Technical University of Munich

Seminar on Data Analytics for Cybercrime and Undesirable Online Behaviour (IN0014)

Professorship of Cyber Trust (I-24)

Seminar Report

Twitter's handling of right-wing users

Elisabeth Neumann, Clemens Schwarz

Examiner: Prof. Jens Grossklags, PhD.

Statement of Proper Attribution, Originality of Work

Elisabeth Neumann, Clemens Schwarz:

We hereby declare that we have prepared the present work independently without any non-trivial help of third parties and without use of other than the specified tools. The data, concepts and other contents acquired directly or indirectly from other sources are clearly indicated by the citation of the literature and appropriately highlighted in the text. The used materials from other sources do not constitute an unreasonable amount of our submitted work, i.e., well over 80% of the submitted work is written by ourselves. We are not reusing non-trivial portions of previous own work (e.g., previous seminar theses).

Abstract

In the following thesis, we are giving an overview of our analysis on the German so-called „far-right“ Twitter-sphere, i.e. the collection of politically right-wing oriented users on the Twitter social network, and their interactions and behaviour. We collected data on several accounts basing our choice on a recently compiled list of known right-leaning users and profiles, publicised by Jan Böhmermann, a German satirist and television presenter.

We analysed those tweets using a sentiment analysis script and tried to infer if the authors where behaving badly or exhibiting other unwanted behaviour.

Following our data-collection and sentiment analysis, our next goal is to build an automated Twitter bot, based on a script using Twitters API to pose as a regular user on Twitter, which is meant to bait or provoke right-wing users/trolls into interacting with it, or to provoke Twitter support into outright banning it.

1. Introduction

In recent years, the political landscape in Germany has become more polarized, following the founding of the right-wing political party Alternative für Deutschland (alternative for Germany, in the following shortened to AfD) in 2013 and its movement to the far-right populist status it has as of the time of this writing. Especially after the AfD's showing of support for the far-right extremist group PEGIDA („Patriotische Europäer Gegen Die Islamisierung Des Abendlandes", patriotic europeans against the islamization of the occident) since 2015, the party has become more and more linked to and shown signs of racist, islamophobic, anti-semitic and xenophobic tendencies, positioning it as the currently most popular right wing party.

Following the shift to the political right in Germany and other Western-European states, more and more right-wing users have begun to pop up on Twitter, where they are both reproducing right-wing and alt-right propaganda and are also harassing left-leaning users such as politicians or media outlets, while exhibiting hateful and generally unwanted behaviours on the social network. As reporting those users to Twitter for their hateful messages rarely has any consequences, users on Twitter have begun to collect users exhibiting right-wing tendencies into lists, which one can use as blocklists, collections of users which can be collectively blocked and or muted, so that one would not see their tweets or interactions with other users anymore. Many of those people publish these lists so others can also block those people.

In the following we present the work of our thesis, in which we are trying to analyse the current atmosphere in terms of sentiment of German Twitter users, regarding tweets to certain topics or in response to other users. We are mainly basing our data-pool on a fairly extensive list of over 1400 right users, which was published by Jan Böhmermann, a German TV presenter and political satirist in May of 2018, and is also known as „Böhmermanns schwarze Liste", Böhmermann's black list. He published this list as part of his anti campaign „Reconquista Internet" to the movement of „Reconquista Germanica" [1]. This far-right online movement started in the german elections 2017 and had a very popular youtube and Twitter account, which both were blocked. They now operate out of a private chatroom using the chat-client Discord, which we couldn't access, since the organisation is nowadays really cautious about journalists and other intruders, who could possibly lead to a new deletion of their community. The Süddeutsche Zeitung cites, that about 5000 user are on this Discord channel [2].

We will try to look into finding those „Reconquista Germanica“ users on Twitter, analyse if they still exist and if they are on the list from Jan Böhmermann.

After describing how we collected our data, we will explain how we analysed the tweets we found and how our sentiment analysis is conducted. After that we’ll try to make our own bots.

For our data collection we used the official Twitter API for Python instead of crawling tools. We chose this because we don’t have an enormously big list of users that we want to analyse and because the Twitter API showed to be a very reliable tool. We are furthermore also relying on official Python extensions or other web tools, that are free and save to access.

2. The Data Collection Phase

While researching our topic we thought about different ways to find right-leaning users on Twitter. At first we did some hashtag analysis of recent important topics, such as „Flüchtlinge“, „Migrationspakt“ and „Gelbe Westen“. We also looked in the „Reconquista Germanica“ hashtag, which unfortunately isn’t actively used anymore by right users. The only tweets with this content were mostly by people who are wondering where the „Reconquista Germanica“ people went to.

As we have seen in our first analysis it is very difficult to get the context in which tweets have been written and also the sentiment of the tweet or its author from just the text of the tweet without knowing the authors stand on the topic in question. Therefore we have narrowed down the user base from which we are pulling tweets to include in our data-base from anyone tweeting about a certain topic, to mainly analysing users who have been included on Böhmermann’s black list and who have such been deemed „right trolls“, or who are outright right-wing politicians. The original authors of the black list stay anonymous, but sources around Böhmermann himself and his production company say, that a data analyst compiled the list from preexisting blocklists and wrote an additional algorithm, which added users who follow more than ten users from the right spectrum and who are also followed back by them [3].

We searched for Böhmermann’s list and got a version of it through archive.is since he deleted his Twitter post. This post contained just the Twitter User-IDs. [4] Our first step was to get a list of the users by their names. Therefore we created a simple script which got us all the usernames through the Ids. We wrote this script in Python using the tweepy library.

After that we already tried to get the last tweets from all those users but our code wouldn’t run through them all and constantly crashed because several users weren’t active anymore or had a private profile. Since it would be really time-consuming to check those people all individually we wrote another separate script using the Twitter-API again in order to weed out inactive or deleted accounts.

The Twitter API limited how fast we could pull users and confirm their activity status to about 65 users per hour, so in the end it took almost 2 days of the script running. Maybe the Twitter API does this intentionally to prevent crawling of a lot of users and this way to get insights to a lot of users in a small timeframe. We suspect it's deliberately slowed.

We removed about 300 users, which left us with about 1100 users to analyse.

Also the Twitter API limits the count of last tweets that you could get to a number of 200. Since we also just wanted to look at the more recent tweets we didn't try to bypass this and just went with these official rules. The last 200 tweets still leaves us with more than 200.000 tweets, since not all users even tweeted 200 tweets.

After sorting out the users we could run our script to get all the relevant data like the last tweets, the data of the creation of the tweet, the source where the tweet was created and furthermore the sentiment, which we'll discuss later. We put our data in a local database to search for specific terms.

3. The correlation with „Reconquista Germanica“

We first wanted to look in detail into Jan Böhmermann's list since we got the usernames from the Ids. We found a graph of users [5], that used the „Reconquista Germanica“ hashtag on twitter in early January 2018, where we could see the usernames. Since there are just 114 accounts, we copied them from the graph and compared them to our list. The interesting thing is, that just 31 accounts are also on our list from Jan Böhmermann. This is quite strange since it is said, that the list from the „Reconquista Internet“ movement contains the users, that were actively involved in the far-right movement.

After checking who of them still has a valid account, we found that 34 accounts were deleted or set to private which is almost a third (about 30%) of the original accounts. In the Böhmermann-list were originally 1454 users and after we used our script, we were left with 1155, which means that just about 20% are deleted or private.

While searching for usernames like „Deutscher“ or „AfD“ we also saw that a lot of users which had been deleted from our Reconquista-list reappear with a very similar screen name in our other list which were seemingly still active. So names like „AfD_Support“ and „AfD_supp“ are hard to distinguish but it's also not possible to say if these are the same people who created a new account.

4. The Sentiment Analysis

Jan Böhmermann's list

After building our data base, we started to analyse the tweets on their sentiment, which means that we want to use an algorithm which categorises tweets as inherently positive, negative or neutral. Using the German language extension of the TextBlob library [6] in Python, we first looked at each tweet, and let the text blob tool automatically analyse the sentiment. The algorithm assigned each word a numerical value corresponding to if it is positively (+1), negatively (-1) or neutrally (0) connotated. summing up the connotation-values of all words in a tweet, we can extrapolate the whole tweets sentiment, i.e. conclusively say if a tweet is rather positively or negatively connotated.

Given that we are pulling those tweets from a list of users, we now calculated for each individual user which percentage have which sentiment.

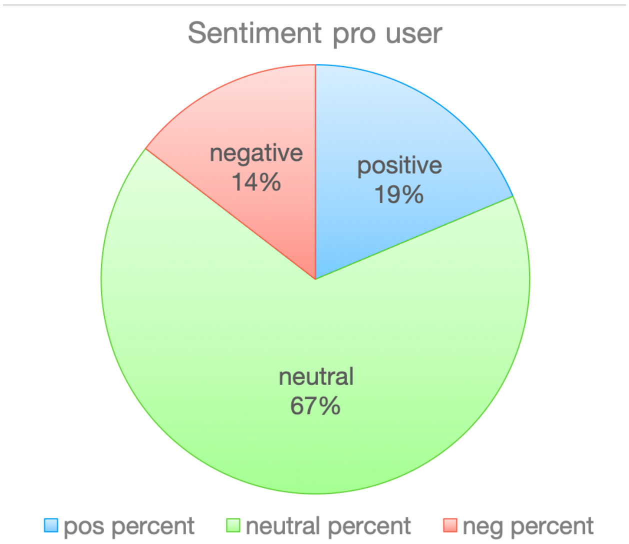

fig. 1, sentiment analysis

The pie-chart in figure 1 shows the overall median percentage of tweets and their connotations; the results are very surprising; about 14% of all collected tweets show negative connotation (red), 19% show positive connotation and the remaining 67% show neutral connotation. In our opinion this is a surprising outcome since everyone expects worse numbers.

The following line-chart shows the percentage of negatively connotated tweets in regards to all observed users, showing that while most users achieve only the average of 14,5% negatively connotated tweets, there are some significant exceptions both in the negative and positive direction, which we will look at specifically in detail.

The „Reconquista Germanica“ list

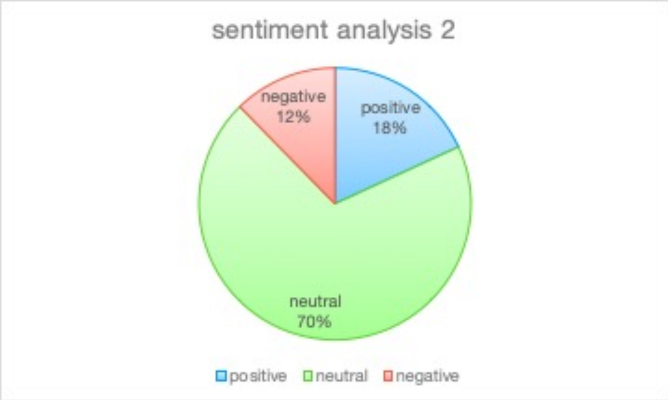

fig. 2, sentiment analysis

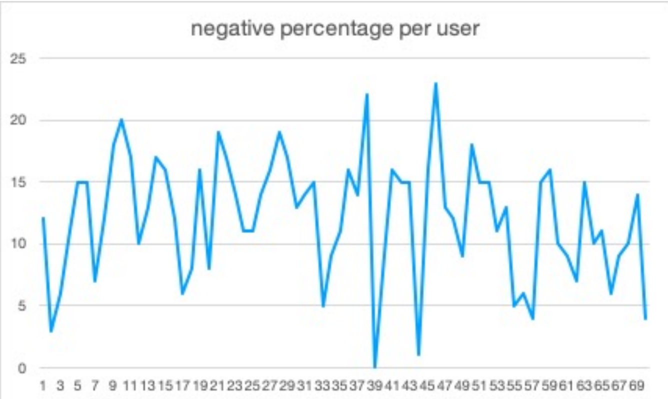

fig. 3, negativity

The sentiment analysis of those 70 remaining users was a real surprise. We expected them to at least differ a bit from our first analysis but as is shown in the graph (figure 2) the numbers are even better. The negative connotation is at just 12% (the Böhmermann-list has 14%).

The distribution of negative percentage per user as seen in figure 3 is also quite interesting. While the analysis of the first list showed, that there are a lot of outliers that even have higher that 40% negative percentage of tweets we can now see that the maximal negative percentage here is at just 23%. We'll look into possible reasons for this interesting behaviour later.

5. Processing Data into WordClouds

Following the Sentiment Analysis, we sorted the data list to only include users which show an average of more than 20% negative or positive tweets for our next step in processing the data. We chose both very positive and very negative users in order to have two sets of groups to compare since we didn't understand how the sentiment analysis could reveal such supposedly positive results.

We also wanted to find a way to see the most used words to generate bots that react to those words.

As a third control WordCloud we generated a WordCloud for all users as showed in figure 6.

fig. 4, all nouns

fig. 5, negative nouns

fig. 6, all nouns, Reconquista Germanica list

Using the already reduced set of tweets we cut off the endings of words so that only their roots remained for the sentiment analysis. After that we cleaned the tweets from the retweets.

We proceeded to visualise this data into WordClouds. These diagrams display the most used words graphically using different sizes and colours to indicate higher usage of a word in more tweets, and also how their connotation is valued by our sentiment analysis algorithm.

For this we used the German stop set from the natural language toolkit from Python [7], which also includes a German library. Since we also wanted to ignore sentences like „The Tweet is unavailable", which we'll get for deleted tweets, we also included them in our stop set.

It was quite difficult to find a good German grammatical tagging tool, that correctly got us the nouns since we just wanted to look at them. We continued to use TextBlob with it's build in tagging tool. While the provided stop words and the noun tagging worked relatively good, Twitter users unfortunately don't always use the right spelling or use abbreviations that the toolkits doesn't know.

The above images show the final results of our algorithm for the Böhmermann list. figure 4 shows the WordCloud for all collected nouns, while figure 5 only displays all nouns collected from negative marked tweets into a WordCloud.

For comparison you can see the WordCloud of the „Reconquista Germanica“ list in figure 6.

It's interesting that the most used words in both lists and even in the „all nouns“ and in the „negative nouns“ cloud looked quite similar. Looking into the most used words we saw „Deutschland“, „AfD“ and „Merkel“. Most of the words weren't surprising and in most cases even expected. Just by looking at random tweets from the users of all lists we got a feeling for what they were writing about. The only maybe remarkable thing is that the word „Nazi“ seems to be used more often in the WordCloud with all nouns. We searched in our database for tweets, which contained „Nazi“ to understand how those users were using the word in context because it seems quite ironic for right-leaning users to write something about nazis. Here are examples to get a better understanding:

„[…] Jeder, der #Blutwurst mag, ist ein Nazi und muss auch sofort blockiert werden.“

„Immer diese Nazis von der AfD. Moment mal... gibst die überhaupt? […]“

„Nazis haben auch Wasser getrunken, darf ich jetzt kein Wasser mehr trinken? […]“

There are a lot more tweets that also point in the direction that the term nazi is mostly used in a sarcastic or self-ironic way.

Remarkably is also that the authors of those three tweets shown above were connotated to mostly positive tweets (each of these three has less than 15% negativity in the sum of all their tweets) by our sentiment analysis. This is again a good example for why it is so hard to get an algorithmically analysis of the sentiment because things like sarcasm or mockery always have to be analysed within a certain context.

6. Sources and detecting bots

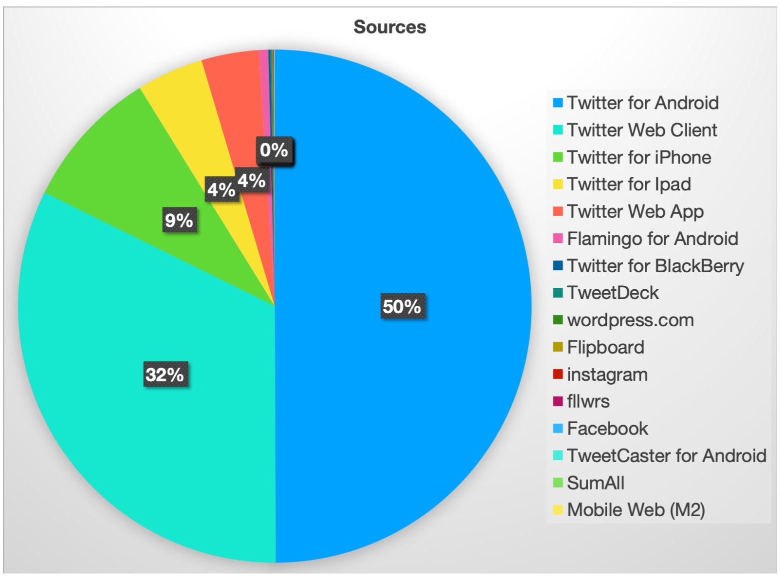

fig. 7, sources, reconquista

Following that, we wanted to check if there are even any bots in our list before we make our own. The first aspect we wanted to look at were the sources from where the users were tweeting. In following figure 8 you can see the different sources of the users from the „Reconquista Germanica“ list, sorted by their quantity. It's quite interesting that Twitter makes this data public. While it is no surprise to see that the most-often used sources are the official Twitter for Android or the Twitter Web Client in the list, we were surprised about sources like „fllwrs“ or „SumAll“ because we did not know them as Twitter-clients. After researching them ([8], [9]) we found that those both are Web-applications used to keep track of how many followers you have or notify when someone unfollows. Other sources like Flipboard and TweetDeck are third-party Twitter-clients used as options other than the official website or app.

We expected to maybe find some traces of users using bots with certain extensions, that make it easier to use bots without any programming skills, but just looking at the sources didn't help us.

After that we looked at tools to detect even just one bot in our lists. The main problem is that most of the tools work just for checking a specific user in an online web tool and then getting a probability for this user being a bot [10].

While they're great for a small number of users and seem to have relatively good results it's difficult for our big number of users. We didn't want to implement a machine learning code to check all our users ourselves so we have to pre sort to just check at the most suspicious or interesting ones.

fig. 8, bot analysis

We already collected all the data so we took our preprocessed data from the earlier analysis. At first we took the ten users with the highest negative sentiment and then the ten with the most positive. The most interesting thing is that we found two accounts which show a high probability to be bots in the ten most positive users. We were able to export the results in a JSON-file but we took some screenshots from the website. In figure 9 you can see the three most suspicious accounts being „leaks_nsu“, „SchreibedenText“ and „Claudia_Hin24“. In addition to „Botometer“ we also used a browser-extension named „botcheck“ [11] to validate the results.

The interesting thing is that both tools had different results. „Botcheck“ wasn’t sure about „Claudia_Hin24“, said that „leaks_nsu“ isn’t a bot, but was sure that „Schreibeden-Text“ is most definitely a bot. And indeed, when we looked at the user profile we found that it had no tweets but 233 answers. It also follows nearly a thousand users. The machine learning algorithm that both tools use also look at those features and also the frequency of posting, the specific time it was posted and all the indicators for algorithmic specifications. The differences between those two tools could be from the different evaluations of language indicators. The tools are in English so they maybe have their problems with the German language and yield therefore different results.

We used these results to make our own bot as described in the following section.

7. Twitter-Bots

By condensing all used nouns into these WordClouds, we can now begin to build our bots as part of the practical evaluation of the collected data. The general idea of Twitter-bots is to built an app for Twitter which uses read- and write-privileges to take control over a Twitter-account. Usually such apps are used by companies or groups for automating an official/representative account’s tweets, i.e. retweeting another users tweets, sharing links from a different website (e.g. when a new blog-post is published) or as third-party tools, such as the aforementioned tools to keep track of tweet-performance or followers. In the case of bots, especially the write-permissions of the application are being used to make an account tweet by itself. Using scripts or already existing Twitter-applications, it is fairly easy to build a convincing bot, which will tweet natural-seeming sentences periodically.

There already are multiple web-applications, such as „Cheap Bots Done Quick“ which help users to easily build grammars from which a script pulls parts to build natural sentences [12]. We chose to build our bot ourselves by writing its behaviour with Python’s Tweepy-library [13], which makes use of the official Twitter-APIs to interact with Twitters services.

8. Building our Bot

After our first steps in the data-collection part we already had an extensive list of users which we wanted to observe. To do this we firstly had to set up a convincing Twitter-account which did not outright look like a bot-account. One of the more indicative signs of a bot on Twitter are the name und the identifying screen-name. Carelessly set-up bots use the default screen names, which almost always include the users first name and a string of arbitrary numbers. Also most bots (or most throw-away or troll accounts for that matter) use the default-profile picture. Using an account with all those defaults is almost always a sign of an account which is meant to not last long, as it will be reported for misbehaviour and consequently banned from the site.

fig. 9: our bot

Thus, we employed several strategies to make our bot's account as convincing as possible: as the username, we chose a combination of 2018's most popular baby names for boys, and a loose interpretation of the German placeholder name, „Max Mustermann". Seeing as how we wanted to blend in with the users from our list, we chose to use Paul_Jonas_de as the screen name, as most users on the list use some sort of indicator of their patriotism in their screen name.

Not wanting to use a real persons face, we found a website by the name of ThisPersonDoesNotExist.com, which uses machine learning and a neural network to generate convincing facial images of non-existing people [14].

Following the set-up of the account as showed in figure 9 and linking it to an authorised Twitter-Developer account, which is how we were going to be able to let a script run the bot's behaviour, we wrote a short script to follow all people from the list. As to not be limited by the Twitter-APIs permissions on how often to perform certain actions, we only followed the about 65 users from the Reconquista-Germanica list, as those seemed to be rather active on Twitter anyway and a large enough group to run our experiment.

The next step was to build a script which makes the bot actually interact with the people it follows. We wrote a script which periodically looks at the newest 20 tweets and listens for certain keywords. These keywords were sourced from the before-generated WordClouds and were picked to be most-likely used by our targets. Once the script recognises one of those words in a tweet, it will respond to the tweet with one of 42 prewritten phrases, which are written as sceptical questions to the original tweet; e.g. asking for reliable sources on the mentioned topic, asking if the original tweet was

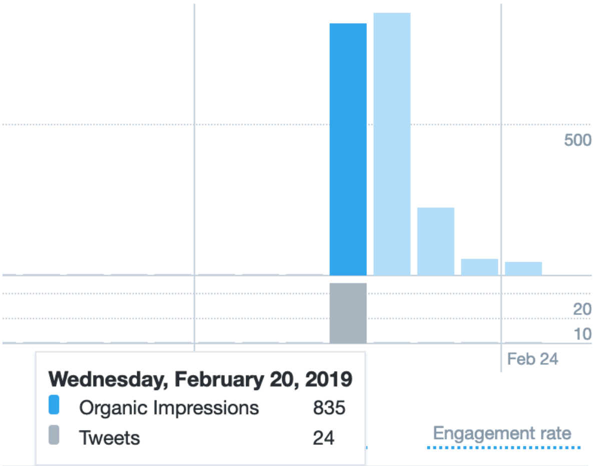

fig. 10: Tweet Activity analysis

meant seriously, or generally expressing disbelief in what the original tweet was saying is true.

During testing this script we quickly found that most people on the list were actually not tweeting themselves, but rather retweeting, i.e. sharing other user's tweets or links. Those tweets or links were in almost all cases themselves either polarising opinions on the political topic-du-jour or even straight-up hate speech. While this first posed a problem as we had to decide if we wanted to include the retweets as well, or wanted to focus on exclusively original tweets, we came to the conclusion that it would not only seem more natural for a sceptical user to not only answer to original tweets, but also to retweets, but that it would also increase the scope of users which our bot could interact with.

We set up the script to collect all tweets which it replied to in a list, as to not reply more than once to a tweet with more than one keyword. We also included a time-delay in the script so that it would only pull the newest 20 tweets every 20-40 minutes and take one to three minutes to reply to tweets with keywords in them, as to not be limited by the API's maximum allowed tweets, which would also very likely trigger systems meant to detect automated bots.

As the script which governs the bot's behaviour has to run constantly to interact with the other users in real time, we used a cloud-service called PythonAnywhere [15] where we could upload our code and execute it without it having impact on our own computers.

9. The Bot's Performance

After starting to run the script in the cloud, our bot began to reply to tweets and retweets in its timeline which included one or more of our keywords. Sadly, our bot only managed to tweet 24 replies to tweets before it was either reported for spamming or our API permissions were revoked by Twitter itself. As of the writing of this, we have yet to receive any word from Twitter as to why our permissions were revoked, but we suspect that our bot's behaviour infringed on Twitter's rules for „good bot behaviour", which includes that a bot should not tweet directly at users without being mentioned in a tweet first. Strictly speaking, while our bot does not adhere to what could be considered „basic Twitter bot etiquette“ [16] our bot does not contravene this rule, as it only replies to certain tweets, but this actually is a good sign, as it shows that Twitter has implemented systems in order to fight against automated bots or trolls which harass people on the platform.

Apart from that, our bot achieved higher rates of interactions than we expected. Twitter’s integrated Tweet Activity analysis (pictured in fig. 11) shows, that our mere 24 Tweets got over 2000 interactions over a span of just three days, meaning that users have seen or clicked on our tweets and visited the bot’s profile over 2000 times. We even got six people to reply to our tweets, which means that they at least were convinced enough by our questions and our profile to engage with it.

10. Problems

While our initial goal was to build multiple bots based on our findings in the data analysis part, we quickly found that Twitter has actually not only made it more difficult to create multiple profiles, but that it was also too difficult for our goals to build multiple bots with only a single Twitter Developer account. Twitter requires a phone number to be associated with an account, greatly limiting the number of accounts we could even possibly create.

After we found a rather time-consuming workaround for that, we noticed that we could only control a single account’s behaviour, which has to be directly linked to the developer account. To build an actual application that multiple users can sign into and enable it to automatically tweet or interact with other users on the platform, requires not only the script, which governs the behaviour, but also a website or other means of enabling a users to sign into the app, which was out of the scope for what we wanted to accomplish in this thesis.

Also, our initial goal of building a so-called „honeybot“ [17] fell short as we quickly saw, that German Twitter users behaved differently than US-based Twitter users: while German users replicate hate-speech and share right-wing publications’ articles, US-based right-wing users are actively searching on Twitter for other users who tweet about certain topics, as to reply to their tweets and harass them. This behaviour is similar to how our bot was meant to operate, only that we were restricting our bot to its own timeline and the tweets of people it follows, as it would be impossible to write a simple script to filter out in which context random users are using certain keywords.

Apart from our bot, it was our surprise that during the sentiment-analysis, most tweets were flagged neutral, which shows that either the algorithm is not harsh enough as to how the over-all sentiment of a tweet is calculated, or that the strategy of calculating a tweets connotational sentiment is flawed from the beginning, as a lot of context and subtext is missing when only looking at a single tweet.

Context also plays a big role in the over-all sentiment analysis, as we could easily observe, that while a lot of hateful content in tweets is being generated, many responses to negatively connotated tweets may be flagged as positively connotated, even though from context, the response is in agreement to the negative tweet, thus counting as a positive tweet.

Furthermore, the algorithm can not recognise rhetoric tools such as satire or cynicism, which makes it impossible to see, if a tweet is written as for example a genuine criticism of current political happenings, or rather just comedic comments or musings.

11. Results

As seen in our data analysis there are a lot of users that spread far-right thoughts and behave in a way that one could say are quite undemocratic and even alarming in terms of German history. While it is easy to come to this conclusions when looking at specific users that say things like:

„*Nazis: Seit 1945 eine minimale Randerscheinung; Kommunisten: in der Regierung und drauf und dran Deutschland zu zerstören*“,

„*Männer bekamen das Wahlrecht, weil sie ihr Leben im Militär aufs Spiel gesetzt haben. Frauen bekamen das Wahlrecht, weil sie genervt haben.*“, or

„*Frauenwahlrecht abschaffen! #saynotofeminism #frauenwahlrechtabschaffen*“, (just to name some best-of's) it is quite hard to find an algorithmic way to track those users and delete them from Twitter. The only way is to report them manually, but as many of those users seem to be in a so-called peer bubble of right users, that mostly interact between each other, they somehow stay at the platform. Also Twitters policy seems to be „as far as no one is concerned everything is fine“ meaning that you really actively have to report them when they are real users. Most bots can be identified as such by analysing their behaviour algorithmically in order to delete them if they're suspicious, but as shown in our analysis it is not too easy to detect social bots from the outside because well set-up bots are meant to blend in with other regular (non-bot) users, and to behave like a normal person.

On the other side it became quite clear, as we ourselves noticed, that Twitter really complicated the process of creating of bots by requiring binding to a telephone number. This really limits the number of bots that one can write so bots may nowadays have a lot less presence in Twitter.

References

[1] Spiegel Online: Wie die Rechtsextremen mit Hetze mobil machen (September 2017): http://www.spiegel.de/spiegel/wahlkampf-mit-trollen-wie-die-rechten-im-netz-mobil-machen-a-1167063.html

[2] Süddeutsche: Wie rechte Internet-Trolle versuchten, die Bundestagswahl zu beeinflussen (February 2018): https://www.sueddeutsche.de/politik/manipulation-im-netz-wie-rechte-internet-trolle-versuchten-die-bundestagswahl-zu-beeinflussen-1.3875073

[3] Kritik am Internet-Pranger für Rechte: “Jan Böhmermann zeigt, wie totalitäres Denken wieder cool werden kann” (May 2018): https://meedia.de/2018/05/11/kritik-am-internet-pranger-fuer-rechte-jan-boehmermann-zeigt-wie-totalitaeres-denken-wieder-cool-werden-kann/

[4] Archive of Jan Böhmermanns Twitter account (originally from May 2018): http://archive.is/TBeAI

[5] Getarnt als Gamer: Einblicke in eine rechtsradikale Troll-Armee https://netzpolitik.org/2018/getarnt-als-gamer-einblicke-in-eine-rechtsradikale-troll-armee/

[6] TextBlob: Simplified Text Processing (2018): https://textblob.readthedocs.io/en/dev/

[7] Natural Language Toolkit (November 2018): https://www.nltk.org

[8] Welcome to *fllwrs*, the Twitter follower tracker: http://fllwrs.com

[9] Share your success automatically: https://sumall.com/performancetweet

[10] Botometer: https://botometer.iuni.iu.edu

[11] Detect & Track Twitter Bots: https://botcheck.me

[12] Cheap Bots, Done Quick!: https://cheapbotsdonequick.com

[13] Tweepy library: http://www.tweepy.org

[14] https://www.thispersondoesnotexist.com;
Explanation in the The Verge article „ThisPersonDoesNotExist.com uses AI to generate endless fake faces“(February 2019): https://www.theverge.com/tldr/2019/2/15/18226005/ai-generated-fake-people-portraits-thispersondoesnotexist-stylegan

[15] PythonAnywhere https://www.pythonanywhere.com/

[16] basic Twitter Bot Etiquette (March 2013): http://tinysubversions.com/2013/03/basic-twitter-bot-etiquette/

[17] Nora Reed, The Saga of @christianmom18 (October 2016): https://medium.com/@NoraReed/the-saga-of-christianmom18-8ac09ceab9c6