UPPSALA
UNIVERSITET

UPTEC STS 20010
Examensarbete 30 hp
Maj 2020

Fantastic bots and where to find them

Agaton Svenaeus

UPPSALA
UNIVERSITET

Teknisk- naturvetenskaplig fakultet
UTH-enheten

Besöksadress:
Ångströmlaboratoriet
Lägerhyddsvägen 1
Hus 4, Plan 0

Postadress:
Box 536
751 21 Uppsala

Telefon:
018 – 471 30 03

Telefax:
018 – 471 30 00

Hemsida:
http://www.teknat.uu.se/student

Abstract

Fantastic bots and where to find them

Agaton Svenaeus

Research on bot detection on online social networks has received a considerable amount of attention in Swedish news media. Recently however, criticism of the research field of bot detection on online social networks has been presented, highlighting the need to investigate the research field to determine if information based on flawed research has been spread. To investigate the research field, this study has attempted to review the process of bot detection on online social networks and evaluate the proposed criticism of current bot detection research by: conducting a literature review of bots on online social networks, conducting a literature review of methods for bot detection on online social networks, and detecting bots in three different politically associated data sets with Swedish Twitter accounts using five different bot detection methods. Results of the study showed minor evidence that previous research may have been flawed. Still, based on the literature review of bot detection methods, it was determined that this criticism was not extensive enough to critique the research field of bot detection on online social networks as a whole. Further, problems highlighted in the criticism were recognized to potentially have arose from a lack of differentiation between bot types in research. An insufficient differentiation between bot types in research was also acknowledged as a factor which could lead to difficulties in generalizing the results from bot detection studies measuring the effect of bots on political opinions. Instead, the study acknowledged that a good bot differentiation could potentially improve bot detection.

Handledare: Matteo Magnani
Ämnesgranskare: Niklas Wahlström
Examinator: Elísabet Andrésdóttir
ISSN: 1650-8319, UPTEC STS 20010

Sammanfattning

Efter det svenska riksdagsvalet 2018 publicerade Totalförsvarets forskningsinstitut en rapport vilken utvärderat politiska diskussioner på Twitter. Rapporten hävdade att 6% av de undersökte Twitter-kontona som varit delaktiga i politiska diskussionerna i samband med riksdagsvalet var botar, med botar syftandes på konton vilka uppvisat ett automatiserat beteende. Ett flertal svenska nyhetsmedier rapporterade om Totalförsvaret hävdade förekomst av botar på Twitter och citat som ” Bedömare befarar nu att en armé av botar ska störa valet genom negativa nyheter, falsk information, uppvigling och splittring. På partikanslierna tar man uppgifterna på största allvar.” förekom i artiklarna. Nyligen presenterades dock kritik riktad mot botdetektering på sociala medier som forskningsfältet, med andra ord kritik riktad mot de metoder som använts för att urskilja vilka konton som är botar på sociala medier. För att säkerställa att information som spridits av svenska nyhetsmedier inte är baserad på felaktig fakta har därför denna studie, granskat de processer som används för att detektera botar på sociala medier, samt evaluerat den kritiken av forskningsfältet som presenterats. För att kunna uppnå dessa två mål utfördes en litteraturstudie av bot detekteringsmetoder på sociala medier, en litteraturstudie av olika typer av botar, samt ett experiment där fem olika bot-detekteringsmetoder användes för att detektera botar i tre olika data set med svenska Twitter-konton som diskuterat politik.

Resultatet av studien visade på små indikationer på att kritiken hade belägg för sina påståenden. Dock, när kritiken utvärderades i relation till litteraturstudien av bot-detekteringsmetoder drogs slutsatsen att kritiken inte var omfattande nog för att inkludera hela forskningsfältet. Ett antal av de problem i bot-detekteringsmetoder som belystes i kritiken konkluderades också potentiellt uppkomma på grund av att studierna saknat en tydlig differentiering mellan olika typer av botar. En mer välutvecklad differentiering och kategorisering av olika typer av botar identifierades emellertid också som en möjlig faktor vilken skulle kunna förbättra nuvarande bot-detekteringsmetoder. Slutligen, en saknad av tydlig differentiering mellan olika typer av botar identifierades även som en faktor vilken skulle kunna försvåra möjligheten att generalisera resultatet från studier som uppmäter botars förmåga att påverka människors politiska åsikter.

Contents

1 Introduction

On the 9th of September 2018, Sweden successfully finished their general election with the highest voter turnout since 1985 [87]. The voter turnout increased from 85,81% in 2014 to 87.18% in 2018 [96][95]. However, in the aftermath of the general election 2018, the Swedish Research Agency released a study on political bots on Twitter and their influence on the Swedish general election, *Political Bots and the Swedish General Election* [30]. The results of the study showed that 6% of the examined accounts active in Swedish political discussion on Twitter were suspected to be bots, bots in the study referring to accounts displaying an automated behavior [30, p.124–127]. A memo indicating the same results as [30] was also published prior to the election by the Swedish Research Agency [31]. The research by the Swedish Research Agency received a considerable amount of attention in Swedish news medias, Aftonbladet, Svenska Dagbladet, Dagens Nyheter, SVT, Expressen, TV4, Ny Teknik, Hela Hälsingland, Göteborgsposten, Sveriges Radio, all reported about the findings [4][94][91][52][42][82][74][3][25]. A snapshot of the articles reporting on the study is provided in the following quotes:

- ”En växande armé av botar – automatiserade konton – attackerar islam, liberala partier och etablerad media, men älskar SD”[3]. (Authors translation: *A growing army of bots – automated accounts – attacks Islam, liberal parties and established media, but loves SD.*

- ”Bedömare befarar nu att en armé av botar ska störa valet genom negativa nyheter, falsk information, uppvigling och splittring. På partikanslierna tar man uppgifterna på största allvar.”[25]. (Authors translation: *Assessors fear that an army of bots will interfere with the election through fake news, incitement and disruption. The party offices is taking this information most seriously.*

- ”Det är tydligt att botverksamheten som beskrivs i studien gynnar högerpopulistiska och högerextrema grupperingar. Botarna bidrar till samhällspolarisering, vilket också gynnar Ryssland.” [52]. (Authors translation: *It is obvious that the bot activity described in the study favors right-wing populists and far right formations. The bots add to the societal polarization which favors Russian.)*

Although these excerpts from news articles only provide a mere glance at a much broader context, it clearly indicates strong reactions in Swedish news media. Furthermore, the Swedish election was not the only event which researchers examined and found suspected bots active in political Twitter discussions. A study of the 2017 French Presidential election found that 18342 out of 99378 accounts using the political hashtag #MacronLeaks on Twitter were suspected to be bots [33, p.8]. Using the same bot detection framework as in the French Presidential examination, Botometer[1], both the 2018 US Midterms and the US 2016 Presidential Election were examined. The findings of the Botometer research showed that 21.1% of the examined Twitter accounts discussing the Midterms were suspected to be bots, while 15% of the examined Twitter accounts active in political discussion regarding the Presidential Election were suspected to be bots [32, p.5, p.12]. Oxford University has even launched a specific

[1] `https://botometer.iuni.iu.edu`

project to examine algorithms and automation in politics, *The Computational Propaganda Research Project* (COMPROP)[2]. Similar to the reactions in Swedish news medias, the research on Twitter bots active in a political context caught the attention of US news media. For instance, The New York Times and The Atlantic reported on Twitter bot research conducted by COMPROP on the 2016 US Presidential Election [93][8], the following quotes are excerpts from these articles:

- ”Propaganda bots made a powerful showing during Election 2016”[93]
- ”How Twitter Bots Are Shaping the Election ”[8]

The newspaper Time also reported on Twitter bot activity in 2016 US Presidential Election but referred to research from Swansea University and University of California instead [38], the article states as follows:

- ”Twitter bots may have altered the outcome of two of the world’s most consequential elections in recent years ”[92]

However, the work conducted on bots and bot detection on online social media, such as Twitter, is not undisputed. The above quoted article from Time is based on a paper from Yuriy Gorodnichenko, Tho Pham and Oleksandr Talavera, researchers whose work on Twitter bot detection has been questioned. Former Google employee Mike Hearn, who worked with anti-automation platforms stated that their criteria for detecting bots are hopeless [46]. For example, Hearn raises concerns regarding the criteria ”abnormal tweeting time (from 00:00 to 06:00 UK time) ” [38, p.8] since real people according to Hearn have been known to tweet after midnight, which in turn may lead to real people being classified as bots [46]. A more in depth criticism of bot detection on online social networks was presented by the German journalist Michael Kreil at the OpenFest Conference in Sofia, Bulgaria, 2019. Kreil’s talk *The Army that Never Existed: The Failure of Social Bots Research* argued that there are foundational problems in the research from three of the biggest research groups in the field of bot detection. Similar to the criticism from Hearn, Kreil questioned the criteria which the researchers use for bot detection [58]. In addition, Kreil claimed that the machine learning based tool Botometer, a bot detection framework used by researchers to detect bots on Twitter, has an unacceptably high misclassification rate [58]. For instance, when evaluating the Botometer framework Kreil found that out of 396 Twitter profiles belonging to staff members of the German news agency Deutsche Presse-Agentur, 142 or 35.9% were missclassified as bots [58]. Given the claimed serious problems in the research field of detection of bots on online social networks, Kreil stated that all papers within the research field should be reviewed and revoked if necessary [58].

Herein lies a potential problem: If the existing research on bot detection is criticized and shown to be partly incorrect as argued by critics, then there is no consensus on either the degree to which bots are present in online social networks discourse, or their influence on political opinions or elections. Still, it would seem that extensive reporting on the bot phenomenon has been done as demonstrated above, where bots are described as not only a

[2] `https://comprop.oii.ox.ac.uk`

given presence in some online social network discussions, but also that they have a measurable impact on political opinions or elections. As such, the uncertainty within the research field on bot detection is at odds with the reporting of bot prevalence by news media worldwide.

Such a discrepancy between the research community and the news media discourse risks undermining the legitimacy of the news reporting on bot prevalence. Established news media are commonly known to affect democratic politics and policy makers [7, p.25–26], and news reporting based on flawed research may therefore lead to misguided political responses or policy interventions based on the perceived threat of bots' influence online. In California the state passed the law *Bots: disclosure*[3], where bot accounts are required by law to reveal that they are bot accounts in order to prevent them misleading others in regards to influencing political opinions or incentivizing the purchase or sales of products.

Given the considerable amount of attention that research on bot detection on online social networks have received in Swedish news media, the above proposed criticism warrants for further investigation into the research on bot detection on online social networks to determine if information based on flawed research is being spread in Sweden. This study will therefore:

- Conduct a literature review of bots on online social networks.
- Conduct a literature review of methods for bot detection on online social networks.
- Use five different bot detection methods to detect bots in three different politically associated data sets with Swedish Twitter accounts.

To be able to:

- Review the process of bot detection on online social networks.
- Evaluate the proposed criticism of current bot detection research.

This study is structured as follows: Section 2 provides literature reviews of both bot detection research and research of bots on online social networks, a summary of the criticism of bot detection research is also included in the section, as well as an overview of the algorithm random forest, one of the three methods in the study used to detect bots with. Section 3 describes the data used in the study, including how the data was obtained. Section 4 describes the carried out process of detecting bots in politically associated data with Swedish Twitter accounts. Section 5 displays the result obtain from the process described in Section 4. In Section 6 the result in Section 5 is discussed, the process of bot detection and the proposed criticism is also discussed in the light of the literature reviews provided in Section 2. Lastly, Section 7 provides a conclusion of the study and suggestions for further research.

[3] https://leginfo.legislature.ca.gov/faces/billTextClient.xhtml?bill_id=201720180SB1001

2 Theory

2.1 General definition of bots

In the early stages of bot development, bots were in general terms defined as autonomous agents, systems pursuing their own agenda by sensing, reacting and acting in accordance to the environment they were placed within [35, p.5]. The term has since been used in numerous different settings to refer to different types of objects[4] [61]. The following list provide some examples of how bots have been defined in bot classification literature.

- A social media account that is predominantly controlled by software rather than a human user [32, p.3].
- Accounts operated by programs instead of humans [19, p.1].
- Non-personal and automated accounts that post content to online social networks [62, p.309].
- Automated programs [105, p.21].
- An automated social program [75, p.92].

These provided examples of bot-definitions do not in any way include every aspect of how bots are defined within the bot detection literature, they do however display a common pattern which can be found in the definitions of bots, automatization is included in the definition. Although, certain bot-definitions do not contain the exact word "automated" or some variant of it, they still include an aspect of bots being non-human, or software controlled. Examining the definition of automated from the Cambridge Dictionary[5] the following definition can be found, "carried out by machines or computers without needing human control", suggesting that even though certain bot-definitions do not include the particular word automated, including a non-human or software control implies automatization.

With an established inclusion of automatization in the definition of a bot, a potential problem with this formulation can be highlighted. As already stated, the inclusion of the non-human or software control implies automatization, but likewise, automatization could be interpreted to imply software, this interpretation however creates a problem. The problem is that, at the moment, scientists in the bot-detection research field have no efficient or easy way to completely confirm that the accounts labeled as bots by bot-detection methods actually are software controlled. A scientist cannot physically visit the origin of a tweet to confirm that it was not produced by a human. Some cases of course do not require actual physical confirmation in this sense, since accounts can behave in ways impossible for human to act in, tweeting 3000 times in a minute for instance. Nonetheless, as bots try to mimic human behavior on

[4] Objects in this case do not only refer to physical objects, but includes nonphysical entities, such as computer programs.

[5] `https://dictionary.cambridge.org`

Twitter, theses obviously non-human behavior cannot always be found. Additionally, a human user could potentially act in an automated way, although not being a software or machine.

Given the above mentioned potential problem, the definition of bot will in this research follow the example of Johan Fernquist, Lisa Kaati and Ralph Schroeder in their paper Political Bots and the Swedish General Election, where bots are defined as accounts conveying an automated behavior, meaning bots do not necessary need to be controlled by software. The following definition of bot is from now on used:

- A bot is an account on a social online social network conveying an automated behavior.

Were Online social network (OSN) is defined as proposed by Boyd and Ellison, as a web-based services that that allow individuals to construct a public or semi-public profile within a bounded system, articulate a list of other users with whom they share a connection, and view and traverse their list of connections and those made by others within the system [14, p.211].

It should be noted that, the proposed definition of bots contains no requirement of bots being of a malicious intent. Included within the definition of bot is therefore also bots such as newspaper accounts on Twitter, which openly display being an automated software.

2.2 Categorizing the different types of bots

Much ambiguity has surrounded the term bot since its initial appearance in the early 1990s [39, p.1–3]. Depending on academic context and point in time, what a bot does and what a bot is, has been interpreted differently [39][88]. For instance, in the 2000s the network and information security research field started using the term sybil to describe fake accounts with malicious intent on social networks [81][85][80][103], whereas other research fields in computer science called these forged accounts bots [32].

To prevent confusion regarding the different interpretations of the what a bot is and what a bot does, a categorization of bots is provided in the following section. The categorization is mainly based on the paper *Unpacking the Social Media Bot: A Typology to Guide Research and Policy*, by Robert Gorwa and Douglas Guilbeault. Gorwa and Guilbeault divides bots in to six categories, Crawlers and Scrapers, Chatbots, Spam bots, Social Bots, Sockpuppets and Trolls, Cyborgs and Hybrid Accounts.

Although all of these categories of bots can have some kind of effect on OSNs, Chatbots, Crawlers and Scrapers are functionally different in their interaction with OSNs. Crawlers and Scrapers refer to programs that takes advantage of the Web graph structure to jump from page to page gathering information, while Chatbots are programs that are able to communicate with a person by analyzing text or speech input from a person and appropriately respond to the input [27][76]. Chatbots, Crawlers and Scraper do have an effect on OSNs, but for the purpose of this research they are not of as much importance and are therefore not elaborated on in the following section. The reasoning behind this choice is as follows, Crawlers and Scrapers do not directly interact with users on OSNs and are therefore removed, Chatbots do not on their

own seek out contact with users on OSNs and are therefore removed, although Chatbots in some cases can be part of tools used to create other types of bots. Given the pruning of the categories proposed by Gorwa and Guilbeault, the following categories remain.

2.2.1 Spam bots

Spam bot originates from the term spam which in the early days of internet, 1970-1990, referred to undesirable text or an excess of communication [65][15, p.22–23]. Over time the term evolved, associating more with certain types of activities like fake password requests, search engine manipulation, Nigerian prince scam and stock market manipulation [39, p.7]. Spam has become closely tied to the constant struggle between anti-spam projects and the economic incentive of "attackers" using spam [15, p.22–23].

In information security literature, spam bots have been traditionally referred to as nodes in a network that have been compromised by malware and can be controlled by a third party [70, p.15–20]. These spam bots are often used in big groups, botnets, which are used for malicious intents [70]. Spam bots are also used to impersonate real people on OSNs, where the spam bots try to gain the trust of legit users by creating profiles which look very similar to those of real people [89]. The spam bots then use the gained trust of people in the OSN to spread their content, such as links to malicious websites [89]. For the purpose of this research project, spam bots will be defined as automated account with the purpose of spreading spam to legit users in an OSN, both in groups and individually.

2.2.2 Social Bots

During the 2000s some of the biggest OSNs where founded, in 2003 Myspace emerged, 2004 was the year Facebook launched, in 2005 Reddit was created and roughly nine months later Twitter was founded. With these new OSNs came opportunities to deploy bots on new types of platforms [39, p.8]. Twitter in particular brought about a large increase of new automated accounts with their open application programming interface (API) [39, p.8]. These new automated accounts where recognized by scientists as a problem in early 2010s, as these "bots" where spreading large quantities of malicious content [23].

After the emergence of automated entities on OSNs, two different terms came to be to describe this phenomenon, social bots and socialbots, notice the difference in one term consisting of two separate words and the other term only one word [39, p.8]. In the information security research field, the term socialbot has mainly been used as a way to describe compromised nodes in a social network, where the compromised nodes often consist of computer programs mimicking real users [69, p.1][13, p.93]. Social bot has instead generally been used by researcher in the social sciences to describe computer algorithm that automatically produces content and interacts with humans on media, trying to emulate and possibly alter their behavior [34, p.96]. A major point of interest in recent research has been a particular subgroup of social bots, social bots used for political purposes, also called political bot [51, p.4][39, p.9]. Woolley and Howard defined in 2016 political bots as, algorithms that operates over social media, written to learn from and mimic real people so as to manipulate public opinion across

a diverse range of social media and device networks [51, p.4]. Following the example of [13, p.93][45, p.1–2], social bot will for the purpose of this research be defined as, an automated social media account which mimics a real user [39, p.10].

The categories social bots and spam bot may overlap with each other in the sense that they share attributes, spam bots could for instance impersonate real users on OSNs [39, p.7–8]. In some cases it is not even possible to clearly distinguish if the examined bots are spam bots or social bots [47]. Still, spam bots are different than social bots in the sense that the main purpose of a spam bots is to push out information and not to mimic human users.

2.2.3 Sockpuppets and Trolls

In general, the term sockpuppets is used to describe forged users interacting with real users on OSNs, while interacting with real users sockpuppets are used to perform a range of activities [10, p.39][16, p.366][44]. These activities include, creating an illusion of support for certain opinions, promoting certain peoples work, spreading misinformation, disputing individuals and communities [10, p. 39]. Sockpuppets with a political agenda are usually labeled as trolls [39, p.10]. Following Gorwa and Guilbeault, 2018, sockpuppets will be defined as accounts with manual curation and control [39, p.10].

2.2.4 Cyborg and Hybrid accounts

In terms of functioning, disregarding the active context, a cyborg or hybrid account is the combination of a social bot and a sockpuppet. A cyborg is a bot-assisted human or a human-assisted bot, the crossover of a bot and a human [23, p.21]. Yet, research in the field of bots still lacks a clear definition of how much automation is required for a human account to be defined as a cyborg, vice versa [39, p.11]. Tools such as Tweetdeck[6] has enabled legit users to perform automated behavior such as scheduling tweets, managing several twitter accounts at once, making it even more difficult to distinguish between cyborgs and normal users, normal users in this case refer to accounts used as indented by Twitter's policy. Because of the lack of a clear definition of when a human is to be considered a cyborg, cyborg will now forward be defined as an account which convey both the traits of a human and a software.

2.3 A categorization of bot detection methods on OSNs

In the pursuit of finding bots on OSNs many different methods have been developed. Methods range from complex machine learning algorithms to simple thresholds for certain types of activities, an example of a threshold is labeling accounts tweeting 50 or more times per day as bots [57]. One way of categorizing these different types methods is to divide them into either inferential approaches or descriptive approaches. The seperation was proposed by Christian Grimme, Dennis Assenmacher and Lena Adam in their paper *Changing Perspective: Is It Sufficient to Detect Social Bots?*.

[6] `https://tweetdeck.twitter.com`

In a broad sense, inferential methods refer to methods based on the assumption that bots share common characteristics in behavior which can be utilized to create a fixed set of rules for finding bots [40, p.447–448]. This rule set does not necessarily need to be simple statements but could also be complicated machine learning models for bot classification. Using labeled data, data with accounts labeled as bot or non-bot, researchers establish behavioral features for bots which are used in the rule set for detecting bots [40, p.447–448]). An example of an inferential approach is bot detection using a deep neural network, where the behavioral features are an input vector to the neural network and the rules are the trained neural network model [63].

In contrast to inferential approaches, descriptive approaches detect bots through examination of individual campaigns on OSNs, analyzing data from the campaigns to find patterns [40, p.448]. The analysis often involves some type of clustering or frequency indicator to compare a number of different accounts [40, p.447–448]. Descriptive approaches utilize the human intelligence, since the approach require researchers to select indicators to examine the data. Additionally, the result from the analysis has to be interpreted by a human, since a descriptive approach does not have labeled data to rely on [40, p.447–448]. An example of a descriptive approach is the discovery of the Bursty Botnet [29]. The botnet was discovered through the examination of a spike in creation of Twitter accounts in February and March 2012 [29, p.4]. When examining accounts created in February and March 2012 researchers found a number of bots exhibiting the same type of traits, the accounts had generated at least three tweets within the first hour after creation and then stopped tweeting, the accounts only tweeted from a source of "Mobile Web" and the content of the tweets consisted of a URL or/and a mention of another user [29, p.4].

Although the proposed categories, inferential approaches and descriptive approaches, give a broad understanding of the research field, a more in-depth categorization of methods to detect bots on OSNs can be found in the paper, The art of social bots: a review and a refined taxonomy by Majd Latah [59]. To display more in-depth picture of the methods currently used to detect bots, the following section is dedicated to a taxonomy based on the paper by Latah, the taxonomy is supplemented by material from the categories proposed by Ferrara et al., in their paper *The Rise of Social Bots* [34].

2.3.1 Graph-based methods

Graph-based methods leverage the features of graphs created with accounts on OSNs to detect bots, these graphs are also called social graphs [34, p.100], [20, p.5]. The social graph methods are built upon the assumption that graphs of bots in an OSN have different properties than graphs of honest users in an OSN [6, p.383]. By utilizing these differences in the graphs of bots and humans, bots can be successfully detected [6, p.383]. It should be noted that the social graphs of bots and humans often include a mix of connections between bots and legitimate accounts [6, p.395]. Furthermore, both the supervised machine learning approaches and the unsupervised machine learning approaches, see Section 2.3.2 and Section 2.3.3, can also utilize the properties of social graphs to detect bots. Graph-based methods however differentiate from these types of methods in the sense that their main focus is

using properties of social graphs, unsupervised and supervised machine learning approaches are instead broader categories including a bigger span of different bot detection methods. An example of a feature used in social graph-based detection is the longest distance between two nodes in a graph [6, p.383]. Methods utilizing these differences in properties of graph can be divided in to six groups, the following six sections represent these groups.

2.3.1.1 Random-walk-based approaches

Based on the pattern of random walks performed on an OSN graph, random-walk-based methods label accounts as honest users or bots. Attributes of the walk patterns used are for example, which nodes have been crossed by a walk, how many times a node have been crossed by one or several walks, how a particular random walk compares to the mean and standard deviation of a walks [59, p.8–10]. SybilGuard for instance labels accounts as honest if random walks from honest nodes intersect with random walks from the nodes being evaluated. The method assumes that a creating a connection to an honest node requires establishing a certain amount of human established trust with the honest node, which in turn limits the number of connections a malicious user can established to honest nodes since establishing trust is considered difficult [104, p.578].

2.3.1.2 Community detection approaches

Community detection approaches make the assumption that social graphs can be divided into different types of communities, the first community consisting of tightly connected honest users and the second community consisting of tightly connected bots [104, p.586]. Identifying these communities with either honest users or bots enables the methods to distinguish between bots and honest users [104, p.586–587]. However, not all community identifying concepts are successful in the process of detecting bots.

For instance, maximizing the modularity to identify communities [72] was determined not to work for bot detection in an experiment [36, p.7]. Modularity being defined as the fraction of edges between within-community nodes, minus the expected quantity of edges between nodes if the same quantities and communities of a network is applied but the edges are connected randomly between nodes [72, p.8]. The reason behind modularity's inefficiency as a tool were determined to stem from the fact that roughly half of the bots were isolated, meaning they were only connected to honest nodes, and honest nodes and the bots had a lot of connections between each other [36, p.7]. Other concepts, such as the conductance of a graph has also been used in community detection [68]. Researchers have identified communities of bots by minimizing the conductance of sets of nodes, where conductance is defined as a measure of the intensity of connections between a set of nodes and the rest of a graph [90, p.482]

2.3.1.3 Weighted trust propagation-based approaches

Weighted trust propagation-based approaches (WTPBA) utilize approaches similar to the famous algorithm PageRank by Google. PageRank is an algorithm providing a hierarchal structure for websites, where the authority, a of measure of importance [11, p.93], of a website is based upon the number of incoming links and the authority of the website providing the links [11, p.94]. The more incoming links and the higher authority of the website providing

the links, the higher authority of the website [11, p.94]. Generally, the idea of PageRank algorithm is applied to bot detection as follows, the equivalent of websites are users or nodes in OSNs, links from one website to another website are edges between users/nodes in the graph, authority is trust attached to the edges between nodes/users and/or nodes/users in themselves [59, p.11]. Trust, in the sense of which nodes that cannot be trusted as legitimate users, is then propagated through the graph. The propagation starts with a seed node, the trust of the nodes linked to the seed node is then updated based through the edges from the seed, the procedure is repeated but with the newly updated nodes having the same roles as the seed node initially. The initial node can both be of unknown character, in the sense that it is not clear if it is a bot or not, or an account already labeled as bot or not [59, p.11–12].

SybilFence is an example of a weighted trust propagation-based approach. SybilFence utilizes the observation that fake users often receive a significant share of negative feedback from legitimate users, negative feedback being friend request being ignored for instance [17, p.1]. The method penalizes edges of users which have received negative feedback [17, p.1–2], so that when trust is propagated through the graph, trust propagates through the penalized edges to a lesser degree [17, p.5].

2.3.1.4 Loopy belief propagation-based approaches

Similarly, to WTPBA, loopy belief propagation-based approaches (LBPBA) find bots by propagating trust through a graph. However, LBPBA use a small set of known bots and honest users to create a semi-supervised learning problem [59, p.11–12]. The semi-supervised learning problem consists of propagating trust through the social connections from the pre-labeled nodes to the rest of the nodes in the social graph. SybilBelief is an example of a LBPBA, the method models the social network between nodes as pairwise Markow Random Fields, where a Markow Random Field defines a joint probability distribution for a binary variable attached to each node, where the binary variable represents bot or honest users [37, p.976]. With the pre-labeled data, the posterior probability of nodes being an honest user is then inferred, which in term is defined as the trust of the node [37, p. 976].

2.3.1.5 Combinations of graph-based approaches and machine learning aided graph-based approaches

Not all methods utilize only one category of graph-based methods to detect bots, instead certain methods combine different types of graph-based approaches. One way of creating such a combined method is to have a clear step by step procedure. Each step in the method corresponds to an element taken from a category of one of the graph-based approaches [59, p.12–13]. An example of such a method is SybilRadar, the method first calculates similarities between nodes using the Adam-Adar metric and Within-Inter-Community metric [71, p.183]. These similarities are then applied as weights to the edges in the social graph [71, p.183]. Lastly, Modified Short Random Walks are run on the graph to produce trust values for each node, were the trust values corresponds to the landing probabilities for random walks [71, p.183].

Other methods utilize elements from both graph-based approaches and machine learning

approaches [59, p.12]. One of these machine learning aided approaches is Integro. Using content-based features of users, such as number of followers, Inegro trains a random forest model to identify potential victims of bot attacks, the machine learning model is then used to identify potential victims which used to supplement the social graph [12, p.4–6]. The social graph is extended by initiating weights of edges based on their vertices adjacency to potential victims [12, p.4–6]. Finally, a modified random walk is performed to determine the trust of the nodes [12, p.6–7].

2.3.2 Supervised machine learning approaches

Utilizing the behavioral patterns of humans and bots on OSNs, behavioral features of accounts on OSN can be extracted which in turn can be used in the training of machine learning models [34, p.101]. Using the behavioral features of accounts on OSNs the machine learning models can identify differences in feature signatures of bots and humans, enabling them to successfully classify accounts as bots or not bots [34, p.101]. Supervised machine learning approaches rely on labeled data to train the models on, data were accounts haven been labeled as bots or not bots.

An example of a commonly used supervised machine learning algorithms is Random Forest [62, p.312] [30, p.125]. Typically meta behavioral features are used in these type of methods , tweets per day, length of username, likes given, for instance [30, p.126] [34, p.102]. Random Forest can also use behavioral features tied to the content of tweets, such as unique hashtags per tweet, length of tweets etc. to detect bots [30, p.126]. Certain methods have even utilized Random Forest to classify tweets as bots created or human created, utilizing the features both from tweets and the accounts tied to the tweets [62]. Other examples of supervised machine learning methods are naïve Bayes classifier and Logistic Regression [84, p.169–171].

2.3.3 Unsupervised machine learning approaches

Similar to supervised machine learning approaches, unsupervised machine learning approaches focus utilizing various features of accounts to label accounts as bot or not bots [59, p.14–15]. Unsupervised approaches though, tries to find underlying patterns in the data without levering labeled data [59, p.15]. In other words, unsupervised approaches do not use prelabeled data of bots and non-bots. A common example of unsupervised machine learning type is clustering, much like descriptive approaches, these methods rather tries to find bot campaigns instead of individual bots [59, p.15].

Unsupervised machine learning approaches can utilize a combination of tweet content and meta behavior features together to detect bots, one example of such a method is DNA-inspired behavior modelling. DNA-inspired modelling encodes the meta behavior and content of tweets from individual accounts into string with letters [24, p.565–567]. These DNA-inspired strings of letters are then compared with DNA-inspired strings of other accounts, in the comparison accounts are group based on their similarity in strings [24, p.567–568]. Groups or accounts with a very high degree of similarity is considered to behave in a non-human way,

indicating that they could be bots [24, p.567–568].

2.3.4 Crowdsourcing

In general crowdsourcing refers to the process of outsourcing work to an unidentified group of people [98, p.2]. In the case of crowdsourcing of bot detection people are shown information from accounts on OSNs, such as photo albums, wall, profile information, based on this information they are then asked to classify the account as bot or human [98, p.5]. Additional groups apart from bot or human could of course also be included in the labeling, such as cyborg for instance [23, p.23]. Some studies use additional process to ensure the best possible result, one example being a majority vote among people classifying accounts, to establish a final more certain classification [98, p.10][5, p.334]. In this case, several different people classify the same account, a final classification is then made based on which label is most prevalent among the previous classifications.

2.4 Random Forest classification

This section is dedicated to the machine learning algorithm Random Forest, which would in the context of bot detection on OSNs be considered as a supervised machine learning approach, see Section 2.3.2. The choice to dedicate an own section to Random Forest classification was however made, since Random Forest classification had such a major role in the experiment performed in this study, see Section 4.

Random Forest is a machine learning method used for classification and regression analysis. In the context of this research, Random Forest classifiers are used to map Twitter accounts to the class bot, or the class not-bot. Random Forest classifiers are based on classification trees, functions which map data to pre-determined classes [64, Chapter 12]. A classification tree uses recursive binary splitting for model creation, a method which selects an input variable from the data together with a cut-off point for the input variable, the algorithm then splits the data in to two data sets based on the cut-off point and input variable, the procedure is repeated until a predetermined stop criteria is met [9, p.15]. The stop criteria could for example be to repeat the splitting procedure until every individual data set contains no more than three data points [64, Chapter 12]. An example of a classification tree is displayed in Figure 1. In a decision tree, the end of the branches are called leafs [64, Chapter 12], in Figure 1 the leafs are represented by the boxes with *bot*, or *not bot* as labels.

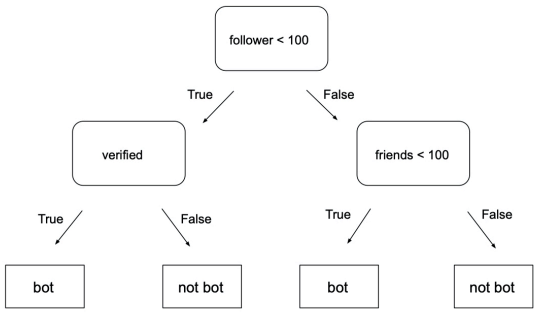

Figure 1: an example of a simple classification tree.

As displayed in Figure 1, each path down in the tree ends in a class prediction. Figure 1 is a very simple example of a classification tree, only classifying data points in to two categories using two layers. When using recursive binary splitting to create a decision tree, each split is based on error minimization, were the input variable and cut-off point is chosen to minimize the error, in other words splitting the data in such a way that as many data points as possible are classified correctly [64, Chapter 12]. Calculating the best possible split can be done by minimizing several different types of errors, common errors to minimize are misclassification error, entropy or Gini [64, Chapter 12]. Recursive binary splitting is a greedy approach, meaning that the splits are each done to minimize the error without considering future splits [9, p.15]. A finished classification tree can be used to classify new data points without labels, each new data point then travels down the paths in the decision tree depending on their set values in relation to the splits in the decision tree, until reaching the bottom and a classification label [100, Chapter 4].

When creating a model, the data is often divided in to training data and test data [100, Chapter 5]. The training data is used to train the model, recursive binary splitting in this case, and the test data is used to evaluate the model performance [100, Chapter 5]. With the two separated data sets one can identify certain traits belonging to classification trees, traits which give rise to a trade-off between low bias and low variance when creating models [64, Chapter 13]. A deep tree, meaning a tree with many layers, usually gives the model a small bias, meaning that the model has enough flexibility to describe the underlying relationships in the data, in other words the model describes the training data very well [100, Chapter 12]. A deep tree however, usually has a high variance, meaning that the model does not perform well when classifying the test data, the model is over-fitted to the training data because of its high flexibility [100, Chapter 12]. Hence, the trade-off becomes choosing between a deep tree with low bias and high variance and a shallow tree with high bias and low variance [64, Chapter 12].

A way of counteracting the problem of bias-variance trade-off is to use a Random Forest classifier instead [64, Chapter 13]). A random forest classifier creates several decision trees, were each tree is trained on its own data set sampled from the original data set [64, Chapter 13]. By using the combined result of all classification trees, the variance can be reduced while still keeping the bias low [64, Chapter 13]. The combined classification of the Random Forest

model is determined by some type of voting among the individual trees in the Random Forest, weighted vote or majority vote are some examples [100, Chapter 12]. Every tree in a Random Forest is trained on its own data sets, sampled from the training set [64, Chapter 13]. Since the training set is limited and a Random Forest requires several trees, a shortage of training data can arise (slide 18). To counteract the shortage of data, a method called bagging is used [64, Chapter 13]. Bagging utilizes the concept of sample with replacement, meaning that when sampling training data sets for individual trees in the Random Forest, the same data point can be used several times, both in the same data set and in different data sets [64, Chapter 13]. Bagging creates another problem however, the data sets used for training of trees in the Random Forest becomes correlated, which diminishes the variance reduction [64, Chapter 13]. To address the problem of correlation a restriction is applied to the splitting in the classification tree training. In each split in each tree, only a random subset of variables is considered, which decorrelates the trees [64, Chapter 13].

2.5 Criticism of social bot research

The second of November, 2019, the Openfest Conference was completed in Sofia, Bulgaria. During the conference the German journalist Michael Kreil held talk, *The Army that Never Existed: The Failure of Social Bot Research*. The talk criticized the current research field of social bots. A summary of the talk can be found on Michael Kreil's Github [58] and a video of the talk on the OpenFest Bulgaria's YouTube channel [7]. Kreil's criticism is divided in to three parts, each part dedicated to a certain research team active in the research field. Kreil derived these research teams from references in news articles written about the subject of social bots. Exploring these references, Kreil distinguished three main teams mentioned:

- *The Computational Propaganda Project* of of Oxford University.
- University of Southern California and Indian University (SC/I).
- University of California, Berkley and Swansea University (CBS).

Investigating the work of these teams Kreil claims to have found serious flaws in their research. Starting his criticism, Kreil examines four papers published by Oxford University [57][50][48][49], Kreil states that the method used for detecting bots in the papers, by picking accounts tweeting at least 50 times per day, is not scientifically tested and based on a pattern easily achieved by humans and not only bots. To strengthen his point, Kreil gathers 300 000 verified [8] Twitter profiles with associated tweets and classifies them according to the threshold of 50 tweets per day, obtaining 1.46% of the profiles being bots. Verified in this case referring to accounts that has been reviewed by Twitter and confirmed to be authentic. Kreil then states that the percentage of bots among the accounts in the study of the US election [50] should be higher than the percentage of bots among the verified accounts, since the verified accounts have been examined by Twitter. However, the percentage of bots is higher among the verified accounts, 1.46% of verified accounts labeled as bots and 0.11% of the accounts in the US

[7] https://www.youtube.com/watch?v=vyTmczjwFRE&t=1667s

[8] https://help.twitter.com/en/managing-your-account/about-twitter-verified-accounts

election study labeled as bots, [50, p.4][58], which Kreil sees as an indication of the method being defected.

Continuing, Kreil examines two papers by University of Southern California and Indiana University which uses machine learning algorithms to detect bots [26] [99]. To train the machine learning models for bot classification, the SC/I team uses labeled data taken from a honey pot study [99, p.2] [60] Kreil argues that the choice of labeled data for the machine learning models is unsuitable for the task of detecting social bots. The criticism is based on the fact that the honey pot study defines their targeted bots as spammers, malware disseminators and content polluters [60, p.1]. These types of bots are according to Kreil not social bots, which in turn would lead to the SC/I team training models to find spammers, malware disseminators and content polluters and not social bots.

Proceeding in the criticism of the two papers published by the SC/I team, Kreil evaluates the Twitter bot detection framework created by the authors, the Botometer [26]. Using the Botometer framework to classify groups of known bots and humans, Kreil receives results showing a high number of misclassified Twitter accounts, some examples are:

- 10.5% of NASA-related accounts are misclassified as bots.
- 12% of Nobel Prize Laureates are misclassified as bots.
- 21.9% of staff members of UN Women are misclassified as bots.
- 36% of known bots by New Scientist are misclassified as humans.

Obtaining these results, with a significant number of misclassified accounts, Kreil deems the Botometer to be an unfit tool to be used in science and that papers using the tool [56] should be revoked. Concluding the criticism Kreil examines the work from University of California, Berkeley and Swansea University. Trying to reproduce the work of CBS, Kreil states to have requested the source code of the work but received nothing in return. Further, Kreil states to also have requested the source code for the Botometer framework but received nothing in return. Unable to easily reproduce the work of the CBS team, Kreil instead reviews the result of the study [38]. In the study Kreil finds claims of social bots having sihifted the outcome of the 2016 EU referendum by 1.76 percentage to "leave" and the 2016 US Presidential election by 3.23 percentage to endorse Trump [38, p.19–20]. Further examining the results Kreil concludes the claims are based on a calculated correlation between the number of tweets with certain hashtags and the result of the US election and the UK election [38, p.10, p.30]. Kreil however, rejects the correlation between number of tweets and election outcome as a base for calculating shifts in percentage of voters, since according to Kreil correlation does not imply causation.

Finally, Kreil also examines research not included in the work of Oxford, CBS and SCI/I. In this additional criticism Kreil examines claims from Professor Sasha Talavera at University of Birmingham. Professor Talvera claims to have detected bots by using the criterion: bots are 'users with exactly 8 digits in usernames', which according to Kreil is not a proper criterion

for detecting bots, since adding 8 digits to names is the standard naming scheme when joining Twitter. Kreil states that, finding 8 digits in a Twitter name is only a sign of the user having accepted the standard naming scheme of Twitter. Apart from the critique formulated by Kreil, researcher David Karpf at Georgia Washington University also presents a criticism of certain aspects in the research field of social bots. The criticism is presented in the form of the article *On Digital Disinformation and Democratic Myths*[55]. Slightly different from the work of Kreil's, Karpf's critique is more targeted at the conclusions drawn in the research field of bots on OSNs, rather than the methods used. To obtain a broad view of the criticism presented by Karpf, the following quote from the critique can be observed:

- "Generating social media interactions is easy; mobilizing activists and persuading voters is hard" [55].

Karpf tries to highlight the necessity of drawing a line between bot activity on OSN and political influence. The criticism presented by Karpf is not targeted at the methods to detect bots, or the results displaying a high presence of bots on OSNs. Rather, Karpf puts emphasis on the need to evaluate if the actions of bots on OSNs actually has an impact on people's actions and beliefs outside of OSNs. As Karpf explains it:

- "Political persuasion is systematically different from other forms of marketing and propaganda " [55].

According to Karpf, one cannot take for granted that political persuasions over OSNs works the same way as influencing someone to buy a certain soft drink for instance, Karpf states that political persuasion is extremely hard. To prove the difficulty of political persuasions Karpf references a meta-analysis by Joshua Kalla and David Broockman [53], the study examines recent American elections finding the following result:

- "We argue that the best estimate of the effects of campaign contact and advertising on Americans' candidates choices in general elections is zero" [53].

To strengthen his point, Karpf also presents an example of how the Russian Internet Research Agency (IRA) supposedly tried to polarize the political discussion in the USA. IRA liked, shared and commented on certain Facebook events and Facebook accounts tied to the Black Lives Matter, attracting extra attention to these particular events and accounts. Continuing Karpf states that, barely anyone actually physically showed up to the IRA promoted events, although the event received significant attention on Facebook. Karpf sees this as an example of gaps in the correlation between actions on OSNs and actions in the physical world.

3 Data

Two different types of data was used for the purpose of this research. The first type of data consisted of Twitter accounts discussing Swedish politics, referenced to as Swedish political data. The second type of data consisted of Twitter accounts labeled as bot or not bot, these Twitter accounts were used for training and testing of machine learning models. To easily separate between the unlabeled and labeled data, two separate section were created. Section 3.1 was dedicated to describing the unlabeled Swedish political data and Section 3.2 was dedicate to describing the labeled data for training and testing of random forest models.

3.1 Swedish political data

Three Twitter data sets with a Swedish political context were collected. Data set 1 consisted of Twitter accounts using common Swedish political hashtags. Data set 2 consisted of Twitter accounts tweeting about the Swedish political event Almedalen. Data set 3 consisted of Twitter accounts belonging to Swedish politicians. After the collection of Twitter accounts, the most recent 200 statuses, tweets and retweets, of each account was also gathered using the Twitter streaming API.

3.1.1 Data set 1

Using the developer tools provided by Twitter and the Python library Tweepy[9][10], 20 000 tweets where gathered within the time span of 2019-11-29 to 2020-02-01. To obtain Tweets discussing Swedish politics the Twitter Streaming API was used, filtering tweets by common Swedish political hashtags, the following hashtags were used *#migpol, #svpol* and *#säkpol.* The authors and retweeters of the gathered tweets were then compiled into a list of 976 accounts, which were used as data set 1.

3.1.2 Data set 2

Using a data set of tweets collected by Infolab at Uppsala University, accounts retweeting and tweeting were compiled in to a list consisting of 1189 accounts. Infolab gathered the tweets using the Twitter streaming API, filtering by the keyword *almedalen* during the period of 2018-07-01 to 2018-07-08. By filtering by the keyword *almedalen*, Infolab aimed to collect tweets associated with the annual political event Almedalsveckan in Sweden.

3.1.3 Data set 3

Using a data set of Twitter accounts belonging to Swedish politicians, collected by Anton Norberg in [73], 238 accounts were compiled into a list. The Twitter accounts gather by Norberg consisted of profiles belonging to Swedish ministers and commissioner. Norberg gathered the profiles by first fetching names of Swedish ministers and commissioners from the Swedish Parliament website [11]. Using the google search engine, the fetched names were

[9] https://developer.twitter.com

[10] https://www.tweepy.org

[11] https://riksdagen.se/sv/ledamoter-partier/

then individually applied as search terms to the search engine together with the word *twitter*. Lastly, the result from the search engine was manually examined by Norberg to find authentic Twitter accounts belonging to Swedish ministers and commissioners.

3.1.4 Evaluation of Swedish political data

The process of gathering data for both data set 1 and data set 2 included a filtering mechanism, filtering tweets by keywords or hashtags. By filtering the gathered data, the aim was to gather politically associated data. To be noted is that, filtering data by certain politically associated hashtags or words does not guarantee that the actual content of the data is part of political discussions. Trending political hashtags could potentially be used for other purposes, spreading of spam for instance, since using trending hashtags enables tweeters to reach a large number of people.

Still, analyzing all tweets using a certain political hashtag remain relevant for this research: since all tweets using a political hashtag, discussing politics or not, affect the hashtags potential to work as a tool for political discussion. For instance, spam bots could use a political hashtag to spread advertisement for soda, although the soda advertisement is not part of the political discussion, constantly bombarding the hashtag with soda advertisement still affect the users when they try to use the hashtags for political discussion. Additionally, using filtering by politically associated words or hashtags does not guarantee that the tweets represent an even distribution of political views. Certain hashtags or words could mainly be used by groups of people with a specific political view. Still, not including all political views in the data does pose a problem, since the purpose of this research is to evaluate the bot detection methods and not the full spectrum of political discussions carried out over Twitter.

The accounts in data set 1 and data set 2 were collected by gathering authors of tweets and retweets. Only gathering authors of tweets and retweets could potentially oversee groups of accounts on Twitter, groups of accounts which are mainly active on Twitter in other ways than tweeting and retweeting. For instance, groups of accounts which only like and comment on tweets would not be included in the Swedish political data. The reason why only authors of tweets and retweets were gathered is that the standard Twitter API, Section 3.3, did not provided possible API requests to fetch accounts commenting or liking on tweets.

3.2 Labeled data for training and testing of random forest models

Three labeled data sets were used to train random forest models used for bot detection. The data sets consisted of Twitter accounts labeled as bot or human. Each labeled data set was used separately to train one random forest model each, data set 1 used to train random forest model 1, etc. All of the labeled data was retrieved from the website `https://botometer.iuni.iu.edu/bot-repository/datasets.html`. After retrieving the accounts from the website, the most recent 200 statuses, tweets and retweets, of each account was also gathered using the Twitter streaming API.

3.2.1 Labeled data set 1

Labeled data set 1 consisted of profiles manually labeled as bot or human by Yang Kai-Cheng. The accounts were labeled in the process of making the paper [101], Yang Kai-Cheng having been one of the authors to the paper.In total 493 accounts were used, of which 211 were labeled as bot and 282 as human.

3.2.2 Labeled data set 2

Labeled data set 2 consisted of 368 accounts, 75 of which were annotated as bot and 293 of which were annotated as human. The annotated accounts were provided by authors from two different papers, [67] [102]. The accounts from [67] consisted of accounts manually labeled bot or human, accounts from [102] consisted of self-identified bots from `https://Botwiki.org`.

3.2.3 Labeled data set 3

The accounts in the labeled data 3 consisted of purchased fake followers from the paper [101] and verified humans from the paper [102]. Verified accounts referred to accounts verified by Twitter itself. Combining the fake followers with the verified human accounts, 664 accounts were used for labeled data set 3. 285 of the accounts were labeled as bot and 379 of the accounts as human.

3.3 Twitter API

To obtain Twitter account information, tweet information and retweet information the Twitter API was used. The Twitter API was divided in to three different types[12], standard, premium enterprise. For the purpose of this research, the free standard API was used. The Twitter API allowed for sending of request for data which returned data in JSON format. Figure 2 displays an example of Twitter account information received from the Twitter API, to be noted is that the personal information in Figure 2 has been replaced.

[12] `https://developer.twitter.com/en/products/products-overview`

```
{
  "id": 42,
  "id_str": "42",
  "name": "Test",
  "screen_name": "the_test",
  "location": "Uppsala",
  "profile_location": null,
  "description": "I am a test",
  "url": "https:test.xcom",
  "entities": {},
  "protected": false,
  "followers_count": 42,
  "friends_count": 42,
  "listed_count": 1337,
  "created_at": "Tue May 23 06:00:00 +0000 2013",
  "favourites_count": 31,
  "utc_offset": null,
  "time_zone": null,
  "geo_enabled": null,
  "verified": true,
  "statuses_count": 1337,
  "lang": null,
  "contributors_enabled": null,
  "is_translator": null,
  "is_translation_enabled": null,
  "profile_background_color": null,
  "profile_background_image_url": null,
  "profile_background_image_url_https": null,
  "profile_background_tile": null,
  "profile_image_url": null,
  "profile_image_url_https": "https:test.com",
  "profile_banner_url": null,
  "profile_link_color": null,
  "profile_sidebar_border_color": null,
  "profile_sidebar_fill_color": null,
  "profile_text_color": null,
  "profile_use_background_image": null,
  "has_extended_profile": null,
  "default_profile": false,
  "default_profile_image": false,
  "following": null,
  "follow_request_sent": null,
  "notifications": null,
  "translator_type": null
}
```


Figure 2: Example of Twitter API response for account information request.

4 Method

Three types of bot detection methods were tested and used to evaluate the Swedish political data, these are described in Section 4.2, Section 4.3 and 4.4. To distinguish between the bot detection methods used to evaluate the Swedish political data and the bot detection methods described in Section 2.3, the bot detection methods used to evaluate the Swedish political data will henceforth be referenced to as the test methods.

4.1 Choice of bot detection methods

In previous studies random forest has been proven to yield the best result in Twitter bot detection when compared to other machine learning algorithms [60, p.190][86, p.15][99, p.3][79, p.819]. The Random Forest algorithm has also successfully been used to detect Twitter spam [41] and in recent studies to detect bots and bot created material in Swedish Twitter data [30][62]. Given the recent successful studies analyzing Swedish Twitter data, combined with previous research indicating the excellence of Random Forest in the process of bot detection, the choice was made to use Random Forest as a test method for bot detection.

To evaluate the criticism proposed by Michael Kreil, see section 2.5, the choice was also made to use the Botometer framework and a criterion proposed by [57][50][48][49] as test methods for bot detection. Both the Botometer framework and the criterion proposed by [57][50][48][49] are included in Kreil's criticism, these methods were therefore chosen as test methods to evaluate Kreil's criticism. The criterion proposed by [57][50][48][49] is stated as follows:

- ”We define a high level of automation as accounts that post at least 50 times a day using one of these election related hashtags, meaning 450 or more tweets on at least one of these hashtags during the data collection period” [57, p.3][50, p.3][48, p.3][49, p.3].

Were highly automated accounts are considered as bots, as can be seen in the following quote from the same research:

- ”A fairly consistent proportion of the traffic on these hashtags was generated by highly automated accounts. These accounts are often bots that are either irregularly curated by people or actively maintained by people who employ scheduling algorithms and other applications for automating social media communication” [57, p.3][50, p.3][48, p.3][49, p.3].

To be noted is that the accounts in [57, p. 3][50][48][49] are examined based on how many times they have tweeted with certain hashtags related to a political event. Examining accounts with the same criterion based on their tweeting behaviour without regards to hashtags usage is therefore not equivalent to the research in [57, p.3][50, p.3][48, p.3][49, p.3]. However, examining accounts based on their tweeting behaviour without regards to hashtags could still be an indication of accounts conveying a high level of automation, which is the equivalent to being a bot following the suggestion by [57][50][48][49] and in line with the definition of bot, see section 2.1 for bot definition. Finally, Botometer has been described as a very well known tool in the research field of bot Twitter bot detection [102, p. 2][40, p.445][101, p.48], which further highlights the importance of testing Botometer.

4.2 Random forest models

Three random forest classification models were created as test methods, each model using one respective labeled data set for training. Each random forest model used the labeled data set marked with the same number, random forest model 1 using labeled data set 1 for training, etc. Before training the random forest models, the most recent 200 statuses[13], tweets and retweets, of every account in the Swedish political data and the labeled data sets were gathered. Additionally, accounts specific information[14] from the accounts in the labeled data and Swedish political data was also gathered. Gathering the statuses and account specific information was done using the Twitter API and the Tweepy Python library.

The statuses and account specific information was then preprocessed to obtain feature specific information, such as account age, likes per follower, Table 1 and Table 2. Preprocessing included calculating statistics regarding the status content of the accounts, the statistics were calculated using the the Python library Statistics[15]. To enable testing of the random forest models, each labeled data set was divided in to one training data set and one test data set, the split was done 75% training data and 25% test data.

4.2.1 Feature selection

The features used in the random forest models were decided based on the features used in the paper [30, p.126], the same paper which received significant attention in Swedish news media, see Section 1. Certain features used in [30, p.126] were excluded because of time and resource limitation, most of the excluded features were related to tweet and retweet content. Although features requiring language specific analysis have been successfully utilized in previous bot detection methods [28] [77], these features were excluded because of potential inconsistencies arising from using two different analyzes, one for the Swedish Political data and on for the English labeled data. Given the features used in the random forest model in [30, p.126], the features displayed in Table 1 and Table 2 were chosen as the features for the random forest models.

[13] https://developer.twitter.com/en/docs/tweets/data-dictionary/overview/tweet-object

[14] https://developer.twitter.com/en/docs/tweets/data-dictionary/overview/user-object

[15] https://docs.python.org/3/library/statistics.html

Table 1: Random forest model features.

Feature name	**Feature description**
Account age	The age of the account in hours
Verified	Boolean, True if the account has been verified by Twitter, otherwise False
Followers count	The number of followers of the account
Friends count	Number of accounts which the account follows
Follower friend ratio	Followers count divided by Friends count
Likes count	Number of tweets liked by the account
Likes per follower	Likes count divided by Followers count
Likes per friend	Like count divided by Friends count
Likes age	Likes count divided by Account age
Length username	Length of the accounts username
Location	Boolean, True if the account has provided a location, otherwise False
Default profile image	Boolean, True if the account uses the default profile image provided by Twitter, otherwise False
Statuses count	Number of times the account has tweeted, retweets included
Hashtags per tweet	Number of extracted hashtags in tweets*, divided by the number of tweets
URLs per tweet	Number of extracted URLs in tweets*, divided by the number of tweets
Mentions per tweet	Number of extracted mentions in tweets*, divided by the number of tweets
Retweet tweet ratio	Number of tweets divided by number of retweets, were tweets and retweets belong to the most recent 200 statuses
Unique mentions per tweet	Number of extracted unique mentions in tweets*, divided by the number of tweets, mentions referring to mentions of other Twitter accounts in tweets
Unique hashtags per tweet	Number of unique extracted hashtags in tweets*, divided by the number of tweets
Unique URLs per tweet	Number of extracted unique URLs in tweets*, divided by the number of tweets

* Tweets referring to the tweets belonging to the most recent 200 statuses of the account.

Table 2: Random forest model features associated with statistic of statuses.

Feature name	**Feature description**
Symbols per tweet	An array of features containing statistics** regarding the number of words in tweets*
Symbols per retweet	An array of features containing statistics** regarding the number of symbols in retweets***
Words per tweet	An array of features containing statistics** regarding the number of words in tweets*
Words per retweet	An array of features containing statistics** regarding the number of words in retweets***
Time between tweets	An array of features containing statistics** regarding the time between tweets*
Time between retweets	An array of features containing statistics** regarding the time between retweets***

* Tweets referring to the tweets belonging to the most recent 200 statuses of the account.
** The array of statistics include, mean (average), median, min, max, variance, standard deviation.
*** Retweets referring to the retweets belonging to the most recent 200 statuses of the account.

All three random forest models used all of the features displayed in Table 1 and Table 2, with the exception of random forest model 3 not using the Verified feature. The choice to exclude the Verified feature in random forest model 3 originated from the type of labeled data used to train random forest model 3. The labeled data used to train random forest model 3 consisted of verified human accounts and self-identified bots. Since all humans were verified the model simply distinguished between bots and humans by examining if they were verified or not, verified accounts being classified as humans and not verified accounts being classified as bots. Classifying accounts by only examining one feature made the random forest algorithm unnecessary and created a model with a very high variance. An example of a type of account which was constantly missclassified was the very common not verified human account, missclassified as bot since it was not verified.

4.2.2 Hyperparameter tuning

To obtained the best random forest models with the available labeled data, a hyperparameter tuning was performed. The hyperparameters were tuned for the random forest models are displayed in Table 3.

Table 3: Hyperparameters tuned for random forest models.

Hyperparameter	**Hyperparameter description**
Max depth of the trees	Maximal number of layers allowed in the classification trees in the forest, a restriction to the depth of the classification trees
Number of trees in forest	Number of classification trees in the random forest model
Minimal samples per leaf	A restriction to the minimal number of data points allowed in each leaf in the classification trees
Criterion for splitting	Criterion used for minimizing the error in each split in the classification trees

Using the Python library Scikit-learn[16][78] a two stage hyperparameter tuning was carried out for each model, using the hyperparameters in Table 3. The hyperparameter tuning consisted of testing which set of hyperparameter values produced the best model using the training data, performance determined based on accuracy. To determine the performance a 2 fold cross validation was used. Before performing the hyperparameter tuning, restrictions were put in place on the intervals for the hyperparameter values tested, to limit the number of hyperparameter value combinations. Limiting the hyperparameter value combinations reduced the required computation time, since less hyperparameter value combinations had to be tested. The Max depth of the trees was restricted to no more than 7, given that 56 respectively 55

[16] `https://scikit-learn.org/stable/index.html`

features were used in the random forest models and the the number of splits in classification trees potentially increase exponentially with the depth of the trees. The Number of trees in the forest was also restricted to no fewer than 40, to reduce the variance of the random forest models. Additionally, a upper bound to the Number of trees in the forest was set to 260, to reduce computing time.

With these restrictions in place, a random search was first performed on each model followed by a grid search. Both the random search and the grid search trained models with different hyperparameter value combinations and compared them to each other to find the best model. The searches were different however in the sense that the grid search tried all possible combinations of hyperparameter values in the intervals, while random search only tried a set number of randomly chosen hyperparameter value combinations from the intervals. The hyperparameters not tested in the tuning were set to the default values of the RandomForestClassifier[17] from the Scikit-learn library. The number of tested hyperparameter value combinations in the random searches was set 1000, to cover a reasonable number of combinations in the intervals while still being restricted to a acceptable computing time. Using 2-fold cross validation the random searches were first performed using RandomSearchCV[18] from the Scikit-learn library. The random searches were performed on the training data with combinations of hyperparameter displayed in Table 4.

Table 4: Hyperparameter value intervals tested in random search.

Hyperparameter	**Hyperparameter interval**
Max depth of the trees	[1, 2, 3, 4, 5, 6, 7]
Number of trees in forest	[40, 60, 80, 100, 120, 140, 160, 180, 200, 220, 240, 260]
Minimal samples per leaf	[None, 1, 2, 3, 4, 5, 6, 7, 8]
Criterion for splitting	[Gini, Entropy]

Using the hyperparameter values received from the random searches, random forest models were trained on the training data. The models were then compared in terms of accuracy when classifying the test data, to models trained on the training data using default options for hyperparameter values. The comparison showed that random forest model 1 and random forest model 2 performed better using the hyperparameter values from the random search, while random forest model 3 performed better using the default options for hyperparameter values. Default options being the default option used by the RandomForestClassifier from the Scikit-learn library. Given the comparison, the best hyperparameter values for each model after the random search were determined to be as displayed in Table 5.

[17] https://scikit\%learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html

[18] https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.RandomizedSearchCV.html

Table 5: Chosen hyperparameter values for random forest models after random search.

Hyperparameter/	**Random forest model 1**	**Random forest model 2**	**Random forest model 3**
Max depth of the trees	7	6	None
Number of trees in forest	80	40	100
Minimal samples per leaf	2	1	1
Criterion for splitting	Gini	Entropy	Gini

The received result from the random search was then used to as an indication of which intervals to use for a grid searches to further examine which combination of hyperparameter values yield the best models. Given the result from the random search, a grid searches with 2-fold cross validation were performed using GridSearchCV [19] from the Scikit-learn library. The previously used restrictions on the hyperparameter values were in this case loosened, to make sure the best possible solution was found. Combinations of hyperparameter values from the intervals displayed in Table 6 were used for the grid search.

Table 6: Hyperparameter value intervals for grid search.

Hyperparameter	**Random forest model 1**	**Random forest model 2**	**Random forest model 3**
Max depth of the trees	[6, 7, 8]	[5, 6, 7]	[5, 6, 7]
Number of trees in forest	[77, 78, 79, 80, 81, 82, 83]	[37, 38, 39, 40, 41, 42, 43]	[97, 98, 99, 100, 101, 102, 103]
Minimal samples per leaf	[1, 2, 3, 4, 5]	[None, 1, 2, 3, 4]	[None, 1, 2, 3, 4]
Criterion for splitting	Gini	Entropy	Gini

The grid search resulted in the hyperparameter value combinations displayed in Table 7 yielding the best models:

[19] https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html

Table 7: Chosen hyperparameter values for random forest models after grid search.

Hyperparameter	**Random forest model 1**	**Random forest model 2**	**Random forest model 3**
Max depth of the trees	7	6	None
Number of trees in forest	80	39	100
Minimal samples per leaf	2	1	1
Criterion for splitting	Gini	Entropy	Gini

4.2.3 Performance of random forest models

Using the hyperparameter values chosen after the grid search, see Table 7, three random forest models were trained using the training data. The random forest models were then tested on the testing data which resulted in the accuracy, precision and recall for the models displayed in Table 8.

Table 8: Accuracy, precision and recall for the random forest models.

	Random forest model 1	**Random forest model 2**	**Random forest model 3**
Accuracy	0.829	0.857	0.963
Precision	0.841	0.875	0.972
Recall	0.725	0.368	0.946

Examining the importance of the features in the random forest models, one could identify certain features that were more important than others in the models. In model 1, Mentions per tweet, Median of words per retweets and Mean words per retweets were the most important features. For model 2 the most important features were, Account age, Unique mentions per tweet, Variance of symbols per tweet and Likes age. Model 3's most important features were, Likes per friend, Friends count and Follower friend ratio.

4.3 Botometer

Using the bot detection framework Botometer, formerly known as BotOrNot [26], all accounts from the the Swedish political data were classified as bot or not bot. The framework uses a Random Forest classifier with 1150 features, 100 trees, Gini-index for splits, to classify the Twitter accounts as bot or not bot [99, p.2–3]. To access the framework, the free API [20] provided at RapidAPI[21] was used. Using the API allowed for sending of requests with

[20] https://botometer.iuni.iu.edu/#!/api

[21] https://rapidapi.com

Twitter ID, which returned the Twitter ID together with an associated Botometer score. An example of a response from the Botomter API is displayed in Figure 3. The Twitter ID and screen name in Figure 3 has been replaced to make sure no personal information is displayed. In the Botometer score, the Universal CAP (Complete Automation Probability) score[22] was used to evaluate the accounts. Every account with a universal CAP score higher than 0.5 were considered as bots, in accordance with previous work from creators of the Botometer framework [32, p.5][102].

[22] https://botometer.iuni.iu.edu/#!/faq

```
{
  "cap": {
    "english": 0.00682617488477388,
    "universal": 0.021041094138717433
  },
  "categories": {
    "content": 0.2348800833972979,
    "friend": 0.2208783213499726,
    "network": 0.41820793343050816,
    "sentiment": 0.27765534203770886,
    "temporal": 0.09933141881781246,
    "user": 0.2120791504162319
  },
  "display_scores": {
    "content": 1.2,
    "english": 0.7,
    "friend": 1.1,
    "network": 2.1,
    "sentiment": 1.4,
    "temporal": 0.5,
    "universal": 1.3,
    "user": 1.1
  },
  "scores": {
    "english": 0.14589765727494236,
    "universal": 0.25152334764543
  },
  "user": {
    "id_str": "42",
    "screen_name": "test"
  }
}
```


Figure 3: Example of Botomter API response.

4.4 Criterion for detecting bots proposed by Kollanyia, Howard and Wolley

To evaluate the accounts in the Swedish Political data with the criterion proposed by [57][50][48][49], the most recent 3200 statuses of each accounts was collected using the Twitter API and the Tweepy library. Further, after examination of the research of [57][50][48][49] it was clarified that the authors evaluated the tweeting behaviour of accounts over a time period of 7 days, meaning that accounts tweeting 450 times or more during 7 days were considered as bots. With the clarification in mind, the criterion for accounts to be considered as bots was formulated as follows:

- Accounts which post at least 450 times within the time span of a week are considered as bots.

Using the most recent 3200 posts, the most recent Tweets were then analyzed with the formulated criterion to find bots.

5 Results of running test methods on Swedish political data

5.1 Individual results of test methods run on Swedish political data

By running the test method on the Swedish political data sets, the number of bots detected by each test method in each data set was determined. The number of detected bots by each test method was then divided by the total number of accounts in the evaluated data set, were evaluated refers to the process having detected bots in a data set. Table 9, Table 10 and Table 11 display the number of bots detected by each test method, divided by the total number of accounts in the data set which the test method was run on. The tables are separated by data set in the Swedish political data.

Table 9: Result of detecting bots in Swedish political data set 1 with test methods.

Method	**Number of bots detected**	**Number of bots divided by total number of accounts in data set**
Criterion proposed by Kollanyi et al.	141	0.144
Botometer	3	0.003
Random forest model 1	20	0.020
Random forest model 2	27	0.028
Random forest model 3	500	0.512

Table 10: Result of detecting bots in Swedish political data set 2 with test methods.

Method	**Number of bots detected**	**Number of bots divided by total number of accounts in data set**
Criterion proposed by Kollanyi et al.	126	0.106
Botometer	5	0.004
Random forest model 1	38	0.032
Random forest model 2	21	0.018
Random forest model 3	596	0.501

Table 11: Result of detecting bots in Swedish political data set 3 with test methods.

Method	**Number of bots detected**	**Number of bots divided by total number of accounts in data set**
Criterion proposed by Kollanyi et al.	2	0.008
Botometer	3	0.013
Random forest model 1	27	0.113
Random forest model 2	4	0.017
Random forest model 3	104	0.437

5.2 A comparison of the results obtained from running the test methods on the Swedish political data

After running the test methods on the Swedish political data, the resulting data sets of detected bots created by the different test methods were compared with each other. The comparison identified the intersection of accounts in the data sets with detected bots, in other words the accounts which the test methods agreed upon being bots were identified. In Table 12, Table 13 and Table 14, a two by two comparison of the intersections between the data sets with detected bots is displayed. Table 12, Table 13 and Table 14 displays the number of accounts in the intersection divided by the number of accounts not included in the intersection, in other words the number of accounts detected as bots by both test methods in the comparison, divided by the number of accounts detected as bots by only one of the test methods in the comparison. By identifying the number of accounts in the intersections, the aim was to recognize if there was any patterns of test methods detecting the same accounts as bots. High numbers indicating that the test methods detected the same accounts as bots and a low number indicating that the test methods did not detect the same accounts as bots. To counteract the phenomenon of methods which detect everything as bots receives a high number, the number of accounts in the intersection was divided by the number of accounts not included in the intersection.

Table 12: A two by two comparison of bot detection methods run on Swedish political data set 1.

	Criterion proposed by Kollanyi et al.	Botometer	Random forest model 1	Random forest model 2
Botometer	0			
Random forest model 1	0.0193	0.0476		
Random forest model 2	0.0122	0	0.1351	
Random forest model 3	0.1058	0.006	0.0417	0.0411

Table 13: A two by two comparison of bot detection methods run on Swedish political data set 2.

	Criterion proposed by Kollanyi et al.	Botometer	Random forest model 1	Random forest model 2
Botometer	0			
Random forest model 1	0.0062	0.081		
Random forest model 2	0	0	0	
Random forest model 3	0.0404	0.0085	0.0681	0.0185

Table 14: A two by two comparison of bot detection methods run on Swedish political data set 3.

	Criterion proposed by Kollanyi et al.	Botometer	Random forest model 1	Random forest model 2
Botometer	0			
Random forest model 1	0	0		
Random forest model 2	0	0.2	0.0345	
Random forest model 3	0	0.0297	0.3506	0.0192

5.3 Combining the results of running the test methods on the Swedish political data

To obtain a broad combined view of the results of running the test methods on the Swedish political data, every account was evaluated individually and then compiled in to lists. For every account the number of times the account had been detected as a bot by a test method was counted, the accounts were then divided in to lists based on the number of times they were detected as bots by a test methods, finally the length of the list were calculated. Before individually evaluating every account, the detection results from certain test methods were excluded.

Examining the result from the test methods run on the Swedish political data in Table 9, Table 10 and Table 11, certain trends could be identified in the results. Random forest model 3 for instance detected a very high number of bots in each Swedish political data set, detecting almost half of the accounts as bots. Given the origin of the Swedish political data, this seems highly unlikely, especially considering the origin of Swedish political data set 3 which consists of manually confirmed human accounts. The result from random forest model 3 was therefore excluded from the combined results. Further, the recall of random forest model 2 was determined to not be good enough, see Table 8, and the result from random forest model 2 was therefore also excluded from the combined results. To visualize the combined results Figure 4, Figure 5 and Figure 6 were created, the figures display bar plots were the bars length represents the number of accounts with a certain number of bot labels. One bot classification can be equated with the account being detected as a bot by one test method.

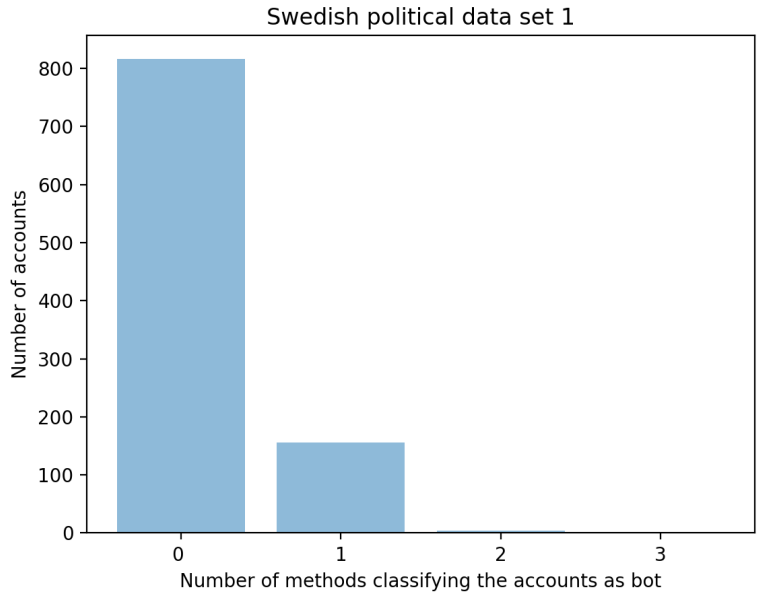

Figure 4: Bar plot of combined result of test methods run on Swedish political data set 1.

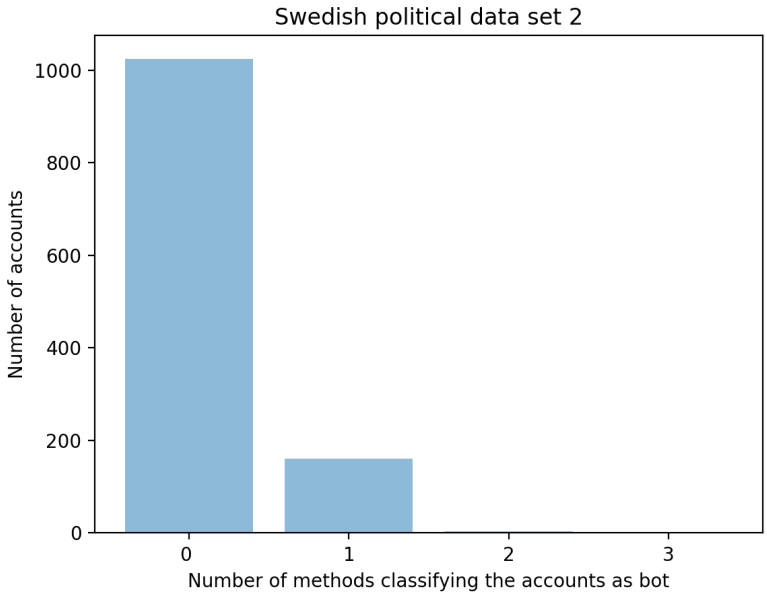

Figure 5: Bar plot of combined result of test methods run on Swedish political data set 2.

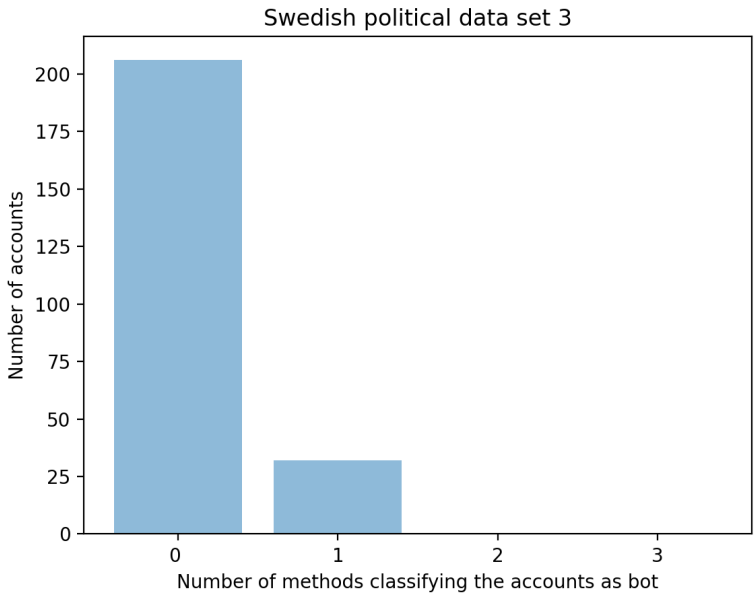

Figure 6: Bar plot of combined result of test methods run on Swedish political data set 3.

6 Discussion

6.1 Evaluating the result of the test methods run on the Swedish political data

As can be seen in Table 9, Table 10 and Table 11, the number of bots detected by the each test methods in the different Swedish political data set varied, none of the test method detected the same number of bots in a Swedish political data set. Comparing the percentage of bots detected by the test method in the Swedish political data to previous research on bot detection in Swedish Twitter data discussing politics, previous research detected 6% [30, p.127] of the accounts as bots, in relation to the test methods which result varied from detecting 0.3% of the account as bots to detecting 51.2% of the accounts as bots, with none of the test methods actually detecting close to the same percentage of bots in the data as in previous research. If the result of the test methods can be compared to previous research is questionable however, since previous research has analyzed other Twitter accounts to detect bots.

Noteworthy is the number of bots detected by the test methods in Swedish political data set 3. Swedish political data set 3 consisted of manually confirmed human accounts, indicating that the test methods should detect 0 bots in the data set, however when examining Table 11 one can see that all test methods detected bots in the Swedish political data set 3. Either the test methods found patterns indicating bot like behaviour which were missed during the manual confirmation, or the test methods failed to accurately detect bots. It should be noted that, apart from random forest model 1 and 3, all other test methods detected a relatively low number of bots in Swedish political data set 3. Random forest model 3 stands out by detecting almost half of the accounts as bots which seems highly unlikely. Random forest model 1 also detected a relatively high number of bots, 11.3% of the accounts detected as bots, which also seem very unlikely. However, not much can be said about the general performance of random forest models as a bot detection method, since the random forest models used in this research are not representative for the research field as a whole of bot detection with random forest models. Several other random forest models have been developed using other sets of features and using other data sets for training [79][83] [60][86][79].

Examining the result of running the test methods on Swedish political data set 1 and 2 instead, Table 9 and Table 10, show that the criterion proposed by Kollanyi et al. and random forest model 3 detected a significantly higher number bots bots compared to the other test methods. The criterion proposed by Kollanyi et al. detected 10-15% of the accounts in Swedish political data set 1 and 2 as bots while random forest model 3 detected roughly half of the accounts as bots. Given the extremely high number of bots detected by random forest model 3 in relation to other test methods result when run on Swedish political data set 1 and 2, together with the specifically high number of bots detected in Swedish political data set 3, random forest model 3 can almost certainly be deemed as an unfit tool to use for bot detection. In Table 12, Table 13 and Table 14 one can distinguish a pattern of almost none of the test methods detecting the same accounts as bots in the Swedish political data sets, the tables consist of overall low numbers. When examining the combined results of the test methods in Figure 4, Figure 5 and Figure 6, excluding the result of random forest model 2 and random forest model 3, the same trends of test methods not detecting the same accounts as bots can be seen. If the test

methods had detected the same accounts as bots, the bars associated with 0 number of bot classifications and 3 number bot of classifications in the figures would have been dominantly larger, meaning that either all three methods detected the accounts as bots, or none of the test methods detected the accounts as bots. However, in Figure 4, Figure 5 and Figure 6 one can clearly see that the bar associated with 0 bot classifications is large, but so is the bar associated with 1 bot classification, further indicating the trend that the test methods did not classify the same accounts as bots.

There could be several different reasons why the test methods are not detecting the same accounts as bots. All of the test methods could be bad at detecting bots, the different test methods could be variously good at finding certain types of bots, one method could be good at detecting bots while the rest are bad. Considering the various reason which could explain the trend of test methods not detecting the same accounts as bots, not much can be said about the test methods separately. However, if all test methods had detected the same accounts as bots, one could have interpreted the result as an indication of the test methods performing well, since the probability of all test method consistently incorrectly detecting the same accounts as bot seems rather unlikely, manual examination of the accounts would still be required of course. Nevertheless, no trend of the test methods detecting the same accounts as bots were found, instead result indicating poor performance of the test methods were found. Running the test methods on Swedish political data set 3 resulted in all test methods detecting bots in a data set with manually confirmed human accounts. The criterion proposed by Kollanyi et al. and Botometer detecting bots among manually confirmed human accounts could even be seen as evidence in favour of the criticism proposed by Kreil in Section 2.5.

6.2 The value of a bot categorization

As described in Section 2.2, several different types of bots on OSNs exist. The bot types can be divided in to different categories based on their purpose, behavioural patterns, how they are controlled, or a combination of these parameters. Depending on the purpose of the research, specific types of bots have been studied, research focused on detecting spam bots on Twitter [97] for instance. Although research on bot categorization exist, the work on the subject has been sparse [66]. It is not clear if a shortage of work on the subject of bot categorization has lead to lack of differentiation between different types of bots in research, still indications of a lack of differentiation can be found in bot related research. For instance, in the paper [21] author's are detecting bots through monitoring of mouse movements and keystrokes. Examining the author's description of bots the following is found, "automated programs, known as bots" [21, p.432], "Bots exploit these online systems to send spam, spread malware, and mount phishing attacks" [21, p.432]. Considering the given description of bots by [21], the study seem detect spam bots, however nowhere in the study does in state that the aim of the study is specifically to detect spam bots. If one chooses to interpret [21] as a study of bots in the general term, then the method developed in [21] is a method to detect social bots, cyborgs, which is not in line with the the type of bots described in the study. It appears like [21] failed to a properly describe what type of bots were examined.

A lack differentiation between different types of bots seem to be key factor in certain parts

of the criticism in Section 2.5 as well. In the criticism, Michael Kreil points out that [26] uses spam bots from [99, p.2] as labeled data to train a model to find social bots, [26] does not seem to have clearly differentiated between the different bot types that exist. Further, Kreil critics the criterion for finding bots used by [57][50][48][49], were bots are defined as accounts tweeting at least 50 times per day. Tweeting 50 times per day could to some extent be a expected behaviour of a spam bot, but nowhere in [57][50][48][49] is stated that the aim of the study is to find spam bots. With a clear definition of what type of bot is being examined, [57][50][48][49] would avoid the possibility of readers misinterpreting the paper as to include detection of social bots and cyborgs. It should be noted that, this research has in no way showed adequate evidence to prove that there is a lack of differentiation between bot types in bot related research. Instead, problems occurring in bot related research have been highlighted in relation to how a lack of differentiation between bot types could be seen as a source of the problems.

Furthermore, when examining the last part of the criticism, see Section 2.2, an additional scenario emphasizing the potential importance of differentiating between bot types can be found. In the last part of the criticism, Karpf criticize the conclusions drawn from bot activity on OSNs, stating that research has drawn too radical conclusions in terms of bot how much bot activity on OSNs affect people's political opinion. Krapf's criticism highlights the need to evaluate if and to what extent bot activity on OSNs actually influence political opinion. Given the lack of studies on bots influence on political opinion [43, p.2], any type of potential preconception should be avoided. With the lack of studies on influence of bots on political opinion in mind, concern could be raised regarding if all type of bots affect political opinion to the same extent. Social bots and spam bots mainly have different activity patterns on OSNs, different activity patterns suggest that they could vary in their effectiveness in affecting political opinions as well. Using tools such as Botomter from [99] or Debot from [18] could hence be a bit problematic in certain type of research, since the methods do not separate between social bots and spam bots. Stating that 10 000 bots were found when examining Twitter accounts discussing the Swedish election does not seem as important if it turns out that 9900 of the detected bots are spam bots which in turn potentially has zero influence on political opinion. If one aims to measure bots' ability to affect political opinion, a lack of differentiation between bot types could lead to difficulties in generalizing the result, the effect on political opinion of 900 bots could potentially greatly vary depending on what type of bots are studied. Moreover, given the wide range of complexity which is stated that social bots can have, from generating simple posts to sophisticated infiltration of human conversations [54, p.156], one can not take for granted that all bots belonging to the social bot category have the same affect on the political opinion of humans either. Further categorization of social bots, as in [88], could therefore be of great value when the effect on political opinion of bots is evaluated.

If social bots can carry out sophisticated task on OSNs, the Twitter Standard API potentially becomes as bottleneck in the process of bot detection. The standard Twitter API enables limited access to likes and replies on tweets, which seemingly is actions a sophisticated bot would carry out if the bot tries to mimic a human. Bot detection methods which analyze data gathered with the Standard Twitter API, such as the test methods, would therefore not be able to utilize all the potential information which could be used to detect bots. The sophisticated

social bots are however a phenomenon which Kreil questions in his criticism. In Kreil's talk at the conference OpenFest Bulgaria [23], Kreil states that he is yet to have been shown a sophisticated social bot and they are simply referred to in research without evidence. The technology does exist to develop very sophisticated social bots, an example is the newly developed chatbot by Google [2], to be noted though is that this chatbot is extremely complicated and not easily accessible, the chatbot is a 2.6 billion parameter neural conversational model. Further research evaluating the level of sophistication of social bots at the moment is however needed to continue a none speculative discussion on this matter .

6.3 The criticism and bot categorization in relation to the categorization of bot detection methods

When considering the criticism proposed by Kreil in Section 2.5 in the light of the categorization of bot detection methods, see Section 2.3, the criticism seems inadequate to question the research field of social bots as a whole. As can be seen in Section 2.3.1 a big part of the bot detection methods used in research are graph-based approaches, methods which the criticism never reviewed, apart from certain features used in Botometer being graph-based [26, p.2]. The criticism does not review unsupervised machine learning approaches or crowdsourcing either , see Section 2.3.3 and Section 2.3.4. This is not to say that the criticism is irrelevant, especially considering that it reviewed social bot research which had received significant attention in news media. Research receiving a lot of media attention could potentially have impact on legislatively proposals, as pointed out by Kreil in the last section of his criticism [58]. The criticism however is need of further development to have a solid foundation on which the research filed of social bots can be critiqued, since only parts of a wide spectrum of bot detection methods have been reviewed.

Further, when considering the wide spectrum of bot detection methods in relation to the different bots types described in Section 2.2, one can emphasize that there is a potential of certain types of bot detection approaches being more suitable for finding certain types of bots. For instance, if ones aims to find a botnets containing spam bots, monitoring synchronized behaviour of huge number of accounts is most likely more relevant than thoroughly examining the social graph surrounding a single account, tailoring the bot detection method to the type of account being detected. With so many radically different methods to detect bots, one has to acknowledge that there is a potential of certain bot detection methods being better at detecting certain behaviour, which in turn could lead to a better performance when finding certain types of bots. Research on bot detection methods can in turn highlight a potential need to develop the bot categorization further. A big part of the bot detection methods are graph-based approaches, methods which utilize social graphs created on OSNs. Studies on social graphs of bot networks have shown that bots have different roles in these networks, certain bots act as content creators while other act as broadcaster, retweeting and liking the content of the content creators [1, p.842-843] [22, p.812]. To better understand the different types of bots that exist additional categories or subcategories of bot types might need to be

[23] https://www.youtube.com/watch?v=vyTmczjwFRE&t=1667s

developed, categories including the different roles bots can have in bot networks.
.

7 Conclusion

This study was conducted to review the process of bot detection on online social networks and evaluate the proposed criticism of current bot detection research. To achieve these aims, five bot detection methods were tested on three different data set to detect bots. Two of the bot detection methods were discarded due to poor performance. The result of running the three remaining bot detection methods showed inconsistency between the different methods in which Twitter accounts were detected as bots. The result gave no indications of the tested bot detection methods performing well, further manual examination of the result is however required to draw any conclusions of the performance of the three tested bot detection methods separately. Still, when the three remaining bot detection methods were used to detect bots in a data set with manually confirmed human accounts, all bot detection methods detected bots in the data set. Since two of bot detection methods were part of the critiqued bot detection methods in the criticism, this could be considered as evidence in favour of the criticism.

Further, a literature review of bot detection methods on online social networks was conducted. The literature review displayed a wide variety of bot detection methods which were not included in the criticism, implying that the criticism needs further development to critique the research field as a whole. A common type of bot detection method not included in the criticism was for instance graph-based approaches. A literature review of bots on online social networks was also conducted, providing a categorization of bot types. When examining the criticism in the light of the bot categorization, certain problems highlighted in the criticism was recognized to potentially have arose from a lack of differentiation between bot types. Moreover, a lack of differentiation between bot types was also recognized to potentially create problems in regards to which conclusions could be drawn from the result of bot detection research. With a lack of differentiation of bot types, a bot's ability to affect political opinion could be hard to measure. Different bot types have different behaviour patterns, which in turn could lead to different bots having different degrees of effect on political opinion, making it hard to generalize the effect on political opinion of bots in the general term. The potential value of adjusting bot detection method to type of bot being detected was also acknowledged. With a wide range of both bot types and bot detection method, certain methods could potentially be better at finding certain type of bots, since the methods use different indicators to find bots and the different bot types vary in their behaviour patterns.

Finally, although it was determined that the criticism of the research field was not extensive enough to question the whole research field of bot detection on online social networks, no evidence disproving the criticism was found either. The uncertainty within the research field on bot detection is highlighted and weak evidence is presented in favour of the research being flawed. The discrepancy between the research community and the news media discourse therefore remains, and news media in Sweden is at risk of spreading information based on flawed research. With this in mind, the author of this study would like to emphasize the need to further evaluate the criticism and the research field of bot detection on online social networks.

References

[1] Norah Abokhodair, Daisy Yoo, and David W McDonald. "Dissecting a social botnet: Growth, content and influence in Twitter". In: *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*. 2015, pp. 839–851.

[2] Daniel Adiwardana et al. "Towards a human-like open-domain chatbot". In: *arXiv preprint arXiv:2001.09977* (2020).

[3] Aftonbladet. *FOI: Botarmé som stödjer SD växer explosionsarta*. https://www.aftonbladet.se/nyheter/a/jPjoXo/foi-botarme-som-stodjer-sd-vaxer-explosionsartat. 2018 (accessed March 30, 2020).

[4] Sveriges Television Aktiebolag. *Ny rapport: Kraftig ökning av politiska bottar på Twitter – starkt stöd för SD*. https://www.svt.se/nyheter/inrikes/ny-undersokning-om-politiska-botar. 2018 (accessed March 30, 2020).

[5] Abdulrahman Alarifi, Mansour Alsaleh, and AbdulMalik Al-Salman. "Twitter turing test: Identifying social machines". In: *Information Sciences* 372 (2016), pp. 332–346.

[6] Lorenzo Alvisi et al. "Sok: The evolution of sybil defense via social networks". In: *2013 ieee symposium on security and privacy*. IEEE. 2013, pp. 382–396.

[7] Kevin Arceneaux et al. "The influence of news media on political elites: Investigating strategic responsiveness in Congress". In: *American Journal of Political Science* 60.1 (2016), pp. 5–29.

[8] The Atlantic. *How Twitter Bots Are Shaping the Election*. https://www.theatlantic.com/technology/archive/2016/11/election-bots/506072/. 2016 (accessed March 30, 2020).

[9] Mariette Awad and Rahul Khanna. *Efficient learning machines: theories, concepts, and applications for engineers and system designers*. Apress, 2015.

[10] Marco T Bastos and Dan Mercea. "The Brexit botnet and user-generated hyperpartisan news". In: *Social Science Computer Review* 37.1 (2019), pp. 38–54.

[11] Monica Bianchini, Marco Gori, and Franco Scarselli. "Inside pagerank". In: *ACM Transactions on Internet Technology (TOIT)* 5.1 (2005), pp. 92–128.

[12] Yazan Boshmaf et al. "Integro: Leveraging Victim Prediction for Robust Fake Account Detection in OSNs." In: *NDSS*. Vol. 15. 2015.

[13] Yazan Boshmaf et al. "The socialbot network: when bots socialize for fame and money". In: *Proceedings of the 27th annual computer security applications conference*. 2011, pp. 93–102.

[14] Danah M Boyd and Nicole B Ellison. "Social network sites: Definition, history, and scholarship". In: *Journal of computer-mediated Communication* 13.1 (2007), pp. 210–230.

[15] Finn Brunton. "Spam: A shadow history of the Internet". In: Mit Press, 2013.

[16] Zhan Bu, Zhengyou Xia, and Jiandong Wang. “A sock puppet detection algorithm on virtual spaces”. In: *Knowledge-Based Systems* 37 (2013), pp. 366–377.

[17] Qiang Cao and Xiaowei Yang. “SybilFence: Improving social-graph-based sybil defenses with user negative feedback”. In: *arXiv preprint arXiv:1304.3819* (2013).

[18] Nikan Chavoshi, Hossein Hamooni, and Abdullah Mueen. “On-demand bot detection and archival system”. In: *Proceedings of the 26th International Conference on World Wide Web Companion*. 2017, pp. 183–187.

[19] Zhouhan Chen and Devika Subramanian. “An unsupervised approach to detect spam campaigns that use botnets on Twitter”. In: *arXiv preprint arXiv:1804.05232* (2018).

[20] Sudipta Chowdhury et al. “Botnet detection using graph-based feature clustering”. In: *Journal of Big Data* 4.1 (2017), p. 14.

[21] Zi Chu, Steven Gianvecchio, and Haining Wang. “Bot or Human? A Behavior-Based Online Bot Detection System”. In: *From Database to Cyber Security*. Springer, 2018, pp. 432–449.

[22] Zi Chu et al. “Detecting automation of twitter accounts: Are you a human, bot, or cyborg?” In: *IEEE Transactions on Dependable and Secure Computing* 9.6 (2012), pp. 811–824.

[23] Zi Chu et al. “Who is tweeting on Twitter: human, bot, or cyborg?” In: *Proceedings of the 26th annual computer security applications conference*. 2010, pp. 21–30.

[24] Stefano Cresci et al. “Social fingerprinting: detection of spambot groups through DNA-inspired behavioral modeling”. In: *IEEE Transactions on Dependable and Secure Computing* 15.4 (2017), pp. 561–576.

[25] Svenska Dagbladet. *Valet 2018 står mellan botar och zombier*. `https://www.svd.se/valet-2018-star-mellan-botar-och-zombier`. 2018 (accessed March 30, 2020).

[26] Clayton Allen Davis et al. “Botornot: A system to evaluate social bots”. In: *Proceedings of the 25th international conference companion on world wide web*. 2016, pp. 273–274.

[27] OV Deryugina. “Chatterbots”. In: *Scientific and Technical Information Processing* 37.2 (2010), pp. 143–147.

[28] John P Dickerson, Vadim Kagan, and VS Subrahmanian. “Using sentiment to detect bots on twitter: Are humans more opinionated than bots?” In: *2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2014)*. IEEE. 2014, pp. 620–627.

[29] Juan Echeverria, Christoph Besel, and Shi Zhou. “Discovery of the twitter bursty botnet”. In: *Data Science for Cyber-Security* (2017).

[30] Johan Fernquist, Lisa Kaati, and Ralph Schroeder. “Political bots and the swedish general election”. In: *2018 IEEE International Conference on Intelligence and Security Informatics (ISI)*. IEEE. 2018, pp. 124–129.

[31] J Fernquist et al. “Botar och det svenska valet. Automatiserade konton, deras budskap och omfattning”. In: *Retrieved from Totalförsvarets forskningsinstitut(FOI): https://www. foi. se/rest-api/report/FOI% 20MEMO* 206458 (2018).

[32] Emilio Ferrara. “Bots, elections, and social media: a brief overview”. In: *arXiv preprint arXiv:1910.01720* (2019).

[33] Emilio Ferrara. “Disinformation and social bot operations in the run up to the 2017 French presidential election”. In: *arXiv preprint arXiv:1707.00086* (2017).

[34] Emilio Ferrara et al. “The rise of social bots”. In: *Communications of the ACM* 59.7 (2016), pp. 96–104.

[35] Stan Franklin and Art Graesser. “Is it an Agent, or just a Program?: A Taxonomy for Autonomous Agents”. In: *International Workshop on Agent Theories, Architectures, and Languages*. Springer. 1996, pp. 21–35.

[36] Peng Gao et al. “Sybilfuse: Combining local attributes with global structure to perform robust sybil detection”. In: *2018 IEEE Conference on Communications and Network Security (CNS)*. IEEE. 2018, pp. 1–9.

[37] Neil Zhenqiang Gong, Mario Frank, and Prateek Mittal. “Sybilbelief: A semi-supervised learning approach for structure-based sybil detection”. In: *IEEE Transactions on Information Forensics and Security* 9.6 (2014), pp. 976–987.

[38] Yuriy Gorodnichenko, Tho Pham, and Oleksandr Talavera. *Social media, sentiment and public opinions: Evidence from# Brexit and# USElection*. Tech. rep. National Bureau of Economic Research, 2018.

[39] Robert Gorwa and Douglas Guilbeault. “Unpacking the social media bot: A typology to guide research and policy”. In: *Policy & Internet* (2018).

[40] Christian Grimme, Dennis Assenmacher, and Lena Adam. “Changing perspectives: Is it sufficient to detect social bots?” In: *International Conference on Social Computing and Social Media*. Springer. 2018, pp. 445–461.

[41] Deepak Kumar Gupta and Ashish Kumar. “Spam and Sentiment Analysis Model for Twitter Data using Statistical Learning”. In: *Proceedings of the Third International Symposium on Computer Vision and the Internet*. 2016, pp. 54–58.

[42] Göteborgs-Posten. *FOI varnar för botar som stödjer SD*. `https://www.gp.se/nyheter/sverige/foi-varnar-fÃűr-botar-som-stÃűdjer-sd-1.7943335`. 2018 (accessed March 30, 2020).

[43] Loni Hagen et al. “Rise of the Machines? Examining the Influence of Social Bots on a Political Discussion Network”. In: *Social Science Computer Review* (2020).

[44] Kerric Harvey. *Encyclopedia of social media and politics*. Sage Publications, 2013.

[45] Yukun He et al. “Understanding socialbot behavior on end hosts”. In: *International Journal of Distributed Sensor Networks* 13.2 (2017).

[46] Mike Hearn. *Did Russian bots impact Brexit?* `https://blog.plan99.net/did-russian-bots-impact-brexit-ad66f08c014a`. 2017 (accessed March 30, 2020).

[47] Simon Hegelich and Dietmar Janetzko. “Are social bots on Twitter political actors? Empirical evidence from a Ukrainian social botnet”. In: *Tenth International AAAI Conference on Web and Social Media*. 2016.

[48] P Howard, B Kollanyi, and SC Woolley. “Bots and automation over Twitter during the second US presidential debate”. In: (2016).

[49] P Howard, B Kollanyi, and SC Woolley. “Bots and automation over Twitter during the third US presidential debate”. In: (2016).

[50] Philip N Howard, Bence Kollanyi, and Samuel Woolley. “Bots and Automation over Twitter during the US Election”. In: *Computational Propaganda Project: Working Paper Series* (2016).

[51] Philip N Howard and SC Woolley. “Political communication, computational propaganda, and autonomous agents-Introduction”. In: *International Journal of Communication* 10.2016 (2016).

[52] Hela Hälsingland. *Kraftig ökning av politiska botar i valrörelsen visar FOI-rapport*. `https://www.helahalsingland.se/artikel/oksanen-kraftig-okning-av-politiska-botar-i-valrorelsen-visar-foi-rapport`. 2018 (accessed March 30, 2020).

[53] Joshua L Kalla and David E Broockman. “The minimal persuasive effects of campaign contact in general elections: Evidence from 49 field experiments”. In: *American Political Science Review* 112.1 (2018), pp. 148–166.

[54] Arzum Karataş and Serap Şahin. “A review on social bot detection techniques and research directions”. In: *Proc. Int. Security and Cryptology Conference Turkey*. 2017, pp. 156–161.

[55] David Karpf. *On Digital Disinformation and Democratic Myths*. `https://mediawell.ssrc.org/expert-reflections/on-digital-disinformation-and-democratic-myths/`. 2019 (accessed April 4, 2020).

[56] Tobias R Keller and Ulrike Klinger. “Social bots in election campaigns: Theoretical, empirical, and methodological implications”. In: *Political Communication* 36.1 (2019), pp. 171–189.

[57] Bence Kollanyi, Philip N Howard, and Samuel C Woolley. “Bots and automation over Twitter during the first US Presidential debate”. In: *Comprop data memo* 1 (2016), pp. 1–4.

[58] Michael Kreil. *The Army that Never Existed: The Failure of Social Bots Research*. `https://michaelkreil.github.io/openbots/`. 2019 (accessed March 30, 2020).

[59] Majd Latah. “The Art of Social Bots: A Review and a Refined Taxonomy”. In: *arXiv preprint arXiv:1905.03240* (2019).

[60] Kyumin Lee, Brian David Eoff, and James Caverlee. “Seven months with the devils: A long-term study of content polluters on twitter”. In: *Fifth international AAAI conference on weblogs and social media*. 2011.

[61] Andrew Leonard. *Bots: The origin of new species*. Penguin Books Limited, 1998.

[62] Jonas Lundberg, Jonas Nordqvist, and Mikko Laitinen. "Towards a language independent Twitter bot detector." In: *DHN*. 2019, pp. 308–319.

[63] Linhao Luo et al. "Deepbot: A Deep Neural Network based approach for Detecting Twitter Bots". In: *IOP Conference Series: Materials Science and Engineering*. Vol. 719. 1. IOP Publishing. 2020, p. 012063.

[64] Stephen Marsland. *Machine learning: an algorithmic perspective*. CRC press, 2015.

[65] Lee-Ellen Marvin. "Spoof, spam, lurk, and lag: The aesthetics of text-based virtual realities". In: *Journal of Computer-Mediated Communication* 1.2 (1995).

[66] Gregory Maus. "A typology of socialbots (abbrev.)" In: *Proceedings of the 2017 ACM on Web Science Conference*. 2017, pp. 399–400.

[67] Michele Mazza et al. "RTbust: exploiting temporal patterns for botnet detection on twitter". In: *Proceedings of the 10th ACM Conference on Web Science*. 2019, pp. 183–192.

[68] Alan Mislove et al. "You are who you know: inferring user profiles in online social networks". In: *Proceedings of the third ACM international conference on Web search and data mining*. 2010, pp. 251–260.

[69] Silvia Mitter, Claudia Wagner, and Markus Strohmaier. "Understanding the impact of socialbot attacks in online social networks". In: *arXiv preprint arXiv:1402.6289* (2014).

[70] Tyler Moore and Ross Anderson. "Internet security". In: *The Oxford Handbook of the Digital Economy'(Oxford University Press 2011)* (2012).

[71] Dieudonne Mulamba, Indrajit Ray, and Indrakshi Ray. "Sybilradar: A graph-structure based framework for sybil detection in on-line social networks". In: *IFIP International Conference on ICT Systems Security and Privacy Protection*. Springer. 2016, pp. 179–193.

[72] Mark EJ Newman and Michelle Girvan. "Finding and evaluating community structure in networks". In: *Physical review E* 69.2 (2004), p. 026113.

[73] Anton Norberg. *Mapping Swedish Parties by Subject Participation on Twitter*. 2019.

[74] Dagens Nyheter. *Försvaret: Ökning av Twitter-botar som försöker påverka väljarna*. `https://www.dn.se/kultur-noje/forsvaret-okning-av-twitter-botar-som-forsoker-paverka-valjarna/`. 2018 (accessed March 30, 2020).

[75] Richard J Oentaryo et al. "On profiling bots in social media". In: *International Conference on Social Informatics*. Springer. 2016, pp. 92–109.

[76] Gautam Pant, Padmini Srinivasan, and Filippo Menczer. "Crawling the web". In: *Web Dynamics*. Springer, 2004, pp. 153–177.

[77] Younghee Park et al. "Antibot: Clustering common semantic patterns for bot detection". In: *2010 IEEE 34th Annual Computer Software and Applications Conference*. IEEE. 2010, pp. 262–272.

[78] Fabian Pedregosa et al. “Scikit-learn: Machine learning in Python”. In: *Journal of machine learning research* 12.Oct (2011), pp. 2825–2830.

[79] Pandu Gumelar Pratama and Nur Aini Rakhmawati. “Social Bot Detection on 2019 Indonesia President Candidate’s Supporter’s Tweets”. In: *Procedia Computer Science* 161 (2019), pp. 813–820.

[80] Muhammad Al-Qurishi et al. “Sybil defense techniques in online social networks: a survey”. In: *IEEE Access* 5 (2017), pp. 1200–1219.

[81] Muhammad Al-Qurishi et al. “SybilTrap: A graph-based semi-supervised Sybil defense scheme for online social networks”. In: *Concurrency and Computation: Practice and Experience* 30.5 (2018), e4276.

[82] Sveriges Radio. *Professor: Twitter måste göra mer mot botar*. `https://sverigesradio.se/sida/artikel.aspx?programid=83&artikel=7031639`. 2018 (accessed March 30, 2020).

[83] James Schnebly and Shamik Sengupta. “Random forest twitter bot classifier”. In: *2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC)*. IEEE. 2019, pp. 0506–0512.

[84] Surendra Sedhai and Aixin Sun. “Semi-supervised spam detection in Twitter stream”. In: *IEEE Transactions on Computational Social Systems* 5.1 (2017), pp. 169–175.

[85] Narendra M Shekokar and Krishna B Kansara. “Security against sybil attack in social network”. In: *2016 International Conference on Information Communication and Embedded Systems (ICICES)*. IEEE. 2016, pp. 1–5.

[86] Monika Singh, Divya Bansal, and Sanjeev Sofat. “Who is who on twitter–spammer, fake or compromised account? a tool to reveal true identity in real-time”. In: *Cybernetics and Systems* 49.1 (2018), pp. 1–25.

[87] Statistikmyndigheten SCB. *Högsta valdeltagandet i riksdagsval sedan 1985*. `https://www.scb.se/hitta-statistik/statistik-efter-amne/demokrati/allmanna-val/allmanna-val-valresultat/pong/statistiknyhet/namnlos/`. Online; accessed March 30, 2020. 2018.

[88] Stefan Stieglitz et al. “Do social bots dream of electric sheep? A categorisation of social media bot accounts”. In: *arXiv preprint arXiv:1710.04044* (2017).

[89] Gianluca Stringhini, Christopher Kruegel, and Giovanni Vigna. “Detecting spammers on social networks”. In: *Proceedings of the 26th annual computer security applications conference*. 2010, pp. 1–9.

[90] Enhua Tan et al. “Unik: Unsupervised social network spam detection”. In: *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*. 2013, pp. 479–488.

[91] Ny Teknik. *Politiska botar ökar under valrörelsen*. `https://www.nyteknik.se/digitalisering/politiska-botar-okar-under-valrorelsen-6927945`. 2018 (accessed March 30, 2020).

[92] Times. *Twitter Bots May Have Boosted Donald Trump's Votes by 3.23%, Researchers Say*. https://time.com/5286013/twitter-bots-donald-trump-votes/. 2018 (accessed March 30, 2020).

[93] The New York Times. *On Twitter, a Battle Among Political Bots*. https://www.nytimes.com/2016/12/14/arts/on-twitter-a-battle-among-political-bots.html. 2016 (accessed March 30, 2020).

[94] TV4. *Botar hotar valet – Skippa nätet och gå till valstugan i stället*. https://www.tv4.se/nyhetsmorgon/klipp/botar-hotar-valet-skippa-nÃďtet-och-gÃě-till-valstugan-i-stÃďllet-11317002. Online; accessed March 30, 2020. 2018.

[95] Valmyndigheten. *Val till riksdagen - Röster*. https://data.val.se/val/val2014/slutresultat/R/rike/. Online; accessed March 30, 2020. 2014.

[96] Valmyndigheten. *Val till riksdagen - Röster*. https://data.val.se/val/val2018/slutresultat/R/rike/index.html. Online; accessed March 30, 2020. 2018.

[97] Alex Hai Wang. "Detecting spam bots in online social networking sites: a machine learning approach". In: *IFIP Annual Conference on Data and Applications Security and Privacy*. Springer. 2010, pp. 335–342.

[98] Gang Wang et al. "Social turing tests: Crowdsourcing sybil detection". In: *arXiv preprint arXiv:1205.3856* (2012).

[99] Onur Varol et al. "Online human-bot interactions: Detection, estimation, and characterization". In: *Eleventh international AAAI conference on web and social media*. 2017.

[100] Ian H Witten et al. *Data Mining : Practical Machine Learning Tools and Techniques*. Elsevier, 2017.

[101] Kai-Cheng Yang et al. "Arming the public with artificial intelligence to counter social bots". In: *Human Behavior and Emerging Technologies* 1.1 (2019), pp. 48–61.

[102] Kai-Cheng Yang et al. "Scalable and generalizable social bot detection through data selection". In: *arXiv preprint arXiv:1911.09179* (2019).

[103] Zhi Yang et al. "Uncovering social network sybils in the wild". In: *ACM Transactions on Knowledge Discovery from Data (TKDD)* 8.1 (2014), pp. 1–29.

[104] Haifeng Yu et al. "Sybilguard: defending against sybil attacks via social networks". In: *IEEE/ACM Transactions on networking* 16.3 (2008), pp. 576–589.

[105] Yang Zhi et al. "Uncovering social network Sybils in the wild". In: *Acm Transactions on Knowledge Discovery from Data* 8.1 (2011), pp. 1–29.