediate implication is the extension of this method to spot malicious activities beyond the propagation of malware. Hacker groups are known to co-ordinately use OSN platforms as a vector for cyberattacks during cyber hacktivism events [60].

Furthermore, beyond using embedded malicious URLs to identify malicious actors, the methods presented in this paper are useful in identifying other types of malicious actors. For example, OSNs such as Twitter, Reddit, and Facebook have been identified as recruiting and training grounds for deep covert terrorist activities [61] used as communication medium for sleeper cells [62].

Finally, there is a need for further improvements to our methodology to capture 'true groups' of colluding users also suggesting the need for further research into simultaneously identifying multiple groups of actors involved in collusive behaviour.

Acknowledgement

This work was supported by the Economic and Social Research Council under Grant ES/S008853/1.

Author contributions

Ruth Ikwu (Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Resources, Writing – original draft), Luca Giommoni (Conceptualization, Investigation, Project administration, Resources, Supervision, Writing – review & editing), Amir Javed (Formal analysis, Investigation, Methodology, Resources, Writing – review & editing), Pete Burnap (Funding acquisition, Project administration, Writing – review & editing), and Matthew Williams (Funding acquisition, Writing – review & editing)

References

1. Santhiya K, Bhuvaneswari V, Murugesh V. Automated crime tweets classification and geo-location prediction using big data framework. *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, 2021;**12**:2133–52.
2. Javed A, Ikwu R, Burnap P,. *et al.* Disrupting drive-by download networks on Twitter. *Soc Netw Anal Min* 2022;**12**:1–17.
3. Rashid A, Baron A, Rayson P,. *et al.* Who am I? Analyzing digital personas in cybercrime investigations. *Computer* 2013; **46**: 54–61.
4. Mueller RS, Dershowitz A. *The Mueller Report: The Final Report of the Special Counsel into Donald Trump, Russia, and Collusion Simon and Schuster*. Simon and Schuster. 2019, NYC, USA.
5. Kumar S, Cheng J, Leskovec J,. *et al.* An army of me: Sockpuppets in online discussion communities. In: *Proceedings of the 26th International Conference on World Wide Web*, Perth, Australia, 2017, p. 857–66.
6. Sanzgiri A, Hughes A, Upadhyaya S. Analysis of malware propagation in Twitter. In: *2013 IEEE 32nd International Symposium on Reliable Distributed Systems*, Braga, Portugal, 2013, p. 195–204.
7. Aggarwal A, Rajadesingan A, Kumaraguru P. PhishAri: automatic realtime phishing detection on Twitter. In: *2012 eCrime Researchers Summit*, Las Croabas, PR, 2012, p. 1–12.
8. Lynch J. Identity theft in cyberspace: crime control methods and their effectiveness in combating phishing attacks. *Berkeley Tech LJ* 2005;**20**:259.
9. Baker WE, Faulkner RR. The social organization of conspiracy: illegal networks in the heavy electrical equipment industry. *Am Sociol Rev* 1993;**58**:837–60.
10. Javed A, Burnap P, Williams ML,. *et al.* Emotions behind drive-by download propagation on Twitter. *ACM Trans Web (TWEB)* 2020;**14**:1–26.
11. Wang D, Navathe SB, Liu L,. *et al.* Click traffic analysis of short URL spam on Twitter. In: *9th IEEE International Conference on Collaborative Computing: Networking, Applications and Worksharing*, Austin, TX, 2013, p. 250–9.
12. Clark D. Characterizing cyberspace: past, present and future. *MIT CSAIL, Version* 2010;**1**:2016–28.
13. Uttley M, Wilkinson B, Van Rij A. A power for the future? Global Britain and the future character of conflict. *Int Aff* 2019;**95**:801–16.
14. Klimburg A. Cyberspace and Governance - A Primer. 2012, https://www.ssoar.info/ssoar/bitstream/handle/document/43560/ssoar-2012-klimburg_et_al-Cyberspace_and_governance_-_a.pdf (Accessed: 19 July 2023).
15. Sundaresan S, McCoy D, Afroz S,. *et al.* Profiling underground merchants based on network behavior. In: *2016 APWG Symposium on Electronic Crime Research (eCrime)*, Toronto, ON, 2016, p. 1–9.
16. Pastrana S, Hutchings A, Caines A,. *et al.* Characterizing eve: Analysing cybercrime actors in a large underground forum. In: *21st International Symposium, RAID 2018*, Heraklion, Crete, Greece, 2018, p. 207–27.

17. Flores-Garrido M, García-Velázquez LM, Cortez-Madrigal RS. Clustering of Twitter networks based on users' structural profile. In: OO Vergara-Villegas, VG Cruz-Sánchez, JH Sossa-Azuela, JA Carrasco-Ochoa, JF Martínez-Trinidad, JA Olvera-López (eds.), *Pattern Recognition. MCPR 2022*. Cham: Springer, 2022, 15–24.
18. Knoke D, Yang S. Social Network Analysis. Thousand Oaks, CA: Sage Publications, 2019.
19. Faghani MR, Saidi H. Malware propagation in online social networks. In: *2009 4th International Conference on Malicious and Unwanted Software (MALWARE)*, Montreal, QC, 8–14, 2009.
20. Zhang CM, Paxson V. Detecting and analyzing automated activity on Twitter. In: N Spring, GF Riley (eds.), *Passive and Active Measurement. PAM 2011*. Berlin, Heidelberg: Springer, 2011, 102–111,
21. Wojcik S, Messing S, Smith A,. *et al*. Bots in the Twittersphere. Pew Research Center, WA, USA. 2018.
22. Adewole KS, Han T, Wu W,. *et al*. Twitter spam account detection based on clustering and classification methods. *J Supercomput* 2020;**76**:4802–37.
23. Chavoshi N, Hamooni H, Mueen A. Identifying correlated bots in Twitter. In: *International Conference on Social Informatics*. p. 14–21, Springer, Bellevue, WA, USA. 2016.
24. Dutta HS, Chetan A, Joshi B,. *et al*. Retweet us, we will retweet you: spotting collusive retweeters involved in blackmarket services. In: *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, Barcelona, Spain, 2018, 242–9.
25. Dutta HS, Chakraborty T. Blackmarket-driven collusion among retweeters–analysis, detection, and characterization. *IEEE Trans Inf Forensics and Secur* 2019; **15**:1935–1944.
26. Adewole KS, Anuar NB, Kamsin A,. *et al*. Malicious accounts: dark of the social networks. *J Netw Comput Appl* 2017;**79**:41–67.
27. Chandra RV, Varanasi BS. Python requests essentials. Packt Publishing, Birmingham, UK. 2015.
28. Morales JA, Al-Bataineh A, Xu S,. *et al*. Analyzing and exploiting network behaviors of malware. In: *Security and Privacy in Communication Networks: 6th Iternational ICST Conference, SecureComm 2010*, Singapore, 2010, p. 20–34.
29. Neumann A, Barnickel J, Meyer U. Security and privacy implications of URL shortening services. In: *Proceedings of the Workshop on Web 2.0 Security and Privacy*. Oakland, California. 2010.
30. Astorino A, Chiarello A, Gaudioso M,. *et al*. Malicious URL detection via spherical classification. *Neural Comput Appl* 2017;**28**:699–705.
31. Sahoo D, Liu C, Hoi SCH. Malicious URL detection using machine learning: a survey. *arXiv:1701.07179*. 2017;**1**:1–37.
32. Wanda P, Jie HJ. URLDeep : continuous prediction of malicious URL with dynamic deep learning in social networks. *Int J Netw Secur* 2019;**21**:971–8.
33. Olson P. Conspiracy (Drives Us Together). *We are anonymous*. London, UK, William Heinemann of Random House, 2013.
34. Chen C, Zhang J, Xiang Y. *et al*., Spammers are becoming "Smarter" on Twitter. *IT Professional* 2016;**18**:66–70.
35. Ackerman S, Schutte K. Social media as a vector for cyber crime. Clark Shaefer Consulting. OH, USA. 2015.
36. Willis A, Fisher A, Lvov I. Mapping networks of influence: tracking Twitter conversations through time and space. *Journal of Audience & Reception Studies*, 2015;**12**:494–530.
37. Castillo C, Mendoza M, Poblete B. Information credibility on Twitter. In: *Proceedings of the 20th International Conference on World Wide Web*, New York, NY, 2011, 675–84.
38. Dudorov D, Stupples D, Newby M. Probability analysis of cyber attack paths against business and commercial enterprise systems. *Proceedings - 2013 European Intelligence and Security Informatics Conference, EISIC 2013*, Uppsala, Sweden, 2013, 38–44.
39. Arsic B, Bašić M, Spalević P,. *et al*. Facebook profiles clustering. In: *6th International Conference on Information Society and Technology ICIST*, Dalian, China. 2016, 154–8.
40. Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate. arXiv:14090473. 2014.
41. Bhargava M, Mehndiratta P, Asawa K. Stylometric analysis for authorship attribution on Twitter. In: V Bhatnagar, S Srinivasa (eds.), *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. **8302**. Cham: Springer International Publishing, 2013, 37–47.
42. Pokou YJM, Fournier-Viger P, Moghrabi C. Authorship attribution using variable length part-of-speech patterns. In: *Proceedings of the 8th International Conference on Agents and Artificial Intelligence - Volume 2: ICAART*, Rome, Italy, 2016, 354–61.
43. Medhat W, Hassan A, Korashy H. Sentiment analysis algorithms and applications: a survey. *Ain Shams engineering journal* 2014; **5**: 1093–13.
44. Manning CD. Part-of-speech tagging from 97% to 100%: is it time for some linguistics?In: *Processing: 12th International Conference, CICLing*, Tokyo, Japan, 2011, 171–89.
45. Singh K, Shakya HK, Biswas B. Clustering of people in social network based on textual similarity. *Perspect Sci* 2016;**8**:570–3.
46. Brysbaert M, Buchmeier M, Conrad M,. *et al*. The word frequency effect. *Experimental Psychology*, 2011. **58**: 412–424
47. Tang D, Wei F, Qin B,. *et al*. Coooollll: a deep learning system for Twitter sentiment classification. In: *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, Dublin, Ireland, 2014, 208–12,
48. Plutchik R. A psychoevolutionary theory of emotions. *Soc Sci Inf* 1982;**21**:529–53.
49. Colnerič N, Demšar J. Emotion recognition on twitter: comparative study and training a unison model. *IEEE Trans Affect Comput* 2018;**11**:433–46.
50. Ohlhorst FJ. Big Data Sources, *Big Data Analytics: Turning Big Data into Big Money*, Vol. **65**. NYC, USA, John Wiley and Sons, 2012.
51. Calefato F, Lanubile F, Novielli N. Emotxt: a toolkit for emotion recognition from text. In: *Seventh International Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW)*, San Antonio, TX, USA, 2017, 79–80.
52. Dodds PS, Harris KD, Kloumann IM,. *et al*. Temporal patterns of happiness and information in a global social network: hedonometrics and Twitter. *PLoS One* 2011;**6**:e26752.
53. Blazek K, van Zwieten A, Saglimbene V,. *et al*. A practical guide to multiple imputation of missing data in nephrology. *Kidney Int* 2021;**99**:68–74.
54. Yu L, Zhou R, Chen R,. *et al*. Missing data preprocessing in credit classification: one-hot encoding or imputation? *Emerg Mark Finance Trade* 2022;**58**:472–82.
55. Hair JF. *Multivariate Data Analysis: An Overview*. Berlin, Heidelberg. International Encyclopedia of Statistical Science, Springer, 2009.
56. Kodinariya TM, Makwana PR. Review on determining number of cluster in k-means clustering. *Int J* 2013;**1**:90–5.
57. Luu VT, Forestier G, Weber J,. *et al*. A review of alignment based similarity measures for web usage mining. *Artif Intell Rev* 2020;**53**:1529–51.
58. Rahutomo F, Kitasuka T, Aritsugi M. Semantic cosine similarity. In: *The 7th international student conference on advanced science and technology ICAST, Vol. 4*, Seoul, South Korea, 2012, 1.
59. Matherly J. *Complete Guide to Shodan*, Vol. **1**. Victoria, BC, Canada, LeanPub, 2015.
60. Pawlicka A, Choraś M, Pawlicki M. Cyberspace threats: not only hackers and criminals. Raising the awareness of selected unusual cyberspace actors–cybersecurity researchers' perspective. In: *15th International Conference on Availability, Reliability and Security*, Ireland, UK. 2020, 1–11.
61. Chatfield AT, Reddick CG, Brajawidagda U. Tweeting propaganda, radicalization and recruitment: Islamic state supporters multi-sided twitter networks. In: *16th Annual International Conference on Digital Government Research*, Phoenix, Arizona, 2015, 239–49.
62. Gialampoukidis I, Kalpakis G, Tsikrika T,. *et al*. Detection of terrorism-related twitter communities using centrality scores. In: *Proceedings of the 2nd International Workshop on Multimedia Forensics and Security*, Bucharest Romania, 2017, 21–5.