University of Zurich[UZH]

Department of Political Science

How and when are Social-Bots used in social media during election campaigns?

Master's thesis
presented to the Faculty of Arts and Social Sciences
of the University of Zurich

Author: Maël Kubli
Student ID Nr.: 10-929-297

Examiner: Prof. Dr. Marco Steenbergen

Submission date: May 31st, 2019

Table of Contents

Table of Tables

Table of Figures

1 Introduction

With the increasing popularity of Social Media, a new research field for political scientists emerged. One of the first politicians to make excessive use of social media was Barack Obama, whose success in the election campaigns was largely founded on his approach to address potential voters directly via social media. The new form of direct interaction between politicians and their electorate was in the beginning praised by many as a medium for democratization and the promotion of positive social change (Howard et al. 2011; Shirky 2011). But nowadays more and more of these tools should be considered with caution. The reason for this lies in the rising number of content which is generated not by humans but by automatic programs, so-called bots which are controlled by entities with substantial motivations and means to abuse the direct interaction of social media. The rise of bots can result in the deterioration of the political discourse on theses platforms if it is executed professionally. Some of these social bots are quite harmless, but the predominant part is used purposefully to exploit and manipulate or even control the discussion on various social media sites. Scholars fear that bots are used to change the public perception of political entities or to manipulate the outcome of elections. In the US presidential elections of 2016, social bots and trolls were made responsible for the election outcome. However, little attention has yet been given to the detection of bots and entire networks of bots on social media and the analysis of their influence on elections.

Therefore, I focus on the 2018 U.S. midterm elections on Twitter as it is the ideal case to not only analyse more in-depth when and how social bots are used during an election campaign. But also, to put my results in relation to earlier results. This allows me to evaluate if the problem of social bots has remained an issue for social media or if Twitter and others are deploying effective counter measures against the threat of social bots. The first part of my master thesis is devoted to the construction of a classification algorithm to detect social bots. The second part is twofold. On one hand, I analyse if social bots are still used to the same extend as found in earlier works, since Twitter announced to banish them. On the other hand, I will analyse if these social bots are as engaged in the regional campaign as they are on the national level. This will allow me to test whether the current available research results hold if I look at social bots' engagement in the dissemination of the political discourse on the regional level. This could potentially show that the scope of influence for groups trying to affect elections is limited to the executive and thus not as alarming as pictured by the media at least when it comes to general elections.

2 Literature

2.1 Characteristics of Social Bots

Before I start characterizing social bots, I will give a definition of social bots. Based on Varol et al. (2017, p. 1) definition social bots are: "… accounts controlled by software, algorithmically generating content and establishing interactions." This is a very narrow definition and only includes accounts, which are clearly automatically distributing and sharing content either created by other accounts or created by themselves through their own artificial intelligence program. One type of account that clearly is no bot according to this definition is the so-called troll. A troll is an actual person who shares content with a high frequency without regarding the truthfulness of the information shared. This type of user clearly has a negative impact on the social discourse, but is not considered in this work, since they are not built on artificial intelligence.

Social bots are therefore accounts created by people which control social media accounts autonomously without any interaction by a person after they are put in place. They use algorithms to share and create content on their own according to sometimes extensive rules implemented by the designer. Bots employ a great variation of different tactics to reach an audience. A great deal of them is created with the goal of attracting followers to extend their social reach while exerting some form of influence on their followers. Others have entirely different targets like the kind of bots whose only goal is to follow other people to give them a perceived increased popularity on social media. Or there are groups of social bots acting coordinated as one force to generate false popularity for various contents (Chavoshi, Hamooni, and Mueen 2016). The following kind of a social bot who shares content already published is far more frequent than the second one. The reason for this simply is the fact that social bots capable of writing their own posts are more difficult to build. Nevertheless, it is quite possible to build bots that do not really write their own content, but rather collect content from other sources and share it. These bots are often used to distribute certain types of information from sources favoured by the individuals behind the bot to reach more people and influence them with this information. Additionally, a lot of bots try to interact with genuine accounts by sharing content created by genuine accounts which they are programmed to follow and support.

The groups of bots which aim at spreading information are often problematic ones since they are mostly malicious by intent and if not, most would probably agree that it is unethical to use them. As the whole point of social networks is to connect people amongst each other, not people and machines together with each other. Furthermore, when they are used extensively on a large scale bots have the potential to influence external events like elections. Nonetheless, there are simple bots like the *@American_Voter* bot or *@big_ben_clock* bot posting automatically harmless information, which is not intended to mislead or influence anyone but rather to inform people of simple things like who did what when. Similar accounts posting or sharing public information automatically are also common. These bots are easy to identify, and they only share a specific type of content while never trying to misrepresent their motives.

However, the potential risk of bots is the reason why scientists started to try to observe and track social bots. Especially since it was realized by Lee et al. (2011) that there are thousands of social bots which post meaningless messages through automated scripts. They are doing this while randomly following other accounts to expand their personal social network. Moreover, they realized that most of them are not being honest about being a social bot.

2.2 Activity of Social Bots

The activity of social bots is reported in several different domains on social media. This includes not only political discourses like healthcare where bots were observed influencing debates about vaccination policies (Deb et al. 2018) or smoking (Ferrara et al. 2014) but conspiracy theories and others as well.

In the political domain, the first time social bots were found trying to influence elections was back in 2010 during the midterm elections (Ratkiewicz et al. 2010). This turned out to be the starting point of a whole field of research. As of 2016 things started to get highly significant. Then it was realized by different scholars in their works that the number of social bots generating content for the election was very high. This resulted in the possibility that they could distort the opinion formation of people. A lot of social bots seemed to share content of others by re-sharing it to expand the scope of this misleading information to a greater number of people. Bots generated the most positive tweets for Donald Trump to imitate grassroot support for the candidate although there was none in reality (Bessi and Ferrara 2016). The same results were reported for other countries as well. There is some evidence that social bots not only are actively engaged in elections but also in the spreading of news with low credibility (Bessi et al. 2015; Menczer et al. 2018; Stella, Ferrara, and De Domenico 2018; Varol et al. 2017). Subsequently the problem became relevant for the public. Bots are also widely used by terrorists or extremist groups for propaganda, which tells us that bots are starting to distort political stability (Abokhodair, Yoo, and McDonald 2015; Berger and Morgan 2015). Hence, making it even more important to identify them and understand their use.

2.3 Detection of Social Bots

As mentioned before, social media platforms could potentially impact public perceptions and politics. Ratkiewiez et al. (2010) were one of the first, which tried to detect and track political abuse in social media by bots. Their aim was to study astroturf (political campaigns disguised as spontaneous grassroots) in political campaigns on Twitter during the 2010 midterm elections in the U.S. They built a simple classification system to accurately detect astroturf in tweets based on extracted features from the topology of the network. Their main goal was to identify how these messages were delivered, as they formed the hypothesis that the delivery of astroturf is different from other campaigns. Their findings showed that the great difference lies in the type of accounts delivering astroturf. Behind these accounts were mostly bots and not humans (Ratkiewicz et al. 2010, p. 303). This paper and others led researchers to start working on methods of how to detect bots in social networks (Bessi and Ferrara 2016; Davis et al. 2016; Ferrara et al. 2014; Hegelich and Janetzko 2016; Kollanyi, Howard, and Woolley 2016; Varol et al. 2017).

It even led to a competition by DARPA to build a superb classifier for bot detection in 2015 (Ferrara et al. 2014, p. 103). Many of these papers put a lot of effort in building a classifier, with a low rate of false negatives. Most of them succeeded but are not 100 % accurate as they classify some people falsely as bots, which is not as grave but still not very desirable, especially since they classify a lot of organizational accounts as bots (Bessi and Ferrara 2016; Davis et al. 2016; Ferrara et al. 2014). Other authors used very simple methods to determine if the account is a bot or not. For instance, Kollanyi et al. (2016) decided to base their classification solely on the number of tweets someone writes in a day (more than 50 and you are a bot). Ferrara et al. (2016) concluded that it is necessary to build a classifier which uses many different methods (supervised and unsupervised machine learning techniques based on features and graphs) to classify accounts correctly.

The problem with many of these papers is that they lack detailed analysis if the detected bots really played a role in changing the outcome of elections or in contributing to an uprising of new campaigns and opinions. This is something that should be watched over considering that according to Varol et al. (2017) approximately 9% to 15% of all Twitter accounts behave bot-like.

There is, however, some literature on the relationship between bots and elections. Allcot, Gentzkow and Yu (2018) found that after the elections the spread of misinformation declined drastically on Facebook but not so on Twitter. Nevertheless, they still argue that the interactions are quite high, and both platforms are quite important for the diffusion of misinformation, but Facebook found more effective ways to minimize the spread of misinformation than Twitter, or so it seems to be. But it may be just simply the fact that after the presidential election the use of social bots declined as there was no more incentive to let them continue working.

The response of Twitter, Facebook and other platforms alike to mitigate the proliferation of malicious content after the 2016 election are twofold. First, these platforms introduced identity verification systems for new and old users. Second, several methods for search and banning of malicious used accounts through machine learning were introduced. Despite these efforts, there are several arguments why identity verification and machine learning may not work as expected.

With the use of machine learning it may start an arms race between the social media platform and the groups behind social bots. Twitter for one reported that they were able to reduce spam and automated accounts to about half of what it was after the 2016 elections (Selina Wang 2018). Furthermore, before the midterms Twitter made it official that they disabled over 10'000 accounts in September and October which tried to keep democrats off from voting (Bing 2018). Nevertheless, detecting social bots and their counterparts is getting more and more difficult as new social bots share more and more human-like characteristics that allow them to go more often undetected by older algorithms especially the ones that only look at the meta data. Cresci et al. (2018) state that to be able to detect social bots, more in-depth analysis of the accounts collective behaviour is needed. This means that it is no longer enough to look at one account alone to distinguish if it is a genuine account or not, but rather one needs to look at accounts collectively to find networks of social bots. Cresci et al.'s conclusion is quite pessimistic as they argue that older approaches used by other researchers as well as their own older approaches are no longer performing well enough to distinguish social bots from genuine users.

For identity verification there are plenty of guides explaining how to bypass the verification system. This is especially worthwhile for malicious users with high incentives to spread malicious content. In fact, the marginal cost for cheating the system declines with the repetition of bypassing (Kandori 1992; Li, Zhang, and Sarathy 2010). Additionally, with bypassed verification, accounts have more credibility which may widen their influence even more. This in turn may result in more shares from misled genuine accounts. Hence, instead of stopping the spread of malicious information through bots, verification processes and machine learning may even result in an amplified spread of malicious information (Wang, Pang, and Pavlou 2018).

Wang, Pang and Pavlou (2018) tested this argument of a negative relation between the verification process and the reduction on malicious information on the example of Weibo, a social media platform like Twitter in China. Their results support their thesis that verification steps do not reduce the number of malicious accounts, but make them more credible as soon as the verification badges indicate more credibility to the accounts messages. Only if the verification has no impact on the credibility of an account, it will reduce the amount of malicious information, but only to a small degree (Wang, Pang and Pavlou 2018 p. 30 ff.).

Some authors examined how engaged social bots are in the discourse by looking if the tweets of bots got retweeted by humans or if they are followed by many accounts of humans (Bessi and Ferrara 2016; Kollanyi, Howard, and Woolley 2016). Others tried to do this with network analysis (Hegelich and Janetzko 2016; Ratkiewicz et al. 2010). One of the most recent academic research papers by Badawy, Lerman and Ferrara (2018) looked closely at the Russian bot network involved in the 2016 U.S. presidential election. They tried to identify which users spread information from networks of social bots. In detail, their results show that moving from leftist ideology to rightist ideology increases the probability of being a spreader. Additionally, it also shows that highly active users are more likely to spread misinformation. Ultimately, they could show that a lot of misinformation shared by trolls and social bots is later again shared by other social bots.

One other recent work by Menczer et al. (2018) looked into the spreading pattern of misinformation through social bots. The authors further analysed the success rate of bots. They tracked the complete production of over 100 low credibility sources and followed each article's distribution on Twitter. The most important result is that the distribution of low credibility articles is very similar to that of fact-checking articles, but they show some distinctive patterns in the spreading pattern. These patterns are related with the type of accounts actively promoting the articles. The authors hypothesize that so-called super-spreaders of low credibility contents are social bots, which are automatically posting links to articles and retweeting other accounts. Overall, Menczer et al. found evidence that social bots actively try to spread misinformation by exposing humans to their content and inducing them to share. Their success rate is quite high, because only 6 % of their accounts classified as bots were responsible for 31 % of the sharing in tweets linking to low credibility sources (Menczer et al. 2018, p.3).

Yang et al. (2019) demonstrated how well the public is able to defend itself against social bots. The authors investigate the different methods used to classify social bots and how well these instruments really work. While looking into the different methods they report that only newer AI (Artificial Intelligence) tools can detect social bots to a satisfying degree, while older methods especially the ones only relying on the accounts meta-data fail to correctly classify newer bots. Furthermore, the different types of social bots, which are at play on Twitter, receive poor classification rates if the bots searched for are having different behaviours than the ones the classifier was built around (Yang et al. 2019, p 50). Nonetheless, the authors present several promising algorithms, which work over a great number of social bots. These are for example classifiers like the deep neural network classifier used by Kudugunta and Ferrara (2018).

They asked themselves the following two questions (Kudugunta and Ferrara 2018, p. 313): "Is it possible to accurately predict whether a given tweet has been posted by a bot or human account? And: "Is it possible to enhance existing labelled datasets to produce more examples of bot and human accounts without the additional (and very expensive) data collection and annotation steps?". The authors propose a contextual LSTM (long short-term memory) architecture allowing to use both the tweet content as well as the metadata to detect bots at the tweet level. With this method, the authors achieve an extremely high accuracy of 96 % AUC (area under curve). These results show that it is possible to build a classifier capable of classifying tweets without having to search for many more tweets and features of the same user to be able to classify if it is a genuine account or a social bot.

From the literature described here, one could conclude that bots influence voting behaviours of people at elections on a regular basis. The main argument behind this theory is: If there is a group trying to influence an election using bots, they will most likely succeed. Through the social bots' capabilities of sharing and tweeting information or misinformation on the election, faster than a human could. If this is scaled up to a high number of social bots these groups can do that for a much wider audience than normal people could. But only if their bot(s) can reach enough genuine accounts (at best whole social groups) which than start sharing malicious content with other genuine accounts as well. Consequently, if the bots are not connecting with genuine accounts their influence will be small as shown by Allcott and Gentzkow (2017).

3 Theory and Hypothesis

The main problem with social bots nowadays is that it is no longer that simple to identify them. Due to the increasing efforts of social companies to get rid of bots, they have to become more and more sophisticated, so they are not being detected. Social bots are found often using far more retweets than genuine accounts but only to some degree since there are many people which are using Twitter to retweet as much as social bots do. These are persons that are so highly interested in politics that they are retweeting as much information as possible to advertise their views and opinions. Hence, after training a multitude of models to classify accounts as social bots, the most relevant variables which gives away social bots are indeed the account age, the overall sentiment of the tweets, the ratio between friends and followers as well as the ratio between favourited statuses and overall status count. Finally, another giveaway is the fact that some bots still aren't making any effort to change the account appearance from the default appearance.

Unfortunately, no recent paper really investigated how elaborate or superficial theses bots are during elections. Hence, many results of former works can only be viewed with caution. As it is possible that they underestimate or overestimate the impact of bots in social media on election debates and political campaigns and policies. Since they have not looked if bots engage in the dissemination of the political discourse on the regional level focused on candidates or if they are only highly engaged in the political discourse on the national level. Furthermore, no work was interested into the factors that promote the use of social bots.

I believe that the robustness of influence exercised by social bots is overestimated. The cause for this overestimation lies in the scope of the previous research papers, which were always focused around the general campaigns of the national parties instead of looking into regional campaigns as well as on the national campaigns. This method may work only when it comes to presidential elections, but regional factors can also play a central role during the formation of ones voting preference even at the national level. But when it comes to parliamentary elections the regional and local effects are often becoming fundamental as the members represent not only the national party, but their district and the local party section as well. Therefore, I argue that the activity of social bots on the local level is in fact significantly lower and thus the influence of social bots is overestimated during general elections. The reason for this lower activity of social bots is due to their inability to engage in the discourse of local elections in contrast to the national level where it is much easier to engage in the election as there is a broader discourse and audience. Therefore, even with elections with high stakes social bot activity will be lower than the overall activity on the national campaign. Hence, I state that on the national level there is more activity by social bots before elections in general, but not as much on the local level where the bots need to be much more sophisticated to take part in the political discourse of a small area. Furthermore, the smaller audience leads to higher costs for the operators of a social bot or network of bots, since the bot has a smaller target population, which he could reach. These high costs prevent higher bot activities from happening even with relative high stakes in a local election. It is often more beneficial to use national social bots. Bot networks try to help win elections rather than helping one single candidate since it is more important to win the majority of the seats rather than a specific seat and lose the majority. This can be supported by the fact that the candidates rarely change their beliefs towards the mean of a district and thus do not converge. The high correlation between incumbents and district ideologies arise mainly by the fact that voters are presented with partisan choices and thus select the candidate whose party is more like them (Ansolabehere, Snyder, and Stewart 2001). This explains why social bots are more used on the national level as it is often enough to simply promote parties rather than candidates, when there is only a choice between two different parties like in most districts in the US.

My second argument is that the use of social bots is highly dependent on several factors of the election campaign, like the candidates or the financial expenditures of the campaign committees. The first and most important reason for more bot activity in my eyes is how close the race is in a local district. A close race between two candidates leads to a fierce competition on votes to gain that little extra in votes for the win, which is a great incentive for the use of social bots. Secondly, I expect the number of undecided voters to play a role for social bot activity. The more voters are undecided or plan to vote in an election, the more voters are taking part in an election who might not be affiliated strongly with either candidate or his party. Hence the incentive to campaign for those votes is higher when there are more of these independent voters around. Thirdly, the amount of money spent in the campaigns in a district is expected to be relevant as well. This argument simply takes the assumption into account that more money generates more campaigning information which gives a potential social bot more content to share. This argument has been proven by several scholars like Abramowitz (1991), Gerber (1998) and Krebs (1998) all of whom were able to demonstrate that higher expenditures lead to higher chances of winning an election even in relative small homogeneous political units. Furthermore, if there is more money around for a campaign it might be that there are also more resources available which are used to build social bots. Lastly, if the former holder of the seat is not standing in for the election again, both candidates activity is expected to increase. Without an incumbent both candidates have roughly the same chance of getting elected, since none has the incumbency advantage over the other. Hence both sides have roughly the same chance of getting elected at the beginning of the campaigning for office. This advantage of the incumbent has been proven by many different scholars in the past (Ansolabehere, Snyder, and Stewart 2000; Cox and Katz 1995; Erikson 1971; Gelman and King 1990). Ansolabehere Snyder, and Stewart (2000) for instance showed that the incumbency advantage plays an important role when it comes to personal votes, although the effect varies strongly over time. The effect has lately been looked at again where it was found that the effect diminished during the late 90s and early 2000s, but they were also able to show that representatives strategically decide when to run for an office and here they showed that experienced incumbents scare-off other viable contesters (Ban, Llaudet, and Snyder 2016).

Therefore, all four factors are positively related with social bot activity. The higher the chance of turning the election in favour of her or him, the higher are the incentives for both sides to use social bots to influence the voters to elect one of the candidates. In case of the closeness of an election the chance to turn the election in favour for one candidate is more likely to be achieved. Both sides only need to gain a relatively small number of undecided voters to start favouring one side over the other and ultimately vote for them in the end. Consequently, the effectiveness of intensive campaigning is higher, which may result in a higher use of bots. Therefore, the source for the use of bots lies in potentially high rate of effectiveness in terms of the returns social bots can generate under those conditions. To that end, a high number of undecided voters result in a great number of people, which are reachable by campaigning. There the use of bots makes more sense as they are possibly able to influence the opinion formation process of just enough voters toward one candidate or the other as in a condition with only a few undecided voters. In the opposite case it is far more difficult to change the opinion of persons, which already have a strong formed opinion on whom to vote for. Therefore, in such an environment the use of bots is expected to be not be as effective.

For the third source of higher bot activity I rely on the same theory used by the general campaigning theory (Abramowitz 1991; Gerber 1998). The assumption holds that more campaigning output results in more votes. One thing worth mentioning as well from Gerber's research paper is the fact that he found that the incumbents return on campaign spending is around double to the challengers return in terms of votes (1998 p. 406). Hence, I deviate my fourth reason of high bot activity from this argument. I argue that with no incumbent running for office neither side has an advantage or a disadvantage over the other. Hence both sides have the same initial position and thus face the same challenge of winning the election, which results in high efforts on both sides to win votes as there is no higher chance of winning for the losing side from the previous election as well as there is no higher chance of winning for the previous winning side. Furthermore, it might be the case that this advantage of the incumbent translates in a higher bot activity when the incumbent is predicted to lose the election, as both sides will start campaigning even more due to the aforementioned findings of Gerber (1998). Additionally, the social bot activity should even rise more as it is a sign for the challengers side to put more effort in the campaign as it is commonly known that the challenger has lesser chances of winning by default than the incumbent as demonstrated by many researchers (Abramowitz 1991; Ansolabehere and Snyder Jr. 2002; Gerber 1998).

I want to clarify: I do not assume that candidates and their staff per se are actively engaged in creating social bots, but groups with high interests into one candidate winning the election over the other as they may have a lot to lose if the "wrong" candidate wins. Therefore, they will be more inclined to spend money and work on social bots targeted to influence voting behaviour through changing the temper, feeling, spirit and opinion around the candidates. At the same time, I do not rule out the possibility of candidates and their staff being involved in this kind of activity as well, for the same reason. All these factors together are responsible that a highly contested district will have more activity from social bots as not so much contested districts.

Finally, I will compare the ratio of bots during the midterm election campaign period with the ratios found during the presidential election in 2016 and the general election in 2014 by older papers. I do this to check if social media corporations really were able to minimize the misuse on their platforms. As approximated in the literature by Varol et al. (2017) 9 to 15 % of all Twitter accounts behave bot-like. This number is quite interesting, as Twitter announced shortly after the 2016 elections that they would start targeting against social bots and thus the numbers should be lower today, if they really were able to tackle the diffusion of misinformation by social bots. Based on the findings of Wang and Pavlou (2017), that with higher verification costs and better detection algorithms built to minimize malicious content spread there is no decline in malicious content. Since verification stamps lead to the believe of users that the information shared by users is more credible. Furthermore, although Twitter stated several times that they are reducing spam and automated accounts, this does not mean they really were able to reduce it to the degree they say (Bing 2018; Selina Wang 2018). Therefore, my three hypothesizes are as followed:

Hypothesis 1:

Overall, social bot activity is higher on the national level than it is on the regional and local level (federal house-districts).

Hypothesis 2:

The more contested a district is, the higher the number of observable tweets made by social bots.

Hypothesis 3:

Social Bot activity during the campaigning before the US midterm elections 2018 is not lower than it was during the campaigning period before the presidential elections in 2016.

4 Data

4.1 Twitter Data

The 2018 midterm elections in the United States for the Congress and the Senate will serve as the case to be studied in this paper. The US elections are an ideal case due to the high use of social media portals like YouTube, Facebook, Instagram Pinterest, LinkedIn and Twitter. In the US, Facebook and YouTube, which are used by circa 68 % respectively 73 % of the adult population – and thus the voters - in the US, are the dominant players in the landscape of social media platforms (Smith and Anderson 2018).

Hence the optimal case to look at the political discourse before elections would be Facebook. Unfortunately, Facebook does not provide an open API (Application Program Interface) to access data. Therefore, the second most ideal social media platform for this study would be Twitter. Since it is used by approximately 24 % of the population and the younger generations are not as overrepresented in it as in other platforms although not as good as on Facebook (Smith and Anderson 2018, p. 4). Twitter has a very user-friendly API, which is free of charge. This is crucial for my purpose as I will use self-generated data to be able to test my hypothesis. I need data which can be sorted into different areas of interest to compare social bot activity over the whole country with the bot activity in districts.

My data is collected over a timespan of four weeks before the election date and one week after the election (Oct. 6th to Nov. 13th). More than one-month worth of data would be very difficult to handle during the analysis as it takes a lot of computational power to work with tens of millions of observations. Data is accessed with a server infrastructure from Amazon to ensure resilience and stability. I started sampling the data around one month before the election, in order to decide for the districts that should be analysed. Once candidates were clear and the polls showed if districts have close races, the ideal districts could be selected.

I do use the stream API instead of the search API in order to access Twitter data, because of the advantages regarding the needed computational infrastructure. The downside is, it is only possible to stream one percent of all the tweets published world-wide at any time. Hence, I use hashtags to filter tweets relevant for my work. This includes hashtags for the general midterm election, both the democratic party and the republican party as well as specific hashtags aimed at campaigns in 19 different districts for seats in the house where the races were said to be very close according to the Cooks reports (Wasserman and Flinn 2017) and the webpage Ballotpedia in August before the election.

State	Districts
California	10 & 25 & 39 & 48
Colorado	06
Florida	16 & 18
Illinois	12
Kansas	02 & 03
Minnesota	01 & 08
New York	19 & 22
Ohio	01 & 12
Texas	07 & 23 & 32

Table 1: List of districts sampled during the midterm elections

The 19 districts are chosen for this reason and for the fact that all districts had lively campaigns at this time. Additionally, some districts have had rather interesting candidates like Harley Rouda in the 48th district in California who changed the party before running for election. Besides that, I choose districts from both the West- and East coast as well as from the middle West. This is important to get a wide geographical variety of districts into the sample.

For the full list of used hashtags, see Appendix A. Thanks to the use of hashtags to filter relevant tweets, the number of tweets to be streamed remained under the one percent limit, which translates to about 7 Million tweets per day as there are about 700 million tweets each day (Internetlivestats.com n.d.). Nevertheless, my stream collected around 600’000 to 1’000’000 tweets every day which resulted in 28.53 million tweets over the five weeks of data collection[1]. That should be most certainly enough data for this work. The number of tweets were generated by a total of 2.76 million user accounts. Thanks to this custom dataset,

[1] During this period there was a short outage of Twitter around 5 AM in the morning of October the 17th. Luckily the stream reconnected shortly (few seconds) after the service was available again. Furthermore, several times the connection to the stream API was lost for a few seconds when the number of tweets streamed per second exceeded 1 % of the entire tweets generated at the same time. Hence, I need to mention that only about 99 % of all tweets were captured.

I can compare the activity of social bots between different districts with different campaign sizes, different partisanship and different share of voter turnout.

Statistics	**Total**
# **Unique Accounts**	2'759'647
# **Collected Tweets**	28'532'087
# **Replies**	2'119'293
# **Retweets**	17'178'400
# **Quotes**	7'236'707

Table 2: Descriptive statistics of the collected dataset

Furthermore, I can compare all these districts with the national averaged bot activity. Additionally, it is relatively easy to group large parts of the tweets groups favouring the Democrats and those favouring Republicans.

4.2 Classification Training Data

Unfortunately, it is not possible to find social bots without having a dataset to train a classifier to identify social bots in my primary data. Hence, I need Twitter accounts which are known to be social bots as well as Twitter accounts which are known to be used by people, companies or organizations. I settle for the data set made by Cresci et al. in 2017, because it would take a lot of work to come up with my own training dataset with an equal quality. Moreover, their data is the most recent freely available data collection for this purpose. There are a few other data-sets available, but they are often older or smaller and not as detailed as Crescis et al (2017). The data includes 14.4 thousand Twitter accounts (Cresci et al. 2017). Their data set not only differentiates between social bots and genuine accounts but classifies accounts into genuine accounts, social spambots and traditional spambots. Unfortunately, only 5995 different accounts are having all the features I need to build a training set to classify the accounts. Nevertheless, this includes the newest accounts within the data of Cresci et al. (2017). On one hand it is always better to have more training data when it comes to machine learning. Therefore, it is unfortunate that some of the examples are not useable. Nevertheless, for a binary classification with a small number of features around 6000 examples should suffice if one uses adequate regularization techniques to counteract overfitting the classifier when training over and over with a small number of examples. Besides that, older simple social bots in the training data might lead to more false positive classifications in the end. The reason for this potential problem with old bots is: Back then social bots often looked like simple new accounts of people who haven't had the time to make a proper account yet. Hence this should not make a classification more difficult, but should potentially improve classification for newer social bots as well.

For my work, I will not differentiate between these two types of bots, since it would make it even harder to train an accurate model with three classes rather than two. In my definition, there is no difference between the two bot types that would require them to be treated separately. The only risk with using a dataset from 2017 is that there might be new bots around, which do not share the same features as the older bots and thus could go undetected by a classifier relying on the data from Cresci et al. However, the risk for this to happen is quite small since newer bots will always share some features and patterns with older bots.

4.3 Election & District Data

Apart from all the data of Twitter, I need not only the results of the midterm elections in the USA, but also detailed data for the congressional districts themselves as well as data on the campaigning within these districts.

For the election results, I rely on the official results reported by the state secretary office from each state, where I get the actual number of votes for each candidate. Additionally, I am interested in the actual voter turnout in each state of interest which I retrieve from the United States Election Project (MCDonald 2018). Unfortunately, it is not possible to obtain the detailed data per district while writing this paper from the United States government site usa.gov. For the demographic and educational data on the district level this is possible, because the U.S. Census Bureau has a nice little tool which allows easy access to all the data needed like the estimated population in each district and its distribution regarding key values like income, age and education (2019). Apart from these variables the only thing necessary then is the spending data from each candidate and party in the districts which are taken from the Federal Election Commission (2019).

The last data I need for testing the hypothesises is data on the partisanship for each district, which I take from Cook (Wasserman and Flinn 2017). His partisanship data is one of the most often used data for this sort of work. Hence I will use his 2017 report data for partisanship in my work (Wasserman and Flinn 2017).

Sadly, it is not possible to get detailed information from all states regarding the number of registered voters per party by district. Therefore, it is not possible to use these numbers to understand if more unaffiliated registered voters in a district have an effect on the social bot activity on Twitter. It is only possible to see if a higher voter turnout has a positive influence on the number of social bots active on Twitter for a specific district.

5 Method

5.1 Data Classification

5.1.1 Architecture

As mentioned, I use data collected via the stream API of Twitter. This data contains a lot of information, but does not tell you if the tweet is from a social bot or from a genuine account. Since it is only possible to test the hypothesis with this information, I need to build a classifier before running the analysis. In detail, I use ANNs (Artificial Neural Networks), because this method is quite flexible compared to other algorithms (Zhang, Patuwo, and Hu 1998). Neural Networks can generalize very well after learning the data presented to them. Therefore, they are well suited to correctly infer the unseen part of a population whether the sample data is noisy or not. Neural Networks describe a specific kind of machine learning algorithms, which are designed to acquire their own knowledge based on the extraction of useful patterns from the training data (Boehmke 2017).

Neural networks are massive parallel computational systems consisting of an extremely large number of processors with many interconnections with each other over many layers, called neurons. They are based on function approximators, mapping inputs to outputs. Each individual neuron on its own possesses only little intrinsic approximation capability;

however, when many neurons function cohesively together, their shared effects show impressive learning performance. Even if a ANN gets stuck in a local optima it turns out to be often not far off from the global optimum (Géron 2017 p. 258). ANNs are engineered computational models inspired by the central nervous system and brain of organisms (Mitchell 2010; Zhang, Patuwo, and Hu 1998). The artificial analogue of such a biologic neuron is shown in Figure 1. In an ANN model the input layer corresponds to the training, test or real input data values, where for every feature there is one single input node. In deeper architectures with more layers of nodes these nodes are than fed from the output of one or more previous artificial neurons, which are called hidden layers. The transfer function or output layer sums all the inputs together. If the cumulated input value reaches a specified threshold, the activation function generates an output signal (all or nothing). This output signal then moves to one or more output neurons or other neurons depending on the specific ANN architecture.

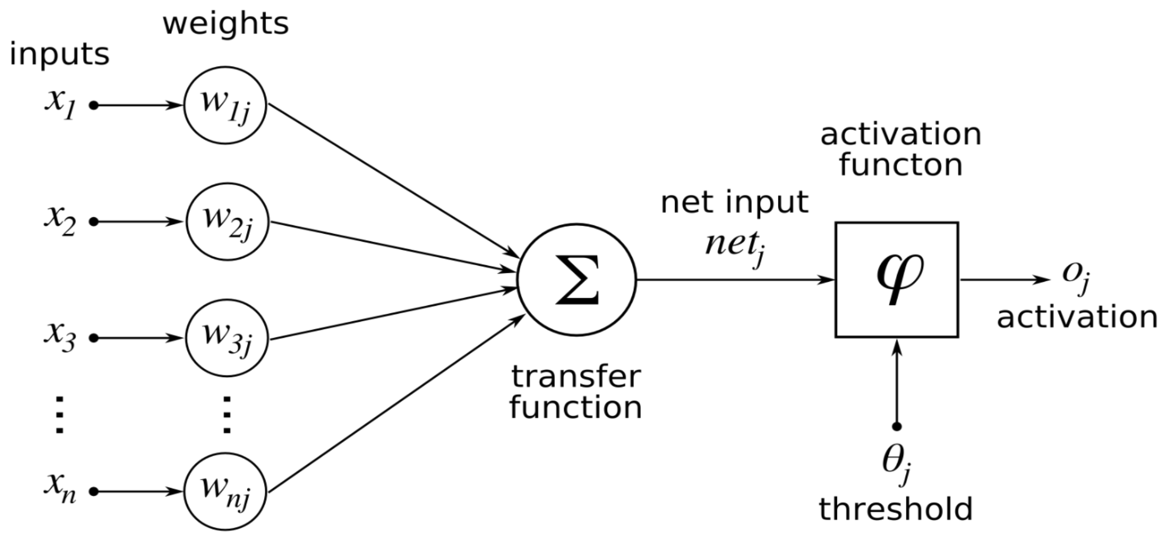

Figure 1: Model of an Artificial Neuron (Source: https://commons.wikimedia.org/wiki/File:ArtificialNeuronModel_english.png)

The capability of ANNs to learn almost any task (given enough training data), is highly dependent on the appropriate selection of the activation function (or activation functions). The activation function is the mathematical conversion used to form the output from the input values. The central advantage of the activation function in an ANN is the possibility to apply nonlinear functions as well as linear functions, which in many other algorithms is more difficult to achieve which helps making Gradient Descent easier if one chooses the right function. The main restriction of ANNs is their limitation of only work at all if every input feature is numeric or at least can be converted into numeric values (categories). Simply put an artificial neurons activation function calculates the weighted sum of one or more outputs from the previous layer or in case of the input layer the weighed input from one feature as every feature needs a single node in the start. Let us consider a neuron where $\boldsymbol{o_j}$ represents the output from the $\boldsymbol{jth}$ neuron in each layer for a network of $\boldsymbol{k}$ input features.

$$\boldsymbol{oj = \varphi(b_j + \sum_{i=1}^{p} w_i x_i)}$$

The output ($\boldsymbol{o_j}$) of the neuron can feed into the output layer of a neural network, or in deeper architectures with hidden layers feed into additional neurons in the next hidden layer. The activation function thus governs if the total sum of the weighted inputs plus a bias term is large enough to cause the firing of the neuron and thus in the end determine the class of an object (Boehmke 2017; Sietsma and Dow 1991; Zhang, Patuwo, and Hu 1998).

The features I use as input for this algorithm are of three different classes which are all either numeric or at least relatively easy to coerce to a numeric format.

Metadata	**Content Features**	**Time Related Features**
# Followers (Integer)	Word Count (Integers)	# Tweets (Integer)
# Of Favorites (Integer)	- Mean	# Retweets (Integer)
# Of Friends (Integer)	- Median	# Tweets per Day (Integer)
Status Count (Integer)	- Standard Deviation	# Friends-Follower Ratio
Listed Count (Integer)	Sentiment Analysis (Integers)	# Favorites-Statuses-Ratio
Default Profile Picture (Logical)	- Mean	# Listed-Tweets-Ratio
Default Banner Picture (Logical)	- Median	
Account Age (Integer)	- Standard Deviation	
Verified Account (Logical)	Tweet / Retweet (Ratio)	
# Digits used in Screen Name	URLs Shared (Ratio)	
# Length of Screen Name	Hashtags (Integers)	
# Digits used in the Description	- Mean	
# Length of Description	- Median	
	- Standard Deviation	
	Mentions (Integers)	
	- Mean	
	- Median	
	- Standard Deviation	

Table 3: List of features used to classify Twitter accounts in the data

The first type of feature is user-based metadata like the number of friends, followers, profile description, profile picture, account age and other settings. The next group of features can be described as content features. This includes the tweet to retweet ratio, the ratio of tweets which contain a URL, and statistics to the number of words in the tweets and simple aggregated data of the sentiment in the texts. The last group of features are time related like the average rate of tweets during a time period. Table 3 shows all the features I use to train my algorithm grouped by the three categories.

Regrettably, there is no universal best practice activation function. Thus, it is necessary to check which type of activation function best suits the data according to the ample information provided by many other researchers. They explain which activation functions work well for ANN solutions for many common problem types (Cheng et al. 1994; Gaudart, Giusiano, and Huiart 2004; Mitchell 2010; Zhang, Patuwo, and Hu 1998). The selection of the activation function dictates the required data scaling required for ANN analysis. In my case, I choose to test whether I can rely either on a rectified linear unit[2] (ReLU) or maxout[3] function.

$$ReLu = \max(0, x)$$
$$Maxout = \max(w_1^T x + b_1, w_2^T x + b_2)$$

A simple test is to run a grid search with both activation functions and see if one ends up having stability problems. Since this is not the case for both types, I use the rectified activation function since it generally makes the training process faster and is easier to compute (Ciaburro and Ventkateswaran 2017; Cook 2016).

The issue of finding a parsimonious model for a problem is quite central for all statistical methods, but it is particularly central with ANNs, because they are very prone to overfitting. Without a working model there is not much generalization capability which renders the ANN useless. Hence it is of great importance to find the right network depth and size with

[2] Rectifier outputs the sum of its weighted inputs but clips all negative values to zero. This makes sure that there are only very few zeros when the neural network is not having many hidden layers and it also means that the positive values are unbounded.

[3] Maxout outputs is a generalization of the ReLU function and has a property of linearity in it. Hence it behaves quite similar with the difference being the fact that it considers not only the max input value but all input values. The downside is that this doubles the total number of parameters for each neuron which needs more time to train.

an appropriate size for classifying or predicting the target variable, while not turning the network in a so-called memory bank (Zhang, Patuwo, and Hu 1998, p. 40). Therefore, a key component for a parsimonious model is to give the network an appropriate size and depth to generalize, while at the same time making sure the network is not growing too big in order to avoid overfitting.

The approach I take to minimize the risk of overfitting is the use of a neural network with random search and dropout technique for the hyper parameterization and regularization. This allows to define a relative wide space of possible hyperparameters while reducing the computation time to find possible areas with certain hyperparameters in which it is possible to find accurate models. This method for training classification models is currently seen as the superior and faster method in respect to a pure grid search while producing results equally well. The main speed advantage is due to the fact that with random search one gets results in all different spaces in the hyperplane area, whereas with grid search one has to wait until all different configurations are trained to get models in all directions of the hyperparameters (Bergstra and Yoshua Bengio 2012).

A Deep Neural Network (DNN) uses multiple hidden layers, various activation functions and L1 or L2 regularization techniques (Lasso Regression or Ridge Regression) and dropout techniques that minimizes vanishing gradient with a shallow neural network, while it minimizes the challenge of overfitting. Furthermore, with a DNN it is easier to work with a large amount of labelled and unlabelled data efficiently.

The key properties to prevent any overfitting is the dropout technique followed by the regularization techniques and early sopping of the training when the model stops improving significantly. Dropout simply put refers to ignoring neurons during the training process of a DNN at random. This means that these neurons are not considered during a forward feeding or backward feeding pass by a probability p. In detail, the dropout ratio describes the probability at each training stage that an individual node is either dropped out of the network or not, which leaves a reduced network at random, which makes it much harder for the algorithm to build a memory bank out of the network and thus suppresses overfitting to a higher degree while improving the accuracy around 1 -2 % (Géron 2017 p. 311 ff). The explanation for this is the fact that with a fully connected layer of all neurons the DNN occupies most of the parameters, and hence, neurons develop co-dependency amongst each other during training which curbs the individual power of each neuron leading to overfitting of training data. Consequently, dropping neurons will help the DNN to rely on more robust and meaningful conjunctions and reduce training time, since the number of possible models is smaller by a factor of the dropout ratio itself (Srivastava et al. 2014).

Lasso and Ridge Regularization are mathematical functions which help reduce the number of complex (parsimonious) models. The Lasso Regularization is a type of linear regression that uses shrinkage to shrink values towards a central point, like the mean. Thus, it is often referred to as a L1 regularization as the Lasso procedure encourages simpler, sparser models, which limit the size of the coefficients with a penalty equal to the absolute value of the magnitude of coefficients. The Ridge regression on the other side adds an L2 penalty, which is why it is referred as L2 regularization, which equals the square of the magnitude of coefficients. All coefficients are shrunk by the same factor. For both types of regularization higher values set for the tuning parameter yield in a higher shrinkage of the coefficients (Géron 2017, p. 309).

Finally, early stopping is used to stop the training as soon as the model either stops improving for a certain amount of training iterations or is already better than a given threshold. This is useful to minimize training time and stop training before overfitting.

One other big downside of DNNs is that with each added hidden layer and many complex feedback-loops it becomes more difficult to follow the ANN's logic decision process up to the point where it is no longer possible to fully understand the decision process at all. This is then often referred to as a black box. Nevertheless, the performance of neural networks is often better than that of most other algorithms. I will deal with the minor inconvenience of a possible black box situation as it is more important to predict as many correct account types as possible instead to fully understand the DNN's decision process. Nevertheless, it is key to use as little layers as possible while training a neural network as more layers result in a higher chance of overfitting.

To evaluate the robustness and accuracy of the DNN I leave 15 % of data for testing plus 15 % for validation (out-of-sample data) instead of using it for training (in-sample data). The training set is used to train the algorithm to find the weights in the network with a 5-fold cross validation. For a binary classification problem, the predicted target variable and the actual target variable can be in any of the following four relations to each other:

- True Positive: All cases where the predicted and the actual value are both true.
- True Negative: All cases where the predicted and the actual value are both false.
- False Positive: All cases where the predicted value is positive, but the actual value is false. This is referred to as a type one error.
- False Negative: All cases where the predicted value is negative, but the actual value is true.

For the evaluation of my randomized grid search for the best model I rely on the log loss metric to evaluate how many layers and which features the DNN shall use. The log loss function measures the performance of a classification model where the prediction input is a probability value between 0 and 1. The goal of any DNN model is to minimize this value. A perfect model would have a log loss of zero. Log loss increases as the predicted probability diverges from the actual label. Therefore, predicting a probability of close to zero when the actual observation label is 1 would be hurtful and result in a high log loss. This is the advantage of this metric over other metrics like accuracy (Cook 2016; Géron 2017). Other forms of evaluation metrics only look at the numbers of rightfully and wrongfully predicted cases, whereas log loss takes the uncertainty of the prediction into account as well, based on how much it varies from the actual label. This gives me a more nuanced view into the performance of my model. To train a DNN I use H_2O as the program of choice for my problem since it is freely available and relatively fast and easy to use, while giving a lot of freedom to set the hyperparameters (Cook 2016).

5.1.2 Classification Training

Since a random grid search takes a lot of time to detect the best model specification with many different hyperparameter settings I use random search in several steps. As mentioned, the first step is to guess which activation function to use while specifying the number of hidden layers needed at minimum to get models that generalize well. All while still being able to classify data with high accuracy, while leaving all other parameters to the default settings H_2O uses.

Activation Function	Mean Log Loss	Standard Deviation of Log Loss	Variance of Log Loss
Maxout with Dropout	0.370	0.212	0.045
Rectifier with Dropout	0.078	0.022	0.001

Table 4: Performance comparison between ReLU and Maxout as activation functions

In total, I test neural networks with one, two and three hidden layers with two configurations each, regarding the number of neurons in each layer, and the two different activation functions (ReLU and Maxout), which adds up to 12 different model configurations in a first step. In Table 5, one can see that from the 12 models trained the type of activation function used for the different DNN's has a mentionable effect. While the average log loss of the models using ReLU with dropout is 0.078, the average log loss of models using Maxout with dropout is 0.370. This demonstrates quite clearly that ReLU is to be used instead of the Maxout activation function. When we now look at Table 5 showing the performance of all 12 models trained during the first step to determine the optimal number of layers and the activation function we can see that models using two, three and four layers are all performing quite well. Hence the extensive randomized grid search with dropout will need models with two, three or four hidden layers which is not deep.

Activation Function	Number of Hidden Layers	Number of Neurons in each layer	Log Loss
Rectifier with Dropout	4	[25, 20, 15, 10]	0.058
Rectifier with Dropout	4	[20, 20, 10, 10]	0.061
Rectifier with Dropout	2	[20, 10]	0.061
Rectifier with Dropout	3	[20, 10, 5]	0.082
Rectifier with Dropout	2	[10, 5]	0.098
Maxout with Dropout	3	[15, 10, 5]	0.110
Maxout with Dropout	3	[15, 10, 5]	0.127
Rectifier with Dropout	2	[20, 10]	0.188
Maxout with Dropout	2	[10, 5]	0.271
Maxout with Dropout	4	[25, 20, 15, 10]	0.457
Maxout with Dropout	4	[20, 20, 10, 10]	0.492
Maxout with Dropout	3	[20, 10, 5]	0.686

Table 5: Performance of all models form first grid search

Nevertheless, the small log loss values show very well that neural networks are working well as a classifier for the problem at hand. I settle for two hidden layers since it makes more sense to use only two layers as the classification problem is not a simple linear function of the input features. I will not add more layers since the preliminary grid search already showed that two hidden layers lead to well performing neural networks and thus it would only increase the risk of overfitting the data. For the number of hidden neurons, I follow the guidelines made by Cook (2016), who argues that the number of neurons should not exceed the number of input features plus the number of output neurons. In case of my data I have 36 input features and two output classes.

Thus, the max number of hidden neurons should not be bigger than approximately 40 and not be smaller than half of that number. Furthermore, Cook and Géron argue that the number of hidden neurons in the final hidden layer is most likely dependent on the number of output neurons. I only have two output neurons, one for bots and one for genuine accounts. Hence the number of hidden neuron in the last layer should be smaller than the number of neurons in the first hidden layer (Cook 2016; Géron 2017).

In the second step, I run an extensive randomized grid search with dropout since this is recommended to use to make it possible for the neural network to set the amount of outputs that are randomly set to zero as a regularization technique to avoid overfitting and give a robust model. I use the following hyper-parameters for the grid:

- Two layers with the number of hidden neurons set to: [30,5], [25,5], [30,3] and [25,3]
- Lasso Regularization values of $1e^{-6}$, $1e^{-7}$, $1e^{-8}$
- Ridge-Regression Regularization of $1e^{-6}$, $1e^{-7}$, $1e^{-8}$
- Input dropout ratios of 0.1, 0.15 and 0.2
- Hidden dropout ratios of 0.3 and 0.4
- Stopping-tolerance of 0.02, 0.03 and 0.04
- Epochs (number of times training data is being used to train) of 2000, 3000, 4000

Other values I change from the default values in H_2O are the number of stopping rounds which I set to three. Then I set the maximum incoming weights to a neuron to two to stop weights of growing too big. This in turn helps reducing the risk that some neurons end up having too big of an impact on the classifier itself, which would hurt the classifiers ability to find potential more complex patterns to identify social bots. Furthermore, I instruct H_2O that missing values are imputed with the mean instead of dropping them, since the training dataset only contains 5'995 users minus the 30% for testing and validation of the models. Lastly, I make sure the samples to train from are stratified and shuffled.

In total, the machine trained 129 models (see: Appendix B for performance table of all models) which yielded models with log loss as small as 0.047 to log loss values as high as 0.206. These models are overall performing well enough. This allows to end the model optimization process, since the chance of overfitting would arise with longer training times which would be unhealthy for the goal of classifying Twitter accounts which posted tweets related to the midterm elections in the US.

Now let's take a closer look at the best model, its configuration and at possible generalizations. Table 6 reveals the model's most important specification. The model uses an input dropout ratio for the features of 0.2 and hidden drop out ratios for 0.3 in both layers with 25 nodes for the first hidden layer and 5 nodes for the second hidden layer with a stopping tolerance of 2% and L1 and L2 regularization values of 10^{-7}.

Parameter	Value	Description
nfolds	5	Number of folds for K-fold cross-validation
fold_assignment	Stratified	The stratified option will stratify the folds based on the response variable, for classification problems
activation	Rectifier with Dropout	Activation function
hidden	25, 5	Hidden layer sizes
epochs	2004.6914348559217	How many times the dataset should be iterated (streamed)
input_dropout_ratio	0.15	Input layer dropout ratio (can improve generalization)
hidden_dropout_ratios	0.3, 0.3	Hidden layer dropout ratios (can improve generalization), specify one value per hidden layer, defaults to 0.5.
l1	1e-8	L1 regularization (can add stability and improve generalization, causes many weights to become 0)
l2	1e-8	L2 regularization (can add stability and improve generalization, causes many weights to be small)
max_w2	2	Constraint for squared sum of incoming weights per unit
stopping_rounds	None	Early stopping based on convergence of stopping metric
stopping_metric	Log loss	Metric to use for early stopping
stopping_tolerance	0.02	Relative tolerance for metric-based stopping criterion (stop if relative improvement is not at least this much)
shuffle_training_data	True	Enable shuffling of training data

Table 6: Important model parameters of the best performing model

The training of the classifier model was done like all other models with a 5-fold cross validation using 70 % of the data to train the model and 15 % of the data to validate the model after each training step and leave 15 % of the data for testing. Since the training data is rather small it turned out that the model needed to iterate 3'000 times through the training data before stopping with the training process.

5.1.3 Classification Evaluation

The model's validation statistics are still very impressive. The overall max accuracy is 0.9856 or 0.9845 at a threshold of 0.5. Hence the model seems to be exceptionally. The same is true for the precision and specificity of the model, which is 0.9932 and 0.9695 respectively at the threshold of 0.5. This means that the model is very well suited to classify accounts since it is capable to correctly identify the type of user of a Twitter account. One can calculate from the numbers in the following Table 7 that the true positive rate and true negative rate is 98.78% and 97.56% respectively if we set the threshold towards the maximum F1-score[4].

Actual/Predicted	**0**	**1**	**Error**	**Rate**
0	160	4	0.0244	4 / 164
1	9	728	0.0122	9 / 737
Total	**169**	**732**	**0.0144**	**13 / 901**

Table 7: Validation metric of the best model with the threshold set to 0.6 at its maximum F1 score.

There is still the possibility that the data used to train the model might lead to overfitted results even after validation. I withhold 15 % more of the data to test the model one more time. Here we see that it looks like the model is not overfitted. The resulting confusion matrix shown in Table 8 on the next page shows similar results as before with a perfect true negative score of 100 % and a true positive score of 99.4 %.

Actual/Predicted	**0**	**1**	**Error**	**Rate**
0	164	1	0.006	1/165
1	0	729	0	0/729
Totals	**164**	**730**	**0.001**	**1/894**

Table 8: Test metric of the best model with the threshold set to max F1 score.

One problem I face with my data is the fact that my classification is based only on 36 input features of which ten are Boolean (binary) for which both states (True / False) have their own input node, while all other features are variables which are interval scaled and thus have only one input node. Small numbers of input features might make it more difficult to build a finer grained generalizing classifier. Hence having an input dropout ratio set with only 36 features might not only reduce the probability of overfitting but reduce

[4] The F1-score or F-measure is a measurement for the accuracy of a binary classifier. It considers the precision (p) and the recall (r) of the test to compute the score. The precision is the number of correct positive results divided by the number of all positive results returned by the classifier, and recall (r) is the number of correct positive results divided by the number of all relevant samples. Therefore, the F1-score is the harmonic mean of precision and recall (Kelleher, Mac Namee, and D'Arcy 2015, p. 414 ff.).

generalization as well as chances rise that some more robust neurons and input features are dropped as well. Nevertheless, it is necessary to work with dropout when the training data is as small as in my case as has been discussed before.

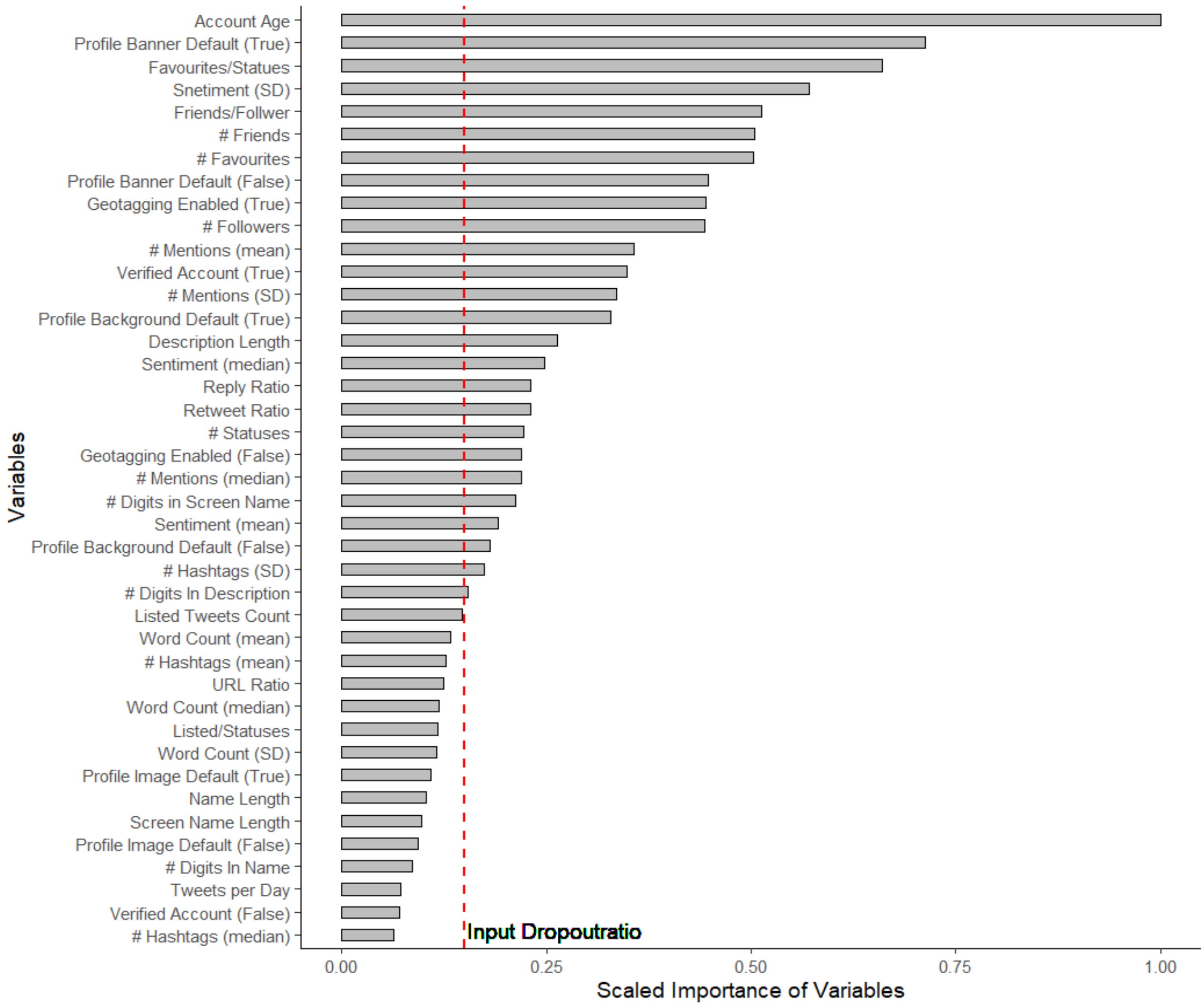

Figure 2: Input variable importance of all features used in the model for the prediction of the account type

5.1.4 Classification Results

Now let's look at the classification of the accounts in my data from the classifier with the threshold set to 0.6 to its maximum F1 score. Out of the 2'759'647 unique accounts the model classifies 416'668 accounts as social bots and 2'342'979 as genuine accounts or 15.1 % social bots and 84,9 % tweets from genuine users. This translates into 8.6 % of the total tweets published by bots over the full collection period while holding the threshold to 0.6 at the classifier's highest F1 score.

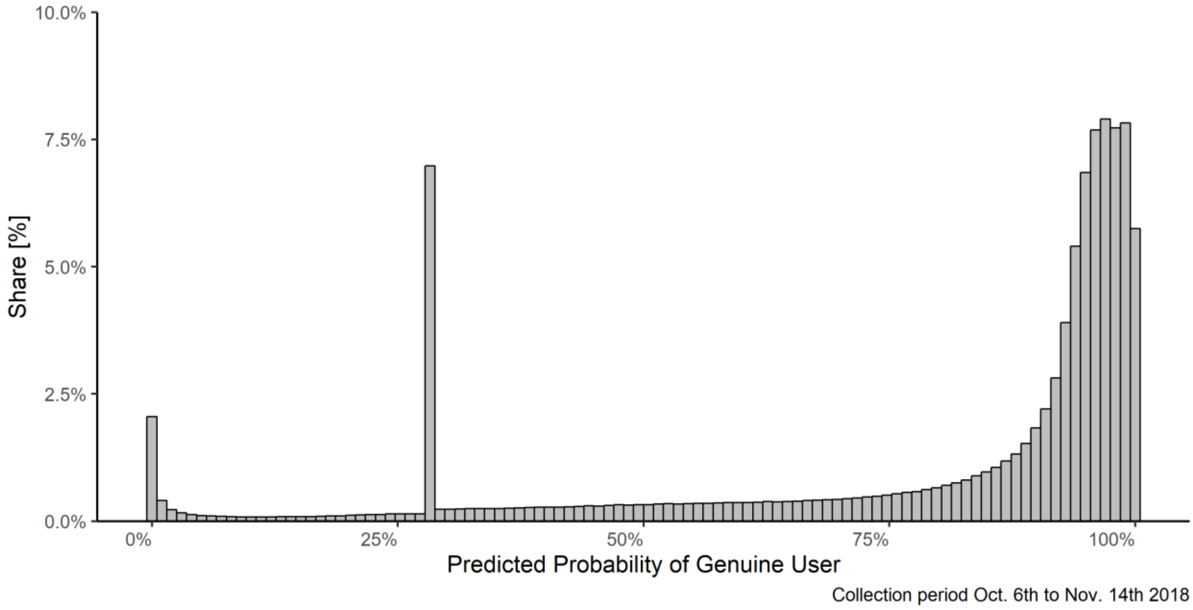

Figure 3: Distribution of the probability of an account being classified as a genuine account in my dataset

One thing which is striking is the fact that about half (187'293) of the accounts classified as bots have a very similar probability of being a bot as can be seen in the following table. This is somewhat suspicious since the distribution shouldn't contain such a strong outlier.
Hence, I have to look more closely into this peak by examining the variation of these accounts over all features used by the classifier as well as the number of tweets obtained by each of these accounts during the collection period. This enables me to see if all these cases are very similar. As this would indicate that the classification worked well, and the problem is in fact just a high number of very bot like accounts in the data. Otherwise it would indicate that the classifier has a severe classification problem.
Fortunately, it looks like it is the first case. Table 9 illustrates that the scaled (zero to one) features used with the classifier have overall relative low variances ranging from 10^{-8} to 0.18. Due to this low overall variance in all features used for the classifier it makes sense that these users receive a similar predicted value for being a bot. Hence it is not necessary for me to lift the threshold even higher up than suggested by H_2O, which uses the F1 score to determine the threshold, by setting it to the highest F1 score achieved by all possible thresholds.

Variables	Variance	SD	Mean	Variable Type
Profile banner default	0.176	0.42	0.771	binary
Retweet ratio	0.092	0.303	0.791	numeric
Description length	0.072	0.269	0.328	numeric
Account age	0.068	0.261	0.321	numeric
Profile image default	0.048	0.218	0.05	binary
Screen name length	0.03	0.173	0.755	numeric
Geo enabled	0.021	0.146	0.022	binary
URL ratio	0.021	0.146	0.071	numeric
SD word count	0.019	0.136	0.268	numeric
Screen name digits	0.016	0.127	0.063	numeric
Median sentiment	0.016	0.127	0.072	numeric
Name length	0.016	0.127	0.25	numeric
Mean sentiment	0.016	0.126	0.071	numeric
Median word count	0.012	0.111	0.284	numeric
Mean word count	0.011	0.107	0.283	numeric
Verified	0.011	0.105	0.011	binary
SD sentiment	0.009	0.097	0.168	numeric
Median n hashtags	0.008	0.091	0.097	numeric
Mean n hashtags	0.008	0.09	0.098	numeric
Reply ratio	0.006	0.077	0.043	numeric
SD n mentions	0.004	0.063	0.065	numeric
SD n hashtags	0.003	0.058	0.059	numeric
Friends follower ratio	0.002	0.041	0.02	numeric
Profile background default	0.001	0.035	0.999	binary
Mean n mentions	0.001	0.032	0.016	numeric
Median n mentions	0.001	0.03	0.012	numeric
Name digits	0.001	0.029	0.005	numeric
Favourites count	0.001	0.028	0.012	numeric
Favourites statuses ratio	0.001	0.026	0.017	numeric
Description digits	0.000	0.018	0.006	numeric
Listed count	0.000	0.015	0.003	numeric
Tweets per day	0.000	0.015	0.005	numeric
Statuses count	0.000	0.014	0.005	numeric
Friends count	0.000	0.014	0.004	numeric
Followers count	0.000	0.008	0.001	numeric
Listed statuses ratio	0.000	0.003	0.000	numeric

Table 9: Showing variances and standard errors as well as the mean and class of variable for the accounts with the predicted bot rate of 0.717383

One problem with the accuracy of the classifier remaining is the fact that the classification in the training data is done with all the metadata regarding the account and their tweets. This is not always the case with my data since I only have as many tweets as the accounts posted with at least one of the hashtags I streamed for. Therefore, I have a lot of users for which I only have very few tweets. This results in high uncertainties around the features regarding the content of the tweets, which might lead to a lot of cases where the account is classified falsely as a bot or a genuine user due to high standard errors, which are not due to highly polarized contents but due to low case numbers. Consequently, I remove all users from the analysis, which have posted less than 10 times in the observed timeframe. This should be enough to minimize the standard errors of the content related features to a low enough variance for the classifier to predict the account type.

Statistics	**Total**
# **Unique Accounts**	335'742
# **Unique social Bots**	28'755
# **Unique Genuine Users**	3'06987
# **Collected Tweets**	23'682'305
# **Tweets by Social Bots**	1'538'050
# **Tweets by Gunine Users**	22'144'255
# **Replies**	1'858'251
# **Retweets**	14'631'533
# **Quotes**	5'863'310

Table 10: Descriptive statistics of the collected dataset after filtering all users with less than ten tweets in the collected data.

After removing users with less than 10 tweets I am left with only 335'742 users from the total of 2.76 million users in the data. Nevertheless, this is a sufficient number of users for the analysis since these roughly 12% of users are responsible for 83% of the tweets. Besides the numbers for the predicted probability of a user being a bot doesn't change a lot as well, if the users with less than 10 tweets are removed.

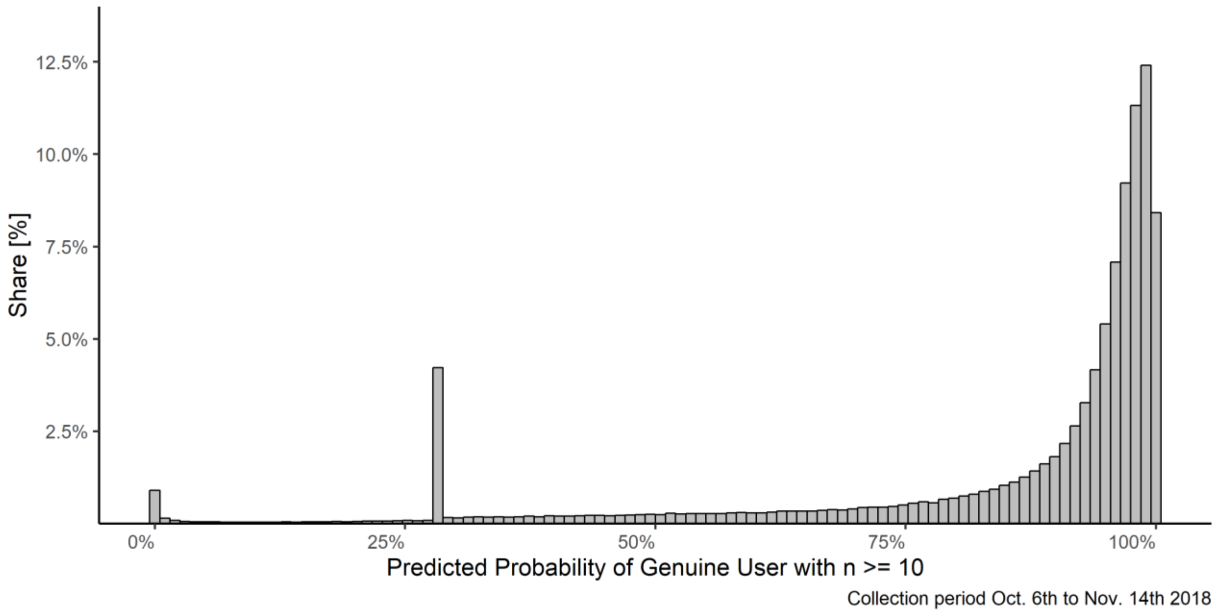

Figure 4: Distribution of the probability of an account being classified as a genuine account in my dataset after dropping all users with less than 10 tweets in the collected data.

This displays the fact that the classification works to some degree even with low numbers of tweets. The only considerable change is the fact that without the users with low numbers of tweets, the overall low scores get a bit higher, whereas the rest stays roughly the same except for the peak value at 0.71 which is now 2.7 % lower than before. Hence, the graph exposes that removing users with low numbers of tweets is a robust decision and should therefore improve the reliability of the results.

The classifier finally classifies 28’755 users as social bots from the remaining 335’742 accounts, which represent 8.6 % of all users and 6.5 % of all tweets in the analysed data set.

5.2 Hypothesis Testing

To test the hypotheses after the classification process my first step is to allocate each tweet to its associated district or to the national campaign group. This includes matching the districts tweets to the population, share of urban population, the median age, median income, and education of each district or for the U.S. Additionally, I will have to add all variables related to the campaigning in each district and the U.S. The matching is achieved through assigning for each tweet the right district via it’s hashtags used in the text of the tweet.

In a first step, I analyse the spatial and temporal dynamics of information production during the observation period, to highlight differences between the national bot activity and the house-district bot activity. Besides this descriptive part I calculate the mean difference between the national campaigning bot activity and the districts campaigning bot activity. Then, I use a logistic regression to test if the observed difference is in fact significant and holds even when we take the other variables mentioned before into account. This allows me to test if the first hypothesis holds or not.

Secondly, I compare the different bot activities in all districts of choice after matching the social bot activity rate with the same covariates as mentioned before and all four independent factors with a logit model as well. The factors are campaign spending by both parties, incumbency, closeness of election results and average partisanship of the district. This is done to test whether there is a significant effect visible regarding more contested districts in contrast to uncontested districts. The construct to measure, the level of campaign intensity is deduced from all variables mentioned before. Only the incumbent status in a district which is examined on its own since it is often named one as of the most important factors in research centred around campaigning.

Because the dependent variable social bot or not is dichotomous, it makes no sense to use a simple linear model. First, the errors are not normally distributed, and a linear regression would not just return either zero or one as a result for an observation. To avoid these problems, we need to force the linear predicted values of $\eta_i = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_k x_{ik} = \mathrm{X}_i\beta$ to lie in-between the unit interval of $\eta_i \in [0,1]$. This is achieved neatly in the logistic regression with the logit link function which maps the linear predictor η_i into the unit interval. The underling concept of the logistic regression is the logit function or the natural logarithm of an odds ratio. In detail the logistic model predicts the logit of the dependent variable Y from the independent variable(s) X. The formula for the logistic model has thus the following form:

$$logit(Y) = \ln(odds) = \ln(\frac{\pi}{1-\pi})$$

Generally, the logistic regression is well suited to describe and test hypothesises around relationships between a categorical dependent variable and one or more categorical or continuous independent predictor variables (Peng, Lee, and Ingersoll 2002). Since it is difficult to grasp the concept of odds ratios, it is often helpful to calculate the predicted probabilities of outcomes to display the effects graphically via a simulation. This allows me to display which conditions in an electoral campaign lead to how much change in social bot activity and thus tend to be more prone to malicious content or even illicit manipulations of the voter.

To test the overall model fit I rely on the Akaike information criterion, which provides a method for evaluating the quality of the model through comparison of related models. Unfortunately, the figure itself is not very meaningful. Only if I have more than one model with similar independent variables and differing numbers of control variables the AIC is suitable to determine which model fits best, but only when all the variables of the reduced models occur in the more complete ones. In this case, I will select the model that has the smallest AIC. But to address the overall fit of the model type the AIC is not useful. To test exactly how well the model fits, I first use the Loglikelihood test, which links the likelihood of the data under the complete model against the likelihood of the data under a model with fewer predictors. Nevertheless, it is necessary to test whether the observed difference in model fit is statistically significant or not. Given that the null hypothesis holds that the reduced model is true, a p-value for the overall model fit statistic which is less than 0.01 would tell us to reject the null hypothesis. It would provide evidence against the reduced model in favour to the current model. Since a logistic regression provides a better fit to the data if it demonstrates an improvement over a model with fewer predictors. Furthermore, I perform an analysis of variance (ANOVA), which is used to evaluate the statistical significance of each coefficient in the model.

6 Results

My analysis objective is the investigation of two different aspects of social bots. First, I examine the temporal dynamics of the publication of tweets by humans (genuine accounts) and bots. Here I investigate the difference of these temporal dynamics between information production concerning the sample of congress district campaigns and the national campaign discourse. Moreover, I try to highlight the differences between the republican and democratic leaning accounts use or non-use of social bots during the midterm election campaign. Then I analyse whether my hypotheses hold up using the analysis proposed in the methods section to test the first two hypothesises. Lastly, I will compare the observed share of social bot activity during the midterm election of 2018 with results from former elections.

6.1 Temporal Dynamics

Figure 5 displays the volume of tweets present in the data set after removing all tweets made by accounts with less than 10 tweets during the collection period from October the 6th to November the 14th. As the graph depicts, bots are only responsible for a relatively small portion of all the tweets released at any given time during a day. Social bots are only responsible for about 4 - 7 % of the tweets published at any given time. Something rather surprising is the fact that social bots and genuine accounts start to increase the number of tweets around the same time before an election. Social bots reach like the humans a maximum of publications at the election day. Therefore, social bots are not showing any signs of heightened activity weeks before the election.

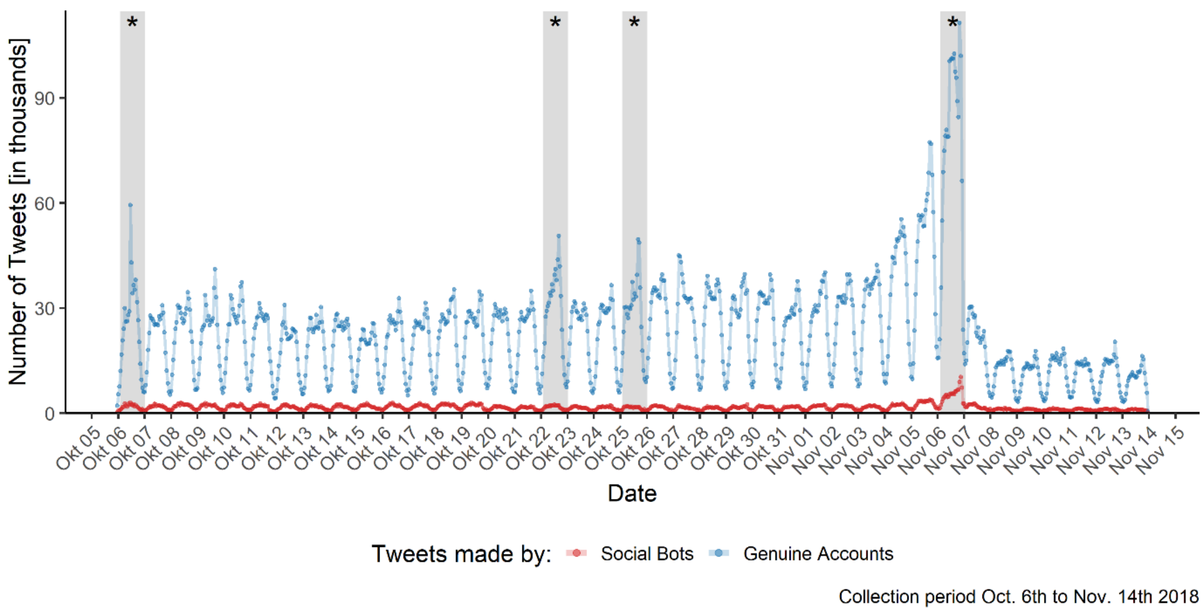

Figure 5: Timeline of the volume of tweets generated during the observation period by account type from accounts with more than 9 tweets in the data set. The grey areas marked with the asterisk show the three days where an event took place which led to higher activity on Twitter as well as the actual election day.

This tells us that social bots did not show any form of a planned takeover or tried to dominate the debate a few days ahead of the election. A fascinating detail in the figure is also the fact that right after the election the number of published tweets drops from 12’424 genuine tweets per hour during the day to only around half of that with 6’553 genuine tweets. The same thing is observable for the social bots where we see around 851 tweets per hour before and 563 tweets after the election[5]. This shows how short the attention span on social media is even for an event as big as a national election. This is contrary to the effect visible before the election. Right from the beginning, the number of tweets published in the data collection is already almost as high as shortly before the election. Only in the last five days before election day the number of tweets start to rise rapidly peaking at election day itself with 2.778 million tweets alone. This shows that social media doesn’t behave much different from the more classical media channels like television and newspaper, where most attention on the election is visible shortly before and after an election when the results are being discussed.

Lastly, one can determine three additional events that happened during the election campaign, which lead to an increase in tweets. The first event happened right at the beginning of my data collection on October 6th. This was the day Brett Michael Kavanaugh was confirmed to become a Supreme Court member by the Senate after weeks of intense hearings and protests by the Democrats and Women's rights movement against him becoming a Supreme Court judge, because of accusations of sexual assault were made public. On October 22nd activity on Twitter was also higher than normal due to the Trump administration revealing plans to include a legal definition of gender in the federal civil rights law. This plan attracted quite a lot attention in the LGBT community as well as the conservative and religious community. These groups had very contradicting opinions regarding that plan, which led to a slightly heated debate on Twitter. The last rise of activity on Twitter was a result of Trumps announcement to reduce the prices on American prescription drugs, which resulted in mixed statements on Twitter amongst all users. The

[5] The mean before the election is calculated with the data from October 6th up to the day before the election (November 6th) and for the mean after the election from November 8th to the end of the collection period (November 14th).

tweets included differing opinions on whether his plan would bring down the drug prices or not.

In a second stage, if possible, I assign tweets to either the Republican or the Democratic party, with hashtags which are clearly biased towards one of the two parties (see Appendix A). Out of this classification I could assign a total of 78.7 % of all tweets to belong to one of the two parties. The most important finding here is that only 16.7 % of all tweets are positively connected with the democratic party while 62 % are positively connected to the republican party. The other 21.3 % are either not favouring one of the two parties or could not be assigned since they contained hashtags favouring both parties.

In Figure 6, I visualize the number of tweets made per hour affiliated with either the democratic or republican party or neither one separated by tweets made by people and social bots. Figure 6 shows that the use of bots is visible in all three types of tweets, which demonstrates that apparently social bots are made not only to support both parties but also to just influence the general discussion. If we look closer at the activity of social bots, we can observe that the ratio between tweets made by social bots and genuine users is different for both the parties and the unaffiliated tweets before the election[6].

Social bots have the highest share in the unassignable group with a share of 13.5 % followed by a share of 7.9 % in the democratic group. The group with the smallest share of tweets made by social bots before the election are the tweets affiliated with the republican party with a share of only 4 %. In absolute numbers, this looks a bit different since there are far more tweets affiliated with the republican party as any other group. The republican bots are on top with an average of 702 tweets per day followed by the unaffiliated bots with 607 tweets tailed by the bots affiliated with the democratic party with 367 tweets. Furthermore, one can notice that the share of unassignable tweets spikes right around the election day. This is attributable to the fact that during these 24 hours a lot of people as well as social bots used hashtags like *#electionday*, *#midterms2018*, *#midterms* and *#election2018* to announce their participation in the election or to encourage others to go to vote as well.

[6] As before the mean number of tweets are calculated within the period from October the 6th to November the 5th. To make sure the results are not skewed by the election day.

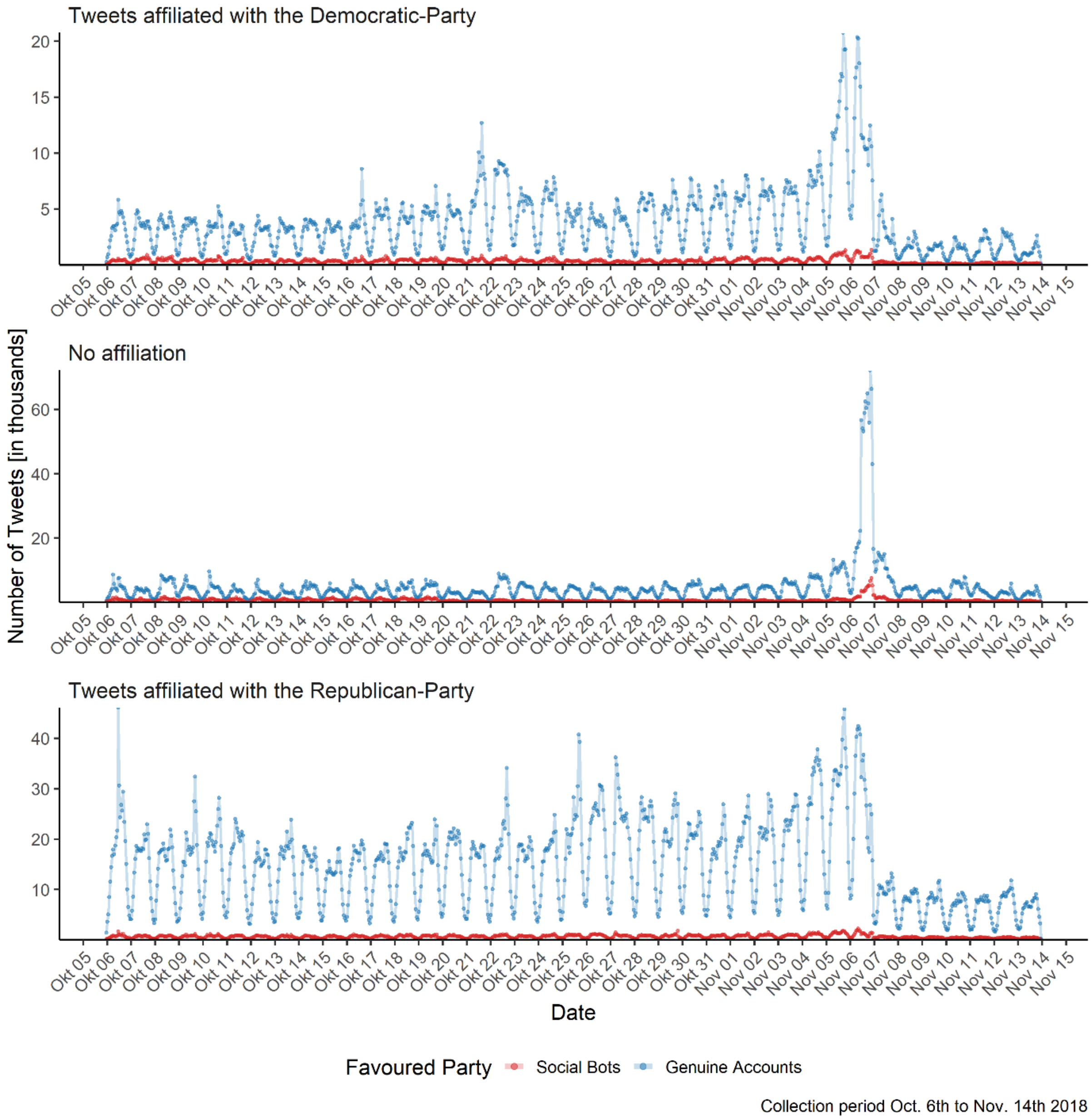

Figure 6: Timeline of the volume of tweets generated during the observation period faceted by party affiliation and separated by account type.

The peak of activity from the events described beforehand are no longer visible in all three separated groups. In the democratic group, the election day peak is reached a day before the election and not during the election day as seen before in the fully aggregated figure. Apart from this peak the publication numbers for the period before the election are unexciting except for October 21st. This was the day the Trump administration made public their wish to include a legal definition of gender in the federal civil rights law, which was of course not very popular within the democratic ranks. As for the tweets in the republican group, the peak right before and during the election day is not as distinct as for the other groups. The reason for the minor increase of tweets lies in the fact that the number of tweets was already very high for the republican group long before the election day itself. Furthermore, there are other events visible where there are almost as many tweets published for the election before the election day especially around the 6th and 25th of October, which were described before as days with important announcements by the government.

In a third step, I visualize the volume of tweets separating them into tweets with content intended for one of the 19 sampled districts of interest and tweets not directed at any of these districts and thus grouped as tweets directed at the national discourse. Figure 7 illustrates the difference between these two groups very broadly.

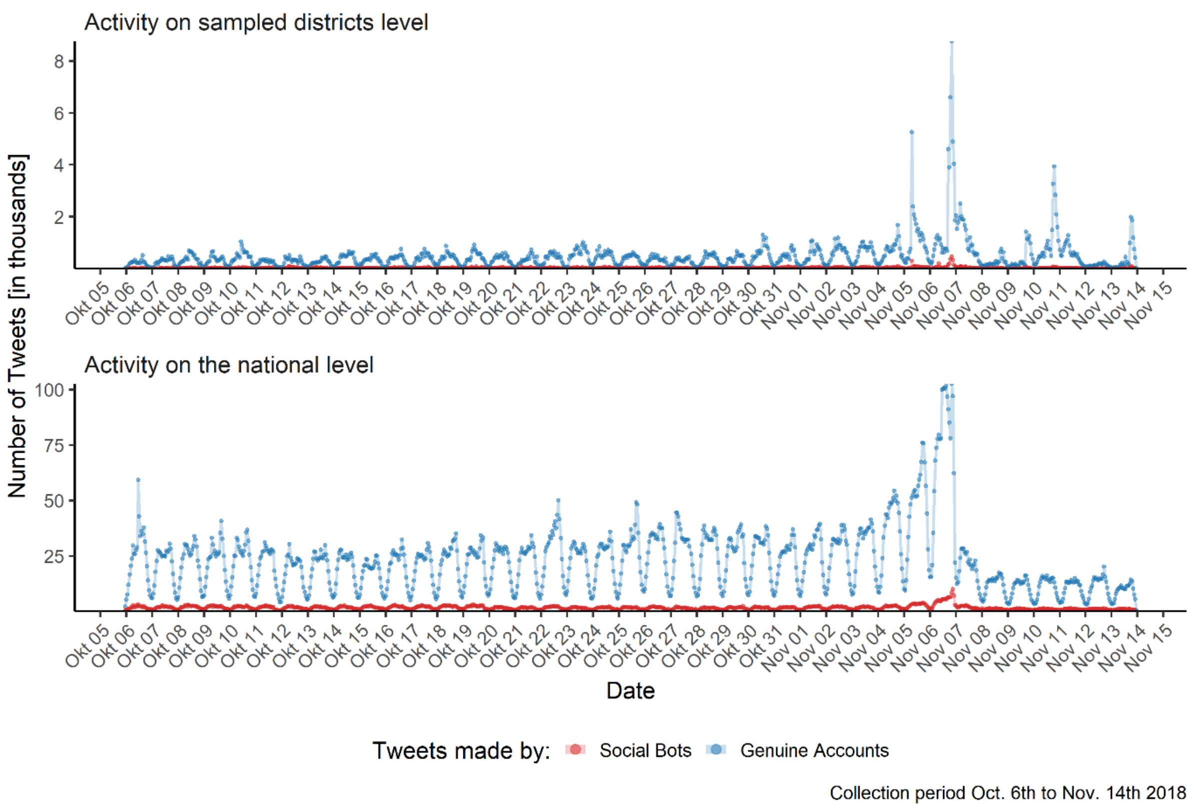

Figure 7: Timeline of the volume of tweets per hour generated during the observation period faceted by tweets made for districts and for the national campaign split by account type.

The key message is that the number of tweets made solely for one of the 19 districts is rather low in comparison to the rest of the tweets which are not directed at any district out of the sample or candidate. Put in numbers, the magnitude of tweets made by genuine accounts on the district side is 405 tweets per day compared to 21 tweets by social bots, which means that only about 5 % of all tweets published are made by social bots.

On the national level things look a bit different, considering that the share of tweets made by social bots is 6.3 % or 1'654 tweets against 24'444 tweets. The ratio of tweets made by either type of account does not change much during the election day as well as after the election except for the fact that for tweets directed at a district the social bot activity drops to a low of only 2.9 % after the election. The other interesting observation which isn't shown by the figures before is the fact that the number of tweets published targeting a district did not only rise at the election day itself, but rather peaked again after the election day in contrast to all other findings. This peak arose most likely because some of the sampled districts were having quite close races were the results didn't get announced by officials during the election night or in the morning after the election, but were rather reported later at November 10th, 11th and 12th which lead to quite a few tweets published in these districts to show excitement or frustration about the elected person in the district. For example, on November 10th the press called the race between Harley Rouda and Dana Rohrabacher in California's 48th district for the democratic candidate Rouda.

6.2 District Dialogue vs. National Dialogue

Apart from these descriptive figures of social bot activity during the general election campaign of 2018, I am highly interested in the effects of several campaigning features on social bot activity.

One of my main arguments states that social bots are less often used to tweet in regional campaigns than they are on national campaigns. To test this hypothesis, I use the logistic regression model. The main independent variable is called National which is dichotomous. Table 12 displays all six different models used to test the first hypothesis. Beginning from a model with just the independent variable, I use more and more control variables to see if the effect remains significant as well as if the model fit improves with the addition of new variables. The essence one can get out from the reported model fit measures is that the inclusion of control variables has no notable effect on the model fit. Adding control variables has only a minor effect on the AIC which gets gradually smaller with more control variables, whereas the loglikelihood gradually becomes a bit less negative. This means that the models are becoming gradually better the more complex they get, but only on a very small scale. The very high AIC and highly negative loglikelihood already indicate that the goodness of fit is not particularly well which indicates that although being significant the actual impact of the independent variable National Tweet cannot explain all the difference between the social bot activity. The loglikelihood ratio test in Table 11 reports that the full models observed difference is statistically significantly better from the models with less variables. The Wald-Test displays a similar picture. Since, the chi-squared test statistic of 71.0, with one degree of freedom is associated with a p-value of 0.0 indicating that the overall effect of national tweets is statistically significant. Hence the logistic regression is suitable for the problem at hand.

	Df.	Deviance	Resid. Df.	Resid. Dev.	Pr(> Chi)
NULL Model			23'682'304	11'384'688.0	
National	1	3'683.7	23'682'303	11'381'004.0	2.2e-16***
ln(Population)	1	78.7	23'682'302	11'380'925.0	2.2e-16***
Bachelor or higher	1	442.2	23'682'301	11'380'483.0	2.2e-16***
ln(Median income)	1	367.5	23'682'300	11'380'115.0	2.2e-16***
Median age	1	427.6	23'682'299	11'379'688.0	2.2e-16***
Urban pop. share	1	103.8	23'682'298	11'379'584.0	2.2e-16***

*Signif. Codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.'*

Table 11: Loglikelihood ratio test for model 6 with all covariates

In essence, model six shows that there is a clearly significant effect between the odds ratio of a tweet being made by a social bot if it is meant for the national campaign rather than a campaign for a single district (regional level). The level of significance is smaller than the 99.9 % with a p-value of almost zero. The model additionally shows that all control variables are showing significant effects. While higher shares of people with an education higher than a bachelor's degree and higher share of people living in urban areas have a positive impact on the odds ratio of a tweet being made by a social bot. All other variables have a negative effect. Hence higher population rates, higher median income and median age reduce the odds of a tweet being made by a social bot. But none is as high as the independent variable, which has an odds ratio many times bigger than the rest.

Logit analysis of the determinants of social bots' activity during the US. Midterm election of 2018

	Bot or Not					
	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6
Constant	-3.071***	-20.882***	-37.523***	-13.779***	23.905***	42.141***
	(0.007)	(2.006)	(2.110)	(2.485)	(3.158)	(3.601)
National Tweet	0.410***	-7.638***	-15.511***	-10.679***	5.926***	12.264***
	(0.007)	(0.906)	(0.959)	(1.002)	(1.333)	(1.455)
Log(Population)		1.319***	2.595***	1.798***	-0.931***	-1.976***
		(0.149)	(0.157)	(0.164)	(0.218)	(0.239)
Education			-0.016***	0.006***	0.011***	0.012***
			(0.001)	(0.001)	(0.001)	(0.001)
Log((Median Income))				-1.237***	-1.144***	-1.591***
				(0.066)	(0.064)	(0.076)
Median Age					-0.052***	-0.043***
					(0.003)	(0.003)
Urban Population Share						0.006***
						(0.001)
N	23'682'305	23'682'305	23'682'305	23'682'305	23'682'305	23'682'305
Log Likelihood	-5'690'502	-5'690'463	-5'690'241	-5'690'058	-5'689'844	-5'689'792
AIC	11'381'008	11'380'931	11'380'491	11'380'125	11'379'700	11'379'598

*p < .1; **p < .05; ***p < .01

Table 12: logistic regression models for hypothesis one. Model one shows the simple model with only the independent and dependent variable and model six shows the full model with all covariates

I will not go into too much detail how to read odds ratio since they are quite difficult to grasp. Consequently, I use a simulation to graphically display the difference of the share of tweets made by social bots between tweets made for the national campaign on Twitter and a tweet made for a hypothetical averaged district in the United States. Figure 8 shows the effect with a confidence interval of 99.9 %. Moreover all 10'000 simulations are plotted as well with their resulting values according to the model's coefficients. The difference in the error size between the two groups is due to the much higher number of tweets classified for the national campaign and much lower numbers of tweets classified for the 19 sampled districts. Nonetheless, the 462'636 tweets are more than enough to report a significant difference and to verify my hypothesis. There is an overall higher social bot activity on the national level than on the regional and local level (federal house-districts).

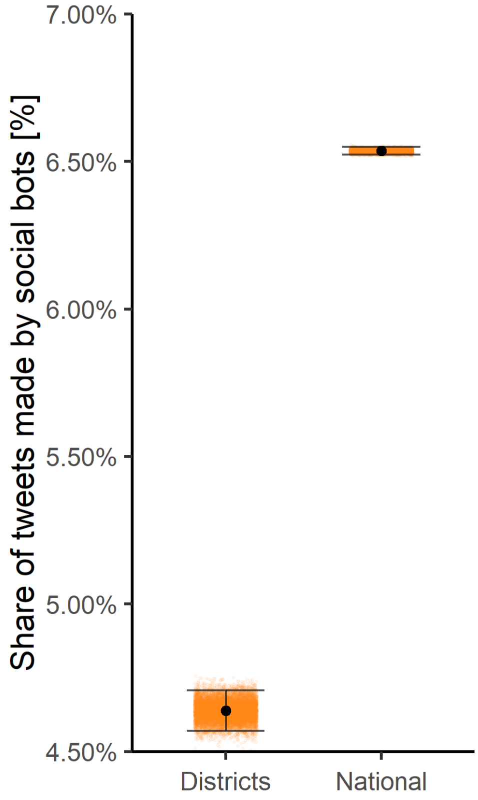

Figure 8: Difference in social bot activity between the national campaign and a hypothetical average districts campaign in the United States with error bars calculated by the obtained confidence interval of the logistic regression and the 10'000 simulated election campaigns as dots.

Finally, the plot demonstrates that the difference between the two groups in bot activity is 1.9 %. This might look like a small difference. but when we remember that the share of tweets made by social bots on the national level is only approximately 6.5 % and in the districts 4.6 % this difference is rather big with a change of around 41 % from the district level to the national level.

6.3 Campaigning Effects on Social Bots

The second hypothesis of my paper states that the more contested a district is before the election, the more activity in social bots we will see. This hypothesis is tested with a logistic regression again. The big difference is that now I have several different indicators, which I can rely on within the districts to see if the hypothesis can be statistically substantiated. The level of contestation is measured via Cooks partisan voter index and the closeness of the vote. To measure the intensity of the contestation I use a principal component analysis to construct one single variable out of the two variables that assess two distinct dimensions of contestation. The constructed contestation or intensity variable can measure 61 % of the variance, which is better than using only one variable to measure the campaign intensity. Additionally, I investigate what influence an incumbent in a race and the voter turnout in the State has on the social bot activity on Twitter. Further, the effects of the amount of money spent by both parties on the activity seen on Twitter by social bots in a district is measured. In Table 13, the most important models are listed. The full model with the independent variables and the control variables is significantly better than the rest of the model according to the Loglikelihood-Test, which displays that adding each variable improves the model slightly. The same is true for the AIC and the Wald-Test, which confirm that the model is positively significant for social bots. Thus, we can reject the null hypothesis.

Logit analysis of the determinants of social bots' activity during the US. Midterm election of 2018

	Bot or Not				
	Model 1	Model 2	Model 3	Model 4	Model 5
Constant	-3.075***	-2.954***	-3.994***	8.870***	20.035***
	(0.007)	(0.019)	(0.123)	(0.794)	(1.052)
Intensity	-0.089***	-0.070***	-0.096***	0.001	0.067***
	(0.006)	(0.007)	(0.007)	(0.010)	(0.010)
Incumbent		-0.142***	0.057*	0.022	-0.281***
		(0.021)	(0.032)	(0.034)	(0.038)
Turnout (State)			0.017***	-0.015***	-0.012***
			(0.002)	(0.003)	(0.003)
Log(Total Spending)				-0.697***	-0.417***
				(0.042)	(0.050)
Log(Median Income)					-1.354***
					(0.096)
Education					0.006***
					(0.001)
Median Age					-0.032***
					(0.002)
Urban Population Share					0.007***
					(0.001)
Observations	462,636	462,636	462,636	462,636	462,636
Log Likelihood	-83,846.560	-83,824.660	-83,786.970	-83,650.850	-83,136.120
Akaike Inf. Crit.	167,697.100	167,655.300	167,581.900	167,311.700	166,290.200
Note:					*p<0.1; **p<0.05; ***p<0.01

Table 13: Here we see the different logistic regression models with various amounts of additional variables to control for. The first model describes the simple model only looking at the effect of the independent variable on the dependant, whereas the following models include various other covariates like incumbency and turnout which are also very important

The effect of the intensity of a campaign is slightly positive but not very robust, since the models without demographic controls suggest that the correlation is negative, while the models using demographic controls all suggest a positive relationship. Hence the effect can only be interpreted with great caution. Nevertheless, model 5 demonstrates that campaigns with a higher intensity due to a close race and an undecided electorate show higher rates of tweets made by social bots, while the opposite is true for elections with almost no campaigning. Furthermore, the model clearly shows that an incumbent running for office has a negative effect on tweets made by social bots. This concurs with classical campaigning theory, which states that districts with a running incumbent use significantly less campaign spending form the challenger party. This seems to hold for social bots as well. Thus, incumbents seem to lower the overall influence of campaigning not only for classical campaign spending.

Surprisingly, the amount of funds spent on the campaigns has a negative effect on social bots. This indicates that the official campaign bureaus of the candidates seem to have no role in the game of making social bots. Because social bots are more likely to occur when the campaign lacks decent funding, because the chances of succeeding to influence the electorate are higher for bots if the official campaign messages are expected to be weak. This is a very good sign for democracies all over the world, since it shows that the funding for social bots are not high enough to change public opinion in small districts when the candidates lead a lively election campaign and thus aren't used as much. On the other hand, it also shows that money is a factor improving a candidate's chances for his or her election.

Figure 9 visually displays the effect the intensity of the campaign has on the share of tweets by social bots on Twitter for districts with average control variables overall important settings. The left graph shows the difference in the quantity of tweets made by social bots for different campaign intensities between districts with an incumbent running for office and no incumbent running for office. The graph in the middle shows the same thing for two different sums of spending by the parties of the candidates. This plot displays the significant positive effect money has on the campaigns share of tweets made by social bots. The left plot depicts the effect the voter turnout has on the share of social bots. Here we can see that the effect is in fact not entirely significant, since both confidence intervals start to overlap when we move away from the centre. This is due to the low number of different districts sampled which results in very narrow ranges where I have enough data to calculate narrow confidence intervals as seen in the boxplots under the graphs showing the effects.

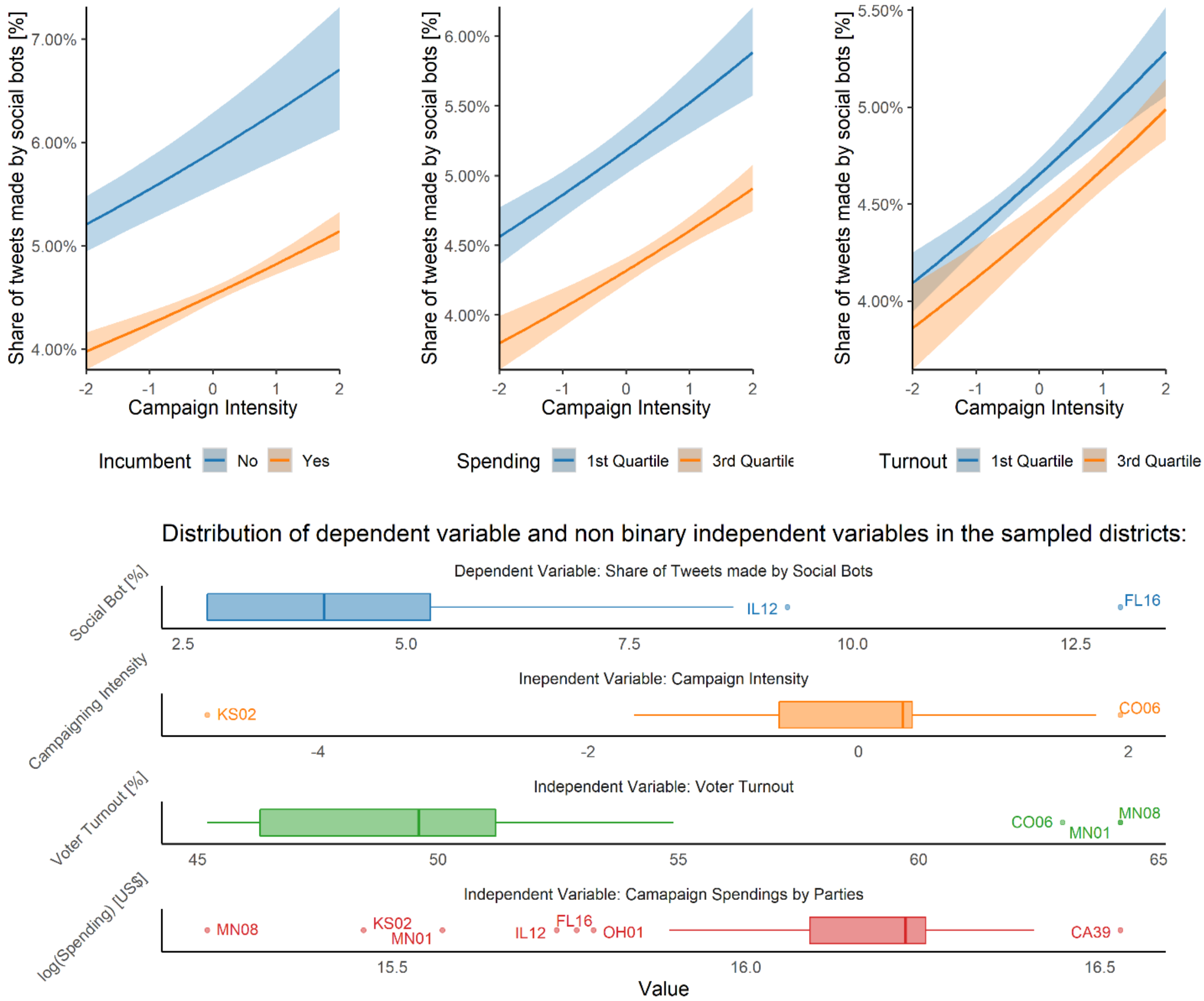

Figure 9: Effect of independent variable on share of tweets made by social bots between average districts with an incumbent running for office in cases where the incumbent is not used for the Monte Carlo simulation since this is more often the case in an election than the other way around where no incumbent is running for office again.

6.4 Comparing the Election 2018 with 2016 and 2014

The question if Twitter was able to reduce the number of tweets made by social bots since the last elections remains. As mentioned, after the 2016 presidential elections Twitter and many other social media platforms faced the bitter truth that their platforms were being misused by social bots and trolls to influence the political discourse and the result of the election. Under the pressure of parliament, the media and the public the platforms all pledged to tackle the problem.

Now did Twitter keep their promise or not? To be able to compare those results with former results, I use the midterm election of 2014 and the presidential election of 2016 as main comparison points. I use two comparison points since the 2014 election was a midterm election as well, but back then not many papers existed regarding the use of social bots on social media. Hence, to be able to compare with more results, I rely on the results found during the 2016 presidential election too. This is not optimal, because a presidential election is clearly different from a midterm election, nonetheless it is the only election where enough papers are available.

In 2014 Twitter reported to the US Securities and Exchange Commission that approximately 5 % to 8.5 % of their user base consists of social bots[7]. This estimate was found to be rather low. Varol et al. (2017) came to the conclusion that the overall share of bots is more likely to be around 9 % to 15 % and not as low as Twitter suggested to the US government back in 2014. This result seems to be more accurate as it has been confirmed by other authors as well. Bessi and Ferrara (2016) found that from the top 50k users in their data, 14,4 % were social bots according to their classifier. Kollanyi, Howard and Woolley (2016) classified tweets during the 2016 US presidential election and found that roughly 18 % of all posted tweets were made by highly automated accounts, which is pretty much the same result as was made by Bessi and Ferrara (2016), who found that 19 % of the tweets are made by social bots. Now during the collection period of the 2018 midterm election camapigns my classifier shows that roughly 8.5 % of all acounts with more than 10 tweets in my sample are social bots. These bots are responsible for 6.5 % of the tweets. This is in fact a result which is somewhat satisfying. It clearly shows that since 2016 the efforts made by Twitter to reduce the number of tweets made by social bots are showing positive results. If my classifer is not over- or underrepresenting the number of social bots, the numbers are about a third lower than back in 2016 and 2014. They are now as low as Twitter reported them to be in 2014 to the US government. Even if I use lower thresholds for my classifier, the numbers do not rise significantly. Hence I can say with confidence that the share of social bots has decreased in the last two years to a considerable degree and is most likely below 10% which is at least a

[7] https://www.sec.gov/Archives/edgar/data/1418091/000156459014003474/twtr-10q_20140630.htm

third less than two years ago and thus disproves my hyothesis of no improvment. Therefore, my data displays that Twitter is clearly doing something against social bots, but there is still a lot of work ahead for them to get rid of bots completely. The results show that the reportings by media seen during the 2018 midterm election stating that Twitter activiely closes down accounts violating their terms of conditions are having an impact on the number of tweets made with malicous content by bots (Bing 2018; dpa 2018; Selina Wang 2018). Therefore, my hypothesis can not be proven as I see a clear tendency showing less social bot activity in 2018 than in the past elections. This is a very positive result since it displays that things are chainging for the better on social media.

7 Conclusion

With the immense diffusion of social networks, social media has become one of the main channels of information dissemination not only for young people but for older generations too. Thus, social media are more central for political discussion than ever. Praised at the beginning for their contribution to democratization it was quickly realized that with this advantage mayor disadvantages arise as well. Social media platforms are liable to the spread of misinformation and to the automation of spreading information.

This led to a widespread outcry from the public by newspapers and congress after the 2016 election, since it undermined the democratic principles of opinion formation as well as the possible alteration of election outcomes. Hence lots of researchers investigated the problem and found that there are reasons to be worried and measures to be taken against this problem. My paper looks at the state of the measures taken by the platforms against the use of mischievous social bots. First, I look more closely at the use of social bots in the national and the regional election campaigns. Then I test if intensive races in districts lead to more bot activity and finally, I compare the use of social bots during the 2018 midterm elections with the 2016 presidential and the 2014 midterm elections. Beyond that I present a small framework for bot detection on Twitter. I introduce a deep learning architecture based on H_2O using only 33 features from an account and his tweets. The final classifier achieves a perfect true negative score of 100 % and a true positive score of 99.4 % with a 5-fold cross validation. It also shows that both meta user data as well as data from the tweet are central for the classification of an account, which results in the limitation that my classifier needs several tweets from an account to classify an account correctly.

Generally, with the classification of the data I can demonstrate that social bots are still actively present in the online political discussion around the midterm elections not only on the national level, but also when it comes to campaigning content especially made for districts. But here we also see that the level of activity is significantly lower on the local level. This shows that in local elections there is much less malicious content to be found and thus the discussions are more likely to reflect the actual discourse. In the national discourse the content has to be viewed more often with caution as it may be that the information seen is invalid or at least it should not be weighted as much since it is not seen as important by the population but only by a small group of people spreading the information through social bots.

This can lead to substantial issues as demonstrated in other works, like the redistribution of influence towards accounts with malicious purposes, an increased polarization of the conversation or the enhanced spread of misinformation. Hence it is important to reduce the number of social bots, which was promised by many social media platforms including Twitter. Therefore, I compare the share of social bots found in my research with older results from 2016 and 2014. The results show that efforts are being made but more has to be done to reduce the number of social bots to a degree low enough to make sure people will most likely never see content shared or made by them. Especially, for the national discourse it is necessary to put more effort in the reduction of social bots since there the activity is still high enough to potentially influence the democratic opinion formation process.

Furthermore, the observed districts show that intensive election campaigns have a higher risk of being influenced by social bots, particularly when there is no incumbent running for office. This result is rather upsetting since this is often the case for executive elections in many other countries. This information must be viewed with much caution since chances are high that misinformation is spread not only by the bots but also by real actors falling for the misinformation. Nevertheless, the results in this work only give a direction of the effects. The results indicate that more work must be put into the question of what promotes the use of social bots before elections and when. Especially since the analysis shows that the results are not very robust although being significant.

Concluding, it is important to stress that my analysis could show that there are differences in social bot use depending on the scope and state of the campaign. However, I cannot say with absolute certainty if the effects are as big or small as reported due to lack in robustness of the results. Furthermore, there still needs to be much work done to tackle the problem of determining who is operating social bots and what their objectives are. Especially because the use of social bots still is a problem on social media even after the active countermeasures made by the operators of social media sites. Therefore, more research is required by social scientists and the machine learning research community to analyse social bot networks in detail and so to unveil the masterminds behind social bots and to stop them once and for all.

8 References

Abokhodair, Norah, Daisy Yoo, and David W. McDonald. 2015. "Dissecting a Social: Growth, Content and Influence in Twitter."." In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing.*, ACM, 839–51.

Abramowitz, Alan I. 1991. "Incumbency, Campaign Spending, and the Decline of Competition in United-States House Elections." *Journal of Politics* 53(1): 34–56. https://www.jstor.org/stable/pdf/2131719.pdf.

Allcott, Hunt et al. 2018. "Trends in the Diffusion of Misinformation on Social Media." *arXiv preprint* arXiv:1809: 1–12. http://web.stanford.edu/~gentzkow/research/fake-news-trends.pdf.

Allcott, Hunt, and Matthew Gentzkow. 2017. "Social Media and Fake News in the 2016 Election." *Journal of Economic Perspectives* 31(2): 211–36. http://pubs.aeaweb.org/doi/10.1257/jep.31.2.211.

Ansolabehere, Stephen, James M. Snyder, and Charles Stewart. 2000. "Old Voters, New Voters, and the Personal Vote: Using Redistricting to Measure the Incumbency Advantage." *American Journal of Political Science* 44(1): 17–34. https://www.jstor.org/stable/2669290.

———. 2001. "Candidate Positioning in U.S. House Elections." *American Journal of Political Science* 45(1): 136–59. https://www.jstor.org/stable/2669364.

Ansolabehere, Stephen, and James M. Snyder Jr. 2002. "The Incumbency Advantage in U.S. Elections: An Analysis of State and Federal Offices, 1942–2000 STEPHEN." *Election Law Journal* 1(3): 315–38. https://www.liebertpub.com/doi/pdf/10.1089/153312902760137578.

Badawy, Adam, Kristina Lerman, and Emilio Ferrara. 2018. "Who Falls for Online Political Manipulation?" *arXiv preprint* arXiv:1808. http://arxiv.org/abs/1808.03281.

Ban, Pamela, Elena Llaudet, and James M. Snyder. 2016. "Challenger Quality and the Incumbency Advantage." *Legislative Studies Quarterly* 41(1): 153–79.

Berger, Jonathon M, and Jonathon Morgan. 2015. "The ISIS Twitter Census: Defining and Describing the Population of ISIS Supporters on Twitter." *The Brookings Project on US Relations with the Islamic World* 3(20): 1–4.

Bergstra, James, and Umontrealca Yoshua Bengio. 2012. "Random Search for HyperParameter Optimization." *Journal of Machine Learning Research* 13: 281–305.

Bessi, Alessandro et al. 2015. "Science vs Conspiracy: Collective Narratives in the Age of Misinformation." *PLOS ONE* 10(2): 1–17.

Bessi, Alessandro, and Emilio Ferrara. 2016. "Social Bots Distort the 2016 U.S. Presidential Election Online Discussion." *First Monday* 21(11): 1–15. http://uncommonculture.org/ojs/index.php/fm/rt/printerFriendly/7090/5653.

Bing, Christoper. 2018. "Exclusive: Twitter Deletes over 10,000 Accounts That Sought to Discourage U.S. Voting | Reuters." *Reuters*. https://www.reuters.com/article/us-usa-election-twitter-exclusive/exclusive-twitter-deletes-over-10000-accounts-that-sought-to-discourage-u-s-voting-idUSKCN1N72FA (November 3, 2018).

Boehmke, Bradley. 2017. "Artificial Neural Network Fundamentals · UC Business Analytics R Programming Guide." http://uc-r.github.io/ann_fundamentals (May 18, 2019).

Chavoshi, Nikan, Hossein Hamooni, and Abdullah Mueen. 2016. "DeBot: Twitter Bot Detection via Warped Correlation." In *Proceedings - IEEE International Conference on Data Mining*, ICDM, 817–22.

Cheng, Bing, D M Titterington, Bing Cheng, and D M Titterington. 1994. "Neural Networks: A Review from a Statistical Perspective." *Statistical Science* 9(1): 2–30.

https://www.jstor.org/stable/2246275.

Ciaburro, Giuseppe, and Balaji Ventkateswaran. 2017. *Neural Networks with R*. 1st ed. ed. Sunith Shetty. Birmingham UK: Packt.

Cook, Darren. 2016. *Practical Machine Learning with H2O: Powerful, Scalable Techniques for Deep Learning and AI*. " O'Reilly Media, Inc."

Cox, Gary W., and Jonathan N. Katz. 1995. *WHY DID THE INCUMBENCY ADVANTAGE IN U.S. HOUSE ELECTIONS GROW?* http://resolver.caltech.edu/CaltechAUTHORS:20140314-120455847.

Cresci, Stefano et al. 2017. "The Paradigm-Shift of Social Spambots: Evidence, Theories, and Tools for the Arms Race." In *Proceedings of the 26th International Conference on World Wide Web Companion.*, International World Wide Web Conferences Steering Committee, 963–72. http://arxiv.org/abs/1701.03017%0Ahttp://dx.doi.org/10.1145/3041021.3055135.

———. 2018. "Social Fingerprinting: Detection of Spambot Groups Through DNA-Inspired Behavioral Modeling." *IEEE Transactions on Dependable and Secure Computing* 15(4): 561–76.

Davis, Clayton A. et al. 2016. "BotOrNot: A System to Evaluate Social Bots." In *Proceedings of the 25th International Conference Companion on World Wide Web.*, International World Wide Web Conferences Steering Committee, 273–74. http://arxiv.org/abs/1602.00975%0Ahttp://dx.doi.org/10.1145/2872518.2889302.

Deb, Ashok et al. 2018. "Social Bots for Online Public Health Interventions." *Proceedings of the 2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM 2018*: 186–89.

dpa. 2018. "Twitter Löscht Vor Den US- Kongresswahlen 10 000 Gefälschte Accounts, Die Demokraten Vom Wählen Abhalten Sollten | NZZ." *Neue Zürcher Zeitung*. https://www.nzz.ch/international/twitter-loescht-vor-den-amerikanischen-kongresswahlen-10-000-gefaelschte-accounts-die-demokraten-vom-waehlen-abhalten-sollten-ld.1433572 (November 3, 2018).

Erikson, Robert S. 1971. "The Advantage of Incumbency in Congressional Elections." *Polity* 3(3): 395–405.

"FEC Campaign Finance Data." 2019. *Federal Election Commission.*

https://www.fec.gov/data/candidates/house/ (March 16, 2019).

Ferrara, Emilio et al. 2014. "The Rise of Social Bots." *Communications of the ACM* 59(7): 96–104. http://arxiv.org/abs/1407.5225%0Ahttp://dx.doi.org/10.1145/2818717.

Gaudart, Jean, Bernard Giusiano, and Laetitia Huiart. 2004. "Comparison of the Performance of Multi-Layer Perceptron and Linear Regression for Epidemiological Data." *Computational Statistics & Data Analysis* 44(4): 547–70. https://doi.org/10.1016/S0167-9473(02)00257-8.

Gelman, Andrew, and Gary King. 1990. "Estimating Incumbency Advantage without Bias." *American Journal of Political Science* 34(4): 1142–64. https://ssrn.com/abstract=1084180.

Gerber, Alan. 1998. "Estimating the Effect of Campaign Spending on Senate Election Outcomes Using Instrumental Variables." *American Political Science Review* 92(02): 401–11. http://www.journals.cambridge.org/abstract_S0003055400214007.

Géron, Aurélien. 2017. *Hands-on Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. " O'Reilly Media, Inc."

Hegelich, Simon, and Dietmar Janetzko. 2016. "Are Social Bots on Twitter Political Actors? Empirical Evidence from a Ukrainian Social Botnet." In *Proceedings of the Thenth AAAI Conferences on Web and Social Media*, AAAI, 579–82. https://www.aaai.org/ocs/index.php/ICWSM/ICWSM16/paper/viewFile/13015/12793.

Howard, Philip N et al. 2011. Project on Information Technologu and Political Islam *What Was the Role of Social Media during the Arab Spring ?* http://papers.ssrn.com/sol3/papers.cfm?abstract_id=2595096.

Internetlivestats.com. "Twitter Usage Statistics - Internet Live Stats." http://www.internetlivestats.com/twitter-statistics/ (September 19, 2018).

Kandori, Michihiro. 1992. "Social Norms and Community Enforcement." *The Review of Economic Studies* 59(1): 63. https://academic.oup.com/restud/article-lookup/doi/10.2307/2297925.

Kelleher, John D, Brian Mac Namee, and Aoife D'Arcy. 2015. *Fundamentals of Machine Learning for Predictive Data Analytics: Algorithms, Worked Examples, and Case Studies*. MIT Press.

Kollanyi, Bence, Philip N. Howard, and Samuel C. Woolley. 2016. "Bots and Automation over Twitter during the First U.S. Election." *Comprop data memo*: 1–5. http://comprop.oii.ox.ac.uk/wp-content/uploads/sites/89/2016/11/Data-Memo-US-Election.pdf.

Krebs, Timothy B. 1998. "The Determinants of Candidates' Vote Share and the Advantages of Incumbency in City Council Elections." *American Journal of Political Science* 42(3): 921–35.

Kudugunta, Sneha, and Emilio Ferrara. 2018. "Deep Neural Networks for Bot Detection." *Information Sciences* 467: 312–22. https://doi.org/10.1016/j.ins.2018.08.019.

Lee, Kyumin, Brian David Eoff, and James Caverlee. 2011. "Seven Months with the Devils: An in-Depth Characterisation of Bots and Humans on Twitter." In *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media*, AAAI, 185–92. http://arxiv.org/abs/1704.01508.

Li, Han, Jie Zhang, and Rathindra Sarathy. 2010. "Understanding Compliance with Internet Use Policy from the Perspective of Rational Choice Theory." *Decision Support Systems* 48(4): 635–45. http://dx.doi.org/10.1016/j.dss.2009.12.005.

MCDonald, Michael P. 2018. "2018 November General Election Turnout Rates." *United States Election Project*. http://www.electproject.org/2018g (March 16, 2019).

Menczer, Filippo et al. 2018. "The Spread of Low-Credibility Content by Social Bots." *Nature Communications* 9(1). http://dx.doi.org/10.1038/s41467-018-06930-7.

Mitchell, T.M. 2010. "Artificial Neural Networks." http://www.cs.cmu.edu/~epxing/Class/10701-10s/Lecture/lecture7.pdf.

"My Congressional District." 2019. *US Census Bureau*. https://www.census.gov/mycd/?st=17&cd=12 (March 16, 2019).

Peng, Chao Ying Joanne, Kuk Lida Lee, and Gary M. Ingersoll. 2002. "An Introduction to Logistic Regression Analysis and Reporting." *Journal of Educational Research* 96(1): 3–14.

Ratkiewicz, Jacob et al. 2010. "Detecting and Tracking the Spread of Astroturf Memes in Microblog Streams." In *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media Detecting*, AAAI, 297–304. http://arxiv.org/abs/1011.3768%0Ahttp://dx.doi.org/10.1145/1963192.1963301.

Shirky, Clay. 2011. "The Political of Social Power Media the Public Sphere , and Political

Change Technology ,." *Council on Foreign Relations* 90(1): 28–41. https://www.jstor.org/stable/25800379.

Sietsma, Jocelyn, and Robert J.F. Dow. 1991. "Creating Artificial Neural Networks That Generalize." *Neural Networks* 4(1): 67–79.

Smith, Aaron, and Monica Anderson. 2018. "Social Media Use in 2018." *Pew Research Center* (March): 1–17. http://www.pewinternet.org/2018/03/01/social-media-use-in-2018/.

Srivastava, Nitish et al. 2014. "Dropout: A Simple Way to Prevent Neural Networks from Overfitting Nitish." *Journal of Machine Learning Research* 15: 1929–58.

Stella, Massimo, Emilio Ferrara, and Manlio De Domenico. 2018. "Bots Sustain and Inflate Striking Opposition in Online Social Systems." 115(49): 12435–40. http://arxiv.org/abs/1802.07292%0Ahttp://dx.doi.org/10.1073/pnas.1803470115.

Varol, Onur et al. 2017. "Online Human-Bot Interactions: Detection, Estimation, and Characterization." *arXiv preprint* arXiv:1703. http://arxiv.org/abs/1703.03107.

Wang, Selina. 2018. "Twitter Ramps Up Fight Against Abuse and Malicious Bots - Bloomberg." *Bloomberg*. https://www.bloomberg.com/news/articles/2018-06-26/twitter-ramps-up-fight-against-abuse-and-malicious-bots (November 3, 2018).

Wang, Shuting, Min-Seok Pang, and Paul P Pavlou. 2018. "Cure or Poison ? Impact of Identity Verification on the Creation of Fake Posts on Social Media." *SSRN Electronic Journal*: 1–17. https://ssrn.com/abstract=3249479.

Wasserman, David, and Ally Flinn. 2017. "Introducing the 2014 Cook Political Report Partisan Voter Index." *Cook Political Report*.

Yang, Kai-Cheng et al. 2019. "Arming the Public with Artificial Intelligence to Counter Social Bots." *Human Behavior and Emerging technologies* 1(1): 48–61. http://arxiv.org/abs/1901.00912%0Ahttp://dx.doi.org/10.1002/hbe2.115.

Zhang, Guoqiang, Eddy Patuwo, and Michael Y. Hu. 1998. "Forecasting with Artificial Neural Networks: The State of the Art." *International Journal of Forecasting* 14(1): 35–62. https://www.sciencedirect.com/science/article/pii/S0169207097000447.

9 Appendix

9.1 Appendix A (Hashtag list)

9.1.1 Election Hashtags

Hashtag	# Tweets	Hashtag	# Tweets	Hashtag	# Tweets
#midtermelection2018	87’702	#midterms	715’633	#midterms2018	1’208’776
#election2018	596’370	#electionday	1’374’120	#trump	2’799’052
#congressman	2’076	#congressmen	279	#congresswoman	432
#congresswomen	0	#registertovote	82’466	#yourvoteisyourvoice	8’074

9.1.2 Pro Democratic Hashtags

Hashtag	# Tweets	Hashtag	# Tweets	Hashtag	# Tweets
#democrats	1’031’841	#voteblue	2’662’194	#bluewave	1’369’258
#masa	21’825	#makeamericssmartagain	0	dumptrump	0
#demswork4usa	38’674	#yeswecan	47’701	#flipitblue	185’044
#democrats2018	248	#winblue	45’520	#flipthehouse	158’881
#redtoblue	102’030	#takeitback	0		

9.1.3 Pro Republican Hashtags

Hashtag	# Tweets	Hashtag	# Tweets	Hashtag	# Tweets
#republicans	311’151	#votered	2’999’599	#gop	635’287
#maga	9’855’259	#makeamericagreatagain	213’034	#redtsunami	0
#greatawakening	287’361	#unhinged	20’856	#walkawaycampaign	41’464
#republicans2018	84	#democratspartyofhate	10’841	#liberalmob	2’763
#keepitred	1’804	#remainred	0	walkaway	3’340’109

9.1.4 Hashtags related to a District

District	Hashtag	# Tweets	District	Hashtag	# Tweets
California District 10:	#CA10	32’203	Florida District 16	#FL16	2’598
	#JeffDenham	113		#VernBuchanan	47
	#JoshHarder	816		#DavidShapiro	0

	#DoNothingAngryDenham	0		#redtidevern	1
	#DefeatDenham	0		#VernsYacht	138
	#DumpDenham	995	Florida District 18	#FL18	19'452
	#BayAreaHarder	161		#TeamBaer	1’515
California District 25	#CA25	29’823		#BrianMast	79
	#SteveKnight	435		#laurenbaer	128
	#KatieHill	1’192		#BrianMastIsToxic	346
	#gethilltothehill	0		#TeamMast	0
	#knightou	308		#MastforCongress	1
	#VoteKnightOut	8	Texas District 07	#TX07	18’083
	#VoteForKnight	1		#CongCulberson	2
California District 39	#CA39	28’826		#JohnCulberson	646
	#YoungKimCD39	5		#TeamLizzie	4’035
	#TeamYoung	513		#Lizzie4Congress	82
	#GillCisnerosCA	0		#VoteOutCulberson	5
	#FlipOCBlue	56		#VoteCulberson	11
	#GilontheHill	2’073	Texas District 32	#TX32	26’042
	#cisneros4congress	0		#TeamCollin	4
California District 48	#CA48	140’736		#collinallred	29
	#DanaRohrabacher	5’299		#collinallredTX	25
	#HarleyRouda	1’898		#PeteSessions	494
	#TeamHarley	492		#VotePeteSessions	18
	#Rohrabacher	3’827	Texas District 23	#TX23	24’416
	#VoteHarley	158		#WillHurd	72
	#HireHarley	122		#GinaJones	5
	#VoteForDana	6		#TX23forGina	8
Colorado District 06	#CO06	11’100		#Gina2018	8
	#MikeCoffman	311		#HurdforCongress	36
	#JasonCrow	209		#hurdonthehill	9
	#TeamCrow	162	Illinois District 12	#IL12	10’633
	#VoteCrow	25		#KellyCoalition	2’127
	#TeamCoffman	1		#RepBost	130
	#VoteCoffman	7		#FlipIL12	1
New York District 19	#NY19	60’728		#VoteKelly	249

	#DelgadoforNY19	34		#MikeBost	25
	#AntonioDelgado	617	Kansas District 02	#KS02	13’821
	#JohnFasoNY	5		#SteveWatkins	107
	#TeamFaso	699		#Steve4Kansas	22
	#FireFaso	4’647		#PaulDavis	167
New York District 22	#NY22	30’313		#PaulDavisKS	2
	#ABrindisiNY	1	Kansas District 03	#KS03	50’467
	#ClaudiaTenney	164		#KevinYoder	378
	#BrindisiforCongress	79		#ShariceDavids	7’832
	#OneTermTenney	938		#YoderVoter	1’164
	#TenneyforCongress	3		#VoteDavids	11
Minnesota District 01	#MN01	19’209	Ohio district 12	#OH12	26’560
	#TeamFeehan	505		#TroyBalderson	91
	#DanFeehan	182		#DannyOConnor	0
	#DanielFeehan	6		#VoteOConnor	16
	#JimHagedornMN	0		#VoteBalderson	2
	#Jimhegedorn	19	Ohio District 01	#OH01	12’492
Minnesota District 08	#MN08	29’188		#SteveChabot	638
	#PeteStauber	107		#AftabPureval	102
	#JoeRadinovich	117		#Chabotage	710
	#VoteStauber	1		#TeamAftab	1’095
	#JoeRadical	0		#TeamChabot	5
				#VoteChabot	13
				#VoteAftab	841

9.2 Appendix B (Model Performance Table)

Epochs	Layer Size	Hidden Dropout Ratios	Input Dropout Ratio	l1	l2	Stopping Tolerance	Model ID	Logloss
2004.7	[25, 5]	[0.3, 0.3]	0.15	1.00E-08	1.00E-08	0.02	77	0.0474
2004.7	[25, 5]	[0.3, 0.3]	0.15	1.00E-08	1.00E-07	0.03	42	0.0487
2004.7	[25, 5]	[0.3, 0.3]	0.1	1.00E-07	1.00E-08	0.03	118	0.0487
2004.7	[25, 5]	[0.4, 0.4]	0.1	1.00E-07	1.00E-07	0.03	89	0.0490
2004.7	[30, 5]	[0.4, 0.4]	0.15	1.00E-06	1.00E-07	0.04	109	0.0492
3004.7	[30, 5]	[0.4, 0.4]	0.2	1.00E-07	1.00E-07	0.04	28	0.0495

2004.7	[30, 5]	[0.4, 0.4]	0.1	1.00E-08	1.00E-06	0.03	3	0.0497
3004.7	[25, 5]	[0.3, 0.3]	0.2	1.00E-06	1.00E-06	0.02	15	0.0504
2004.7	[30, 5]	[0.3, 0.3]	0.2	1.00E-06	1.00E-07	0.02	58	0.0507
3004.7	[30, 5]	[0.4, 0.4]	0.1	1.00E-08	1.00E-06	0.04	12	0.0509
3004.7	[30, 5]	[0.4, 0.4]	0.2	1.00E-07	1.00E-07	0.02	56	0.0511
3004.7	[30, 5]	[0.4, 0.4]	0.2	1.00E-06	1.00E-08	0.04	27	0.0511
3004.7	[25, 5]	[0.3, 0.3]	0.1	1.00E-06	1.00E-07	0.02	119	0.0513
2004.7	[25, 5]	[0.3, 0.3]	0.1	1.00E-08	1.00E-06	0.03	60	0.0513
2004.7	[25, 5]	[0.3, 0.3]	0.2	1.00E-07	1.00E-07	0.02	26	0.0514
2004.7	[25, 5]	[0.4, 0.4]	0.15	1.00E-07	1.00E-06	0.02	108	0.0517
2004.7	[25, 5]	[0.4, 0.4]	0.2	1.00E-08	1.00E-07	0.02	50	0.0521
2004.7	[30, 5]	[0.4, 0.4]	0.1	1.00E-06	1.00E-06	0.03	84	0.0521
3004.7	[30, 5]	[0.3, 0.3]	0.15	1.00E-07	1.00E-06	0.02	31	0.0523
2004.7	[25, 5]	[0.3, 0.3]	0.15	1.00E-07	1.00E-07	0.03	2	0.0524
2004.7	[25, 5]	[0.4, 0.4]	0.1	1.00E-07	1.00E-06	0.02	82	0.0524
3004.7	[25, 5]	[0.3, 0.3]	0.2	1.00E-07	1.00E-06	0.02	70	0.0525
2004.7	[30, 5]	[0.3, 0.3]	0.2	1.00E-08	1.00E-07	0.03	22	0.0528
2004.7	[25, 5]	[0.4, 0.4]	0.2	1.00E-07	1.00E-08	0.04	92	0.0528
3004.7	[30, 5]	[0.4, 0.4]	0.1	1.00E-08	1.00E-08	0.04	76	0.0529
2004.7	[25, 5]	[0.3, 0.3]	0.15	1.00E-06	1.00E-07	0.03	36	0.0533
3004.7	[25, 5]	[0.3, 0.3]	0.1	1.00E-07	1.00E-06	0.04	39	0.0533
3004.7	[25, 5]	[0.4, 0.4]	0.1	1.00E-06	1.00E-07	0.04	16	0.0537
2004.7	[30, 5]	[0.3, 0.3]	0.15	1.00E-07	1.00E-06	0.02	64	0.0539
2004.7	[30, 5]	[0.4, 0.4]	0.1	1.00E-06	1.00E-08	0.03	114	0.0539
2004.7	[30, 5]	[0.4, 0.4]	0.1	1.00E-08	1.00E-07	0.04	11	0.0539
3004.7	[30, 5]	[0.4, 0.4]	0.1	1.00E-07	1.00E-08	0.02	32	0.0539
3004.7	[25, 5]	[0.4, 0.4]	0.15	1.00E-06	1.00E-07	0.04	122	0.0540
3004.7	[30, 5]	[0.3, 0.3]	0.2	1.00E-08	1.00E-08	0.02	112	0.0542
2004.7	[25, 5]	[0.3, 0.3]	0.1	1.00E-07	1.00E-08	0.02	23	0.0543
3004.7	[25, 5]	[0.3, 0.3]	0.15	1.00E-06	1.00E-06	0.03	74	0.0545
2004.7	[30, 5]	[0.3, 0.3]	0.1	1.00E-08	1.00E-08	0.03	17	0.0549
2004.7	[25, 5]	[0.3, 0.3]	0.2	1.00E-06	1.00E-06	0.04	66	0.0550
2004.7	[30, 5]	[0.4, 0.4]	0.15	1.00E-06	1.00E-08	0.03	43	0.0553
3004.7	[30, 5]	[0.4, 0.4]	0.2	1.00E-06	1.00E-07	0.03	97	0.0557
2004.7	[25, 5]	[0.3, 0.3]	0.1	1.00E-08	1.00E-08	0.03	35	0.0557
2004.7	[25, 5]	[0.3, 0.3]	0.1	1.00E-06	1.00E-07	0.02	57	0.0559
2004.7	[30, 5]	[0.3, 0.3]	0.2	1.00E-08	1.00E-06	0.03	73	0.0561
2004.7	[25, 5]	[0.4, 0.4]	0.2	1.00E-06	1.00E-07	0.04	9	0.0561
3004.7	[30, 5]	[0.4, 0.4]	0.15	1.00E-08	1.00E-06	0.04	53	0.0561
3004.7	[25, 5]	[0.4, 0.4]	0.15	1.00E-06	1.00E-07	0.03	38	0.0563
2004.7	[25, 5]	[0.3, 0.3]	0.15	1.00E-06	1.00E-06	0.04	5	0.0566
3004.7	[25, 5]	[0.4, 0.4]	0.2	1.00E-07	1.00E-08	0.04	49	0.0567

3004.7	[25, 5]	[0.4, 0.4]	0.2	1.00E-06	1.00E-07	0.04	14	0.0568
2004.7	[30, 5]	[0.3, 0.3]	0.1	1.00E-07	1.00E-07	0.03	95	0.0569
2004.7	[30, 5]	[0.3, 0.3]	0.15	1.00E-07	1.00E-06	0.03	79	0.0570
2004.7	[30, 5]	[0.3, 0.3]	0.2	1.00E-08	1.00E-08	0.02	13	0.0575
2004.7	[25, 5]	[0.3, 0.3]	0.2	1.00E-07	1.00E-07	0.04	65	0.0576
3004.7	[30, 5]	[0.4, 0.4]	0.15	1.00E-08	1.00E-08	0.04	105	0.0579
2004.7	[25, 5]	[0.4, 0.4]	0.2	1.00E-06	1.00E-08	0.02	7	0.0582
2004.7	[25, 5]	[0.4, 0.4]	0.1	1.00E-08	1.00E-07	0.02	106	0.0582
2004.7	[30, 5]	[0.3, 0.3]	0.1	1.00E-06	1.00E-07	0.02	94	0.0587
2004.7	[30, 5]	[0.3, 0.3]	0.2	1.00E-06	1.00E-07	0.04	29	0.0590
3004.7	[30, 5]	[0.3, 0.3]	0.1	1.00E-07	1.00E-07	0.02	46	0.0594
2004.7	[25, 5]	[0.4, 0.4]	0.15	1.00E-08	1.00E-07	0.03	83	0.0596
2004.7	[30, 5]	[0.4, 0.4]	0.15	1.00E-06	1.00E-07	0.02	81	0.0609
3004.7	[25, 5]	[0.4, 0.4]	0.1	1.00E-08	1.00E-08	0.03	103	0.0616
2004.7	[25, 5]	[0.3, 0.3]	0.1	1.00E-06	1.00E-06	0.03	98	0.0618
2004.7	[30, 5]	[0.3, 0.3]	0.15	1.00E-06	1.00E-08	0.02	54	0.0628
3004.7	[25, 5]	[0.3, 0.3]	0.1	1.00E-06	1.00E-06	0.04	61	0.0634
3004.7	[25, 5]	[0.3, 0.3]	0.15	1.00E-06	1.00E-08	0.04	104	0.0716
2004.7	[30, 3]	[0.3, 0.3]	0.2	1.00E-07	1.00E-06	0.04	126	0.0746
3004.7	[30, 3]	[0.3, 0.3]	0.2	1.00E-06	1.00E-08	0.03	107	0.0755
2004.7	[30, 3]	[0.3, 0.3]	0.2	1.00E-06	1.00E-07	0.04	24	0.0763
2004.7	[25, 3]	[0.3, 0.3]	0.1	1.00E-06	1.00E-06	0.04	110	0.0776
3004.7	[25, 3]	[0.3, 0.3]	0.2	1.00E-07	1.00E-08	0.04	8	0.0777
3004.7	[30, 3]	[0.3, 0.3]	0.15	1.00E-08	1.00E-08	0.03	75	0.0778
3004.7	[30, 3]	[0.3, 0.3]	0.2	1.00E-08	1.00E-08	0.02	52	0.0781
3004.7	[30, 3]	[0.3, 0.3]	0.1	1.00E-06	1.00E-07	0.03	44	0.0798
2004.7	[25, 3]	[0.3, 0.3]	0.2	1.00E-07	1.00E-07	0.03	128	0.0801
3004.7	[30, 3]	[0.3, 0.3]	0.1	1.00E-07	1.00E-08	0.04	85	0.0805
3004.7	[30, 3]	[0.3, 0.3]	0.2	1.00E-07	1.00E-07	0.04	62	0.0811
2004.7	[30, 3]	[0.3, 0.3]	0.15	1.00E-06	1.00E-07	0.03	18	0.0811
3004.7	[30, 3]	[0.3, 0.3]	0.1	1.00E-06	1.00E-08	0.02	90	0.0818
2004.7	[25, 3]	[0.3, 0.3]	0.15	1.00E-08	1.00E-08	0.03	96	0.0827
2004.7	[30, 3]	[0.3, 0.3]	0.2	1.00E-08	1.00E-07	0.03	88	0.0829
3004.7	[30, 3]	[0.3, 0.3]	0.15	1.00E-07	1.00E-06	0.03	125	0.0832
3004.7	[25, 3]	[0.3, 0.3]	0.2	1.00E-07	1.00E-07	0.03	20	0.0841
3004.7	[30, 3]	[0.3, 0.3]	0.2	1.00E-06	1.00E-07	0.03	120	0.0843
3004.7	[30, 3]	[0.3, 0.3]	0.15	1.00E-08	1.00E-08	0.04	34	0.0845
3004.7	[30, 3]	[0.3, 0.3]	0.2	1.00E-06	1.00E-08	0.04	47	0.0848
2004.7	[25, 3]	[0.3, 0.3]	0.1	1.00E-06	1.00E-07	0.03	115	0.0851
2004.7	[30, 3]	[0.3, 0.3]	0.15	1.00E-08	1.00E-08	0.03	63	0.0857
2004.7	[25, 3]	[0.3, 0.3]	0.15	1.00E-07	1.00E-06	0.04	124	0.0860
1788.5	[30, 5]	[0.4, 0.4]	0.2	1.00E-06	1.00E-07	0.04	129	0.0867

2004.7	[30, 3]	[0.3, 0.3]	0.1	1.00E-06	1.00E-08	0.02	87	0.0871
2004.7	[30, 3]	[0.3, 0.3]	0.1	1.00E-07	1.00E-08	0.02	113	0.0875
3004.7	[25, 3]	[0.3, 0.3]	0.2	1.00E-07	1.00E-08	0.03	40	0.0876
3004.7	[25, 3]	[0.3, 0.3]	0.2	1.00E-07	1.00E-06	0.02	48	0.0880
3004.7	[25, 3]	[0.4, 0.4]	0.15	1.00E-08	1.00E-07	0.03	116	0.0892
2004.7	[25, 3]	[0.3, 0.3]	0.15	1.00E-08	1.00E-06	0.03	100	0.0894
3004.7	[30, 3]	[0.3, 0.3]	0.1	1.00E-07	1.00E-07	0.02	10	0.0911
2004.7	[30, 3]	[0.4, 0.4]	0.2	1.00E-06	1.00E-07	0.04	45	0.0916
3004.7	[25, 3]	[0.4, 0.4]	0.15	1.00E-06	1.00E-06	0.02	91	0.0925
3004.7	[25, 3]	[0.4, 0.4]	0.15	1.00E-07	1.00E-08	0.04	117	0.0926
2004.7	[30, 3]	[0.4, 0.4]	0.2	1.00E-06	1.00E-07	0.03	4	0.0927
2004.7	[30, 3]	[0.3, 0.3]	0.1	1.00E-07	1.00E-07	0.03	1	0.0938
2004.7	[25, 3]	[0.4, 0.4]	0.15	1.00E-07	1.00E-08	0.03	101	0.0944
3004.7	[30, 3]	[0.4, 0.4]	0.2	1.00E-07	1.00E-06	0.02	111	0.0946
2004.7	[30, 3]	[0.3, 0.3]	0.1	1.00E-08	1.00E-06	0.04	127	0.0958
3004.7	[25, 3]	[0.3, 0.3]	0.15	1.00E-07	1.00E-06	0.03	41	0.0965
2004.7	[30, 3]	[0.4, 0.4]	0.2	1.00E-08	1.00E-08	0.03	86	0.0984
2004.7	[30, 3]	[0.4, 0.4]	0.1	1.00E-08	1.00E-06	0.02	72	0.0998
2004.7	[25, 3]	[0.3, 0.3]	0.1	1.00E-07	1.00E-06	0.02	30	0.1016
3004.7	[30, 3]	[0.4, 0.4]	0.15	1.00E-08	1.00E-08	0.02	71	0.1021
3004.7	[25, 3]	[0.4, 0.4]	0.15	1.00E-08	1.00E-06	0.04	68	0.1027
2004.7	[30, 3]	[0.4, 0.4]	0.1	1.00E-07	1.00E-06	0.03	123	0.1032
3004.7	[30, 3]	[0.4, 0.4]	0.15	1.00E-07	1.00E-06	0.02	19	0.1050
3004.7	[30, 3]	[0.4, 0.4]	0.1	1.00E-08	1.00E-06	0.04	67	0.1054
3004.7	[25, 3]	[0.4, 0.4]	0.15	1.00E-06	1.00E-08	0.02	102	0.1056
3004.7	[25, 3]	[0.4, 0.4]	0.15	1.00E-08	1.00E-07	0.02	78	0.1060
2004.7	[25, 3]	[0.4, 0.4]	0.15	1.00E-08	1.00E-07	0.02	51	0.1071
3004.7	[25, 3]	[0.4, 0.4]	0.1	1.00E-07	1.00E-08	0.03	99	0.1072
3004.7	[30, 3]	[0.4, 0.4]	0.15	1.00E-08	1.00E-07	0.04	121	0.1084
3004.7	[30, 3]	[0.3, 0.3]	0.1	1.00E-06	1.00E-07	0.02	21	0.1110
3004.7	[30, 3]	[0.4, 0.4]	0.1	1.00E-07	1.00E-07	0.04	6	0.1132
3004.7	[25, 3]	[0.4, 0.4]	0.1	1.00E-06	1.00E-08	0.02	69	0.1138
3004.7	[30, 3]	[0.4, 0.4]	0.15	1.00E-06	1.00E-06	0.04	93	0.1220
2004.7	[30, 3]	[0.4, 0.4]	0.15	1.00E-07	1.00E-06	0.02	33	0.1237
2004.7	[25, 3]	[0.4, 0.4]	0.1	1.00E-08	1.00E-08	0.02	25	0.1558
3004.7	[30, 3]	[0.4, 0.4]	0.2	1.00E-08	1.00E-07	0.04	59	0.1610
2004.7	[25, 3]	[0.4, 0.4]	0.2	1.00E-08	1.00E-08	0.03	55	0.1681
3004.7	[30, 3]	[0.4, 0.4]	0.1	1.00E-07	1.00E-08	0.03	37	0.1690
2004.7	[30, 3]	[0.4, 0.4]	0.1	1.00E-08	1.00E-07	0.03	80	0.2060

For more details and the code see: https://github.com/MaelKubli/Master_Thesis

Universität Zürich[UZH]

Philosophische Fakultät
Studiendekanat

Universität Zürich
Philosophische Fakultät
Studiendekanat
Rämistrasse 69
CH-8001 Zürich
www.phil.uzh.ch

Selbstständigkeitserklärung

Hiermit erkläre ich, dass die Masterarbeit von mir selbst ohne unerlaubte Beihilfe verfasst worden ist und ich die Grundsätze wissenschaftlicher Redlichkeit einhalte (vgl. dazu: http://www.uzh.ch/de/studies/teaching/plagiate.html).

Watt, den 31.05.2019

....

Ort und Datum | Unterschrift