Web of Lies: The Spread of Political Misinformation and Twitter trollbots in the 2018 Senate elections

By Ajay Jain and Justin Tu

During the 2016 U.S. election cycle, a deluge of political misinformation swept across the Internet. Studies have concluded that such misinformation, spread by bot and troll accounts, might have been enough to sway voter opinion and potentially the outcome of the election. Much like tabloid magazines, tweets, retweets, articles containing fake news are designed to make readers believe in controversial and incorrect statements of a political party or candidate, theoretically causing them to hold strong negative feelings, vote for someone else, or even not voting at all. The resultant spread of political misinformation in the election cycles potentially misleads voters, sows distrust, and undermines democracy. With massive importance surrounding the 2018 midterm elections and the continued growth of social media websites including Twitter, our study will determine if Twitter bot and troll accounts are continuing to tweet and retweet fake news and other misinformation pertaining to Senate candidates and races. In addition, we will analyze which races and candidates are being targeted by bot and troll accounts.

Although both Twitter bots and Twitter troll accounts fulfill the same purpose, the two are designed differently. A Twitter bot account is defined as "a computer algorithm that automatically produces content and interacts with humans on social media, trying to emulate and possibly alter their behavior" (Ferrara et al. 1). They are designed to generate content on their own. In addition, bots also retweet content tweeted by other bots as well as articles from fake news websites in order to spread misinformation quickly and efficiently. Simple bots can be written in Python to search for tweets mentioning a keyword, e.g., "Donald Trump," and retweet each tweet with the keyword. They can also be written to retweet every time a new article is published on a fake news website (as well as headlines on real news websites that are politically skewed). Furthermore, Twitter bots can be programmed to tweet the same message across multiple accounts within seconds. On the other hand, troll accounts are managed mainly by humans, yet fulfill the same purpose: spreading misinformation and fake news in order to deceive voters by means of convincing them to vote for a certain candidate or suppressing their motivation to vote.

Some bot and troll accounts fool Twitter users into thinking that they are legitimate accounts by posing as real human beings. In fact, a multitude of Twitter bot and troll accounts were formerly legitimate accounts. However, once these accounts become inactive or hacked, other websites sell these accounts for just a few dollars (Bhat and Phadte 2017). Upon being sold, the accounts are set up to spread misinformation either autonomously, manually, or via a hybrid of the two.

While Facebook, Twitter, and other social media websites have vowed to crack down bot and troll accounts, they have been somewhat unsuccessful. Political bots and trolls have continued their activity on social media since the 2016 presidential election. For example, during the 2018 Illinois gubernatorial primaries, bot and troll accounts were found to tweet and retweet statements supporting Jeanne Ives, a far-right opponent to the incumbent Republican governor Bruce Rauner (Smith 2018). In September of the same year, a tweet depicting multiple Twitter bot accounts tweeting the exact same tweet rallying for Ted Cruz at the same time went viral (White 2018). Rather than staying dormant after the 2016 presidential election, these accounts have continued their activity well into President Trump's administration. Not only are many accounts from 2016 still generating content, but new accounts created within the past year are also spreading misinformation during the 2018 election cycle.

Our research study seek to identify four questions regarding Twitter bot and troll accounts spreading political misinformation in Senate elections in the 2018 midterms. We first wish to tabulate and classify political orientation of Twitter bot and propaganda accounts, alongside the contents they produce. The second goal is to determine whether those accounts generated more content on battleground states versus non-battleground ones. Examples of battleground Senate races include North Dakota, where incumbent Democratic Senator Heidi Heitkamp is in a tight race with Republican Congressman Kevin Cramer, and Pennsylvania, where incumbent Democratic Senator Bob Casey is widely expected to beat his opponent, Republican Congressman Lou Barletta, despite Pennsylvania delegating its electoral votes to President Trump in 2016. Our study's third aim is analyzing the content format these inauthentic Twitter accounts generate, to see the proportion of text, image and articles posts. Do these accounts generate text content or image content?

Are they mainly focused on spreading just text or a combination of information types? Finally, the fourth goal is to measure trollbot engagement. Are bots and trolls mainly interacting with each other, or are they acting in a solidary manner? Interaction between trollbot accounts could mean that the network between these accounts is more complex in nature. However, if these accounts are not interacting, then troll and bot accounts would be more independent from each other and not controlled by similar groups.

In order to answer these questions, our research team started by writing computer programs in Python and R that save tweets mentioning Senate candidates and analyze if the accounts mentioning them are Twitter bots or trolls. The accounts returned by our programs are then passed into two other machine learning algorithms (Botcheck.me and BotSentinel) that detect if the account is a political bot or troll. For each account that is a bot or troll, we determine its political affiliation manually. For each tweet by political trollbots, we analyze the type of content being generated, if other accounts have tweeted the exact same tweet (either via a retweet or using the exact same message in a tweet), and which Senate races are being targeted.

Our program collected data from Twitter between October 29th and the midterm election day, November 6th, 2018 since this date duration was the last week of the 2018 election cycle. From the past case of bot and troll accounts heavily increasing activities in the final two weeks of the 2016 election cycle, we hypothesized that bot and troll accounts would also ramp up their activity closer to the midterms. Therefore, we believed that most of the tweets would be created a day or two before the election.

The research team wrote computer programs in Python that collects live Twitter data through Twitter's application program interface (API) and multiple API keys. Tweets mentioning any Democrat or Republican Senate candidates' names or Twitter handles are streamed to our software in real time. For example, if we want to find tweets about Senator Ted Cruz of Texas, our Python program looks for tweets mentioning either "Ted

Cruz", "@tedcruz", "@SenTedCruz", or "@TeamTedCruz".

	A	B	C	D	E
1	id	user	tweet	created_at	user_created_at
2	1.05706E+18	lezleedee2	Debbie Stabenow has been a fierce protector of the Great L	10/30/18 0:19	12/29/16 2:48
3	1.05706E+18	FrahmNathan	@tedcruz Looking good - I see winners!	10/30/18 0:19	10/4/18 9:11
4	1.05706E+18	JPlauto	Bueno Lobito a mi un whiski si queres cambialo..jaj saludos	10/30/18 0:19	2/27/15 16:46
5	1.05706E+18	prcsjoyyyyy	faithful*	10/30/18 0:19	5/25/18 15:12
6	1.05706E+18	Karmaisablink	Looking for the #Cardi and #Nicki drama? Here is the tea,	10/30/18 0:19	10/25/10 2:43
7	1.05706E+18	Griv0is	@dccc I see you dishonestly omit that @HillaryClinton LOST	10/30/18 0:19	3/17/10 14:09
8	1.05706E+18	BarbaraDarlin	.@realDonaldTrump .@FoxNews .@foxandfriends	10/30/18 0:19	1/26/11 19:40
9	1.05706E+18	JermePhilip	@seungminkim @clairecmc WOW	10/30/18 0:19	3/9/09 7:46
10	1.05706E+18	amabilecorall	pessoa chega em mim e fala	10/30/18 0:19	7/30/18 0:29
11	1.05706E+18	WellingMichael	#29Oct"Brazil woke up because of what happens in Venezue	10/30/18 0:19	8/19/11 17:45
12	1.05706E+18	24Costa_	" morro solteiro mas n boto meu wpp na bio do tt kkkkkkkkl	10/30/18 0:20	11/25/16 22:52
13	1.05706E+18	pippo_lt	@mike_pence @JohnJamesMI @SchuetteOnDuty @GOP V(	10/30/18 0:20	7/23/15 8:00
14	1.05706E+18	lrcpt	Senator Debbie @Stabenow is a fierce advocate of diversity	10/30/18 0:20	4/22/12 16:20
15	1.05706E+18	edallas53	Michigan women earn 78 cents for every $1 earned by men	10/30/18 0:20	12/25/17 1:03
16	1.05706E+18	CynthiaTheBaker	Michigan women earn 78 cents for every $1 earned by men	10/30/18 0:20	3/22/15 13:28
17	1.05706E+18	DairaSnchez1	@diegoarroyogil @elmundoes El prisionero del PREMIO NO	10/30/18 0:20	10/26/16 18:38
18	1.05706E+18	l0ftydrEameR	@maydaymindy9 @BetoORourke @tedcruz lâ€™ll post a ph	10/30/18 0:20	4/15/09 15:39
19	1.05706E+18	bel_azar	" eu nÃ£o sou hÃ©tero, eu sou bonita" Azar, OlÃ-via 2018	10/30/18 0:20	8/4/18 23:19
20	1.05706E+18	SuellOliveira13	mas sempre q chega no domingo Ã¡ noite, mesmo morto eu	10/30/18 0:20	7/26/11 19:26
21	1.05706E+18	kathyb5783	@darrell_partin @ruthMP20 @KyleK1x @VoteMarsha @rea	10/30/18 0:20	8/28/09 14:18
22	1.05706E+18	iPolinho	O botÃ£o de "Fav" vai sair, mas o "editar tweet " nadinha nÃ	10/30/18 0:20	6/6/18 18:25
23	1.05706E+18	PVistalli	@realDonaldTrump @VP @senatemajldr @SpeakerRyan @	10/30/18 0:20	11/7/14 5:45
24	1.05706E+18	squishymom1	@KimsMom3 @BetoORourke This is how WE roll! Could	10/30/18 0:20	5/2/13 17:06
25	1.05706E+18	hjs52	"@JornalOGlobo "	10/30/18 0:20	9/3/09 12:32
26	1.05706E+18	6Gwen	Les mÃ©dias en mode " j'suis pas venue ici pour souffrir OK	10/30/18 0:20	3/3/12 21:54
27	1.05706E+18	Clara_Resists	Debbie Stabenow believes we should export our products &	10/30/18 0:20	4/4/11 23:34
28	1.05706E+18	amaralan3	@jarrodagen @mike_pence @JohnJamesMI YAY - AWESOM	10/30/18 0:20	3/28/11 0:04
29	1.05706E+18	GoclanB	@LaurenAlisabeth @BetoORourke This is what democracy l	10/30/18 0:20	5/12/17 3:06
30	1.05706E+18	Saleh3siri	@Hash_law " Ù¤Ù…Ù† Ø³Ù¤Ø§ Ù¯Ø£Ø¡Ù„Ø- Ù¤Ø£Ø¬Ø±Ù‡	10/30/18 0:20	9/4/12 16:17
31	1.05706E+18	lemisanthrope78	@UncleVeracity @fe5082 @HawleyMO @FoxNews @cvpay	10/30/18 0:20	12/2/14 2:22
32	1.05706E+18	DUANELEEPROCTOR	I WILL WORK FOR ANYONE WHO OPPOSES THIS POS IN MY	10/30/18 0:20	1/5/09 14:12
33	1.05706E+18	ameashmun	" bagus untuk pelancongan " kata orang	10/30/18 0:20	5/22/11 21:50
34	1.05706E+18	the_squarrell	I will never be convinced that s2 of Felicity isn't a masterpie	10/30/18 0:20	1/23/09 23:05
35	1.05706E+18	Yara73814957	" Ø§Ù…Ø±Ø£Ù‹ Ù¤Ø¬Ø·Ø¬Ù…Ù‹Ù„ " ðŸŒ¸	10/30/18 0:20	2/14/17 22:45
36	1.05706E+18	kendra48640	@nature_art_ed @JohnJamesMI @kayleighmcenany @stab	10/30/18 0:20	5/10/09 14:43
37	1.05706E+18	JusMuzic2	Now Playing " Farmer Nappy - Chippin " On https://t.co/rqV	10/30/18 0:20	9/17/13 0:52

Every ten thousand tweets, the Python program exports the tweets into a comma-separated values (CSV) file. Each CSV file holds the ID of the tweet, the user who tweeted it, the tweet mentioning the Senate candidate, the tweet time, and the time when the user was created. All of this information is vital in detecting Twitter bots and troll accounts and examining their activity during the midterm election cycle. For example, using the created_at variable, our program can determine how quickly trollbot accounts generated content.

Once a CSV file is created, it is imported into a Python program on an Amazon Web Services server. The Python program goes through each of the ten thousand individual tweets in a CSV and checks each user to

determine if it is a normal Twitter user or a potential bot or troll account.

botdetection4.py

```
  }
bom = botometer.Botometer(wait_on_ratelimit=True,
                          mashape_key=mashape_key,
                          **twitter_app_auth)
trollbots = set()
notTrollbots = set()
with open('13.csv') as csvfile:
    reader = csv.DictReader(csvfile)
    for row in reader:
        print index
        index += 1
        if row['user'] in trollbots:
            with open('13bots.csv', 'a') as botfile:
                out = [row['user'], index]
                writer = csv.writer(botfile)
                writer.writerow(out)
            botfile.close()
        elif row['user'] in notTrollbots:
            continue
        else:
            try:
                result = bom.check_account(row['user'])['scores']['english']
            except:
                continue
            if result >= 0.7:
                with open('13bots.csv', 'a') as botfile:
                    out = [row['user'], index]
                    writer = csv.writer(botfile)
                    writer.writerow(out)
                botfile.close()
```

If the account exhibits bot or troll like properties, the program logs the username and when it appears in the tweets dataset into another CSV file. Bot checking in Python is done through the botometer[1] Python package, which uses Indiana University's Botometer[2] program to analyze each user. While Botometer did an excellent job detecting Twitter bots and trolls, it did not distinguish well between political and non-political

[1] https://pypi.org/project/botometer/
[2] https://botometer.iuni.iu.edu/

bots. As a result, most accounts that Botometer suggested as political bots and propaganda accounts are not relevant, generates many false positives.

In order to verify which Botometer-suggested accounts are actually political bots and trolls, we fed each suggestion into two web applications, Botcheck.me[3] and Botsentinel[4]. Botcheck.me was designed by Ash Bhat and Rohan Phadte at the University of California, Berkeley; Botsentinel was created by Christopher Bouzy, a computer scientist whose work with fake news accounts, bots, and trolls has been featured in the Washington Post[5]. Botsentinel and Botcheck.me use two different machine learning algorithms to predict the likelihood of an account being a bot or troll. Because Botsentinel and Botcheck.me do not provide APIs, an additional Python program is used to run a virtual web browser that automatically takes each potential bot from the CSV files and gives it to Botsentinel and Botcheck.me to analyze, thus saving several hours of manual tasks.

[3] https://botcheck.me/
[4] https://botsentinel.com/
[5] https://www.washingtonpost.com/news/the-switch/wp/2018/05/18/fake-facebook-accounts-and-viral-lies-multiply-in-hours-after-santa-fe-school-shooting/?utm_term=.13b276150108

```python
import time, csv, os
from selenium.webdriver import Firefox
from selenium.webdriver.firefox.options import Options
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.common.by import By

opts = Options()
opts.set_headless()
assert opts.headless
browser = Firefox(executable_path = "geckodriver",
                  options = opts)
browser2 = Firefox(executable_path = "geckodriver",
                  options = opts)
browser.get("https://botcheck.me/?apikey=6ecbebd4661b4746&username=theajayjain#query")
inputForm = browser.find_element_by_class_name("search")


bots = []
not_bots = set()
filepath = "Finished Files"
for filename in os.listdir(filepath):
    print
    if filename[len(filename) - 4:] == ".csv":

        print filename
        with open(filepath + "/" + filename) as csvfile:
            readCSV = csv.reader(csvfile, delimiter = ",")
            for row in readCSV:
                bot = row[0]
                if bot != "user" and bot not in bots:
                    inputForm.clear()
                    inputForm.send_keys(bot)
                    inputForm.send_keys(Keys.ENTER)
                    time.sleep(5)
                    result = browser.find_elements_by_tag_name("p")
                    if result[1].text == "Propaganda Bot like Patterns Classified":
                        print bot
                        with open('confirmed_bots.csv', 'a') as outfile:
                            writer = csv.writer(outfile)
                            writer.writerow([bot])
                        outfile.close()
                        bots.append(bot)
                    else:
                        url = "https://botsentinel.com/analyze/embedded?handle="
                        url += bot
                        browser2.get(url)
                        time.sleep(10)
                        botOrNot = browser2.find_elements_by_tag_name("span")[2].text
                        if botOrNot == "Problematic" or botOrNot == "Alarming":
                            print bot
                            with open('confirmed_bots.csv', 'a') as outfile:
                                writer = csv.writer(outfile)
                                writer.writerow([bot])
                            outfile.close()
                            bots.append(bot)
                        else:
                            not_bots.add(bot)
        csvfile.close()
browser.close()
browser2.close()
```

If either source classifies the account as a trollbot account, the program adds the account to our trollbot data set so we can scan for trollbots when analyzing through the hundreds of thousands of tweets during the week leading up to the midterm elections. Using this strategy, we greatly reduce the amount of false positives generated by Botometer via cross-referencing through additional bot detection algorithms.

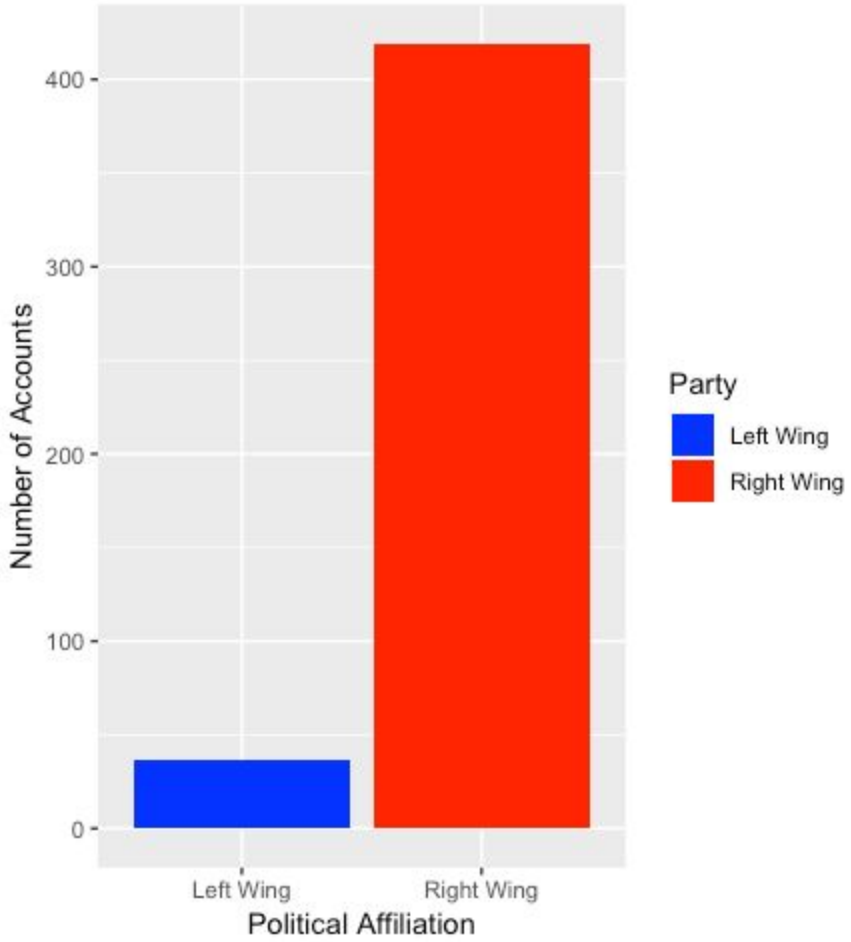
Once we detected Twitter bots and troll accounts, we classified them further depending on a few variables. By the content they produce and retweet, Twitter trollbots generally fall into two categories. Most accounts stick to tweeting or retweeting other bot and troll accounts since that is easy to automate with simple Python scripts as well as tweeting out human-generated text messages. Contents from these accounts are mainly poorly created images filled with propaganda. The other type of account, typified by @patriots4trump (seen above) consistently retweet information from fake news websites like thegatewaypundit.com.

Furthermore, we classified each account manually based on ideology. Classification is done by looking at each trollbot found and determining if the content is right wing or left wing. This process is generally simple, as almost all of the accounts generally state their support for either President Trump or the Democratic Party in their account biography. A majority of accounts pump out political misinformation supporting far-right messages and the Republican Party. However, we detected a small minority of bot and troll accounts tweeting left-wing content as well.

Finally, each tweet is classified based on the candidate or candidates mentioned. Because we are tracking if Twitter bot and troll activity is more active for battleground races, we count the number of mentions

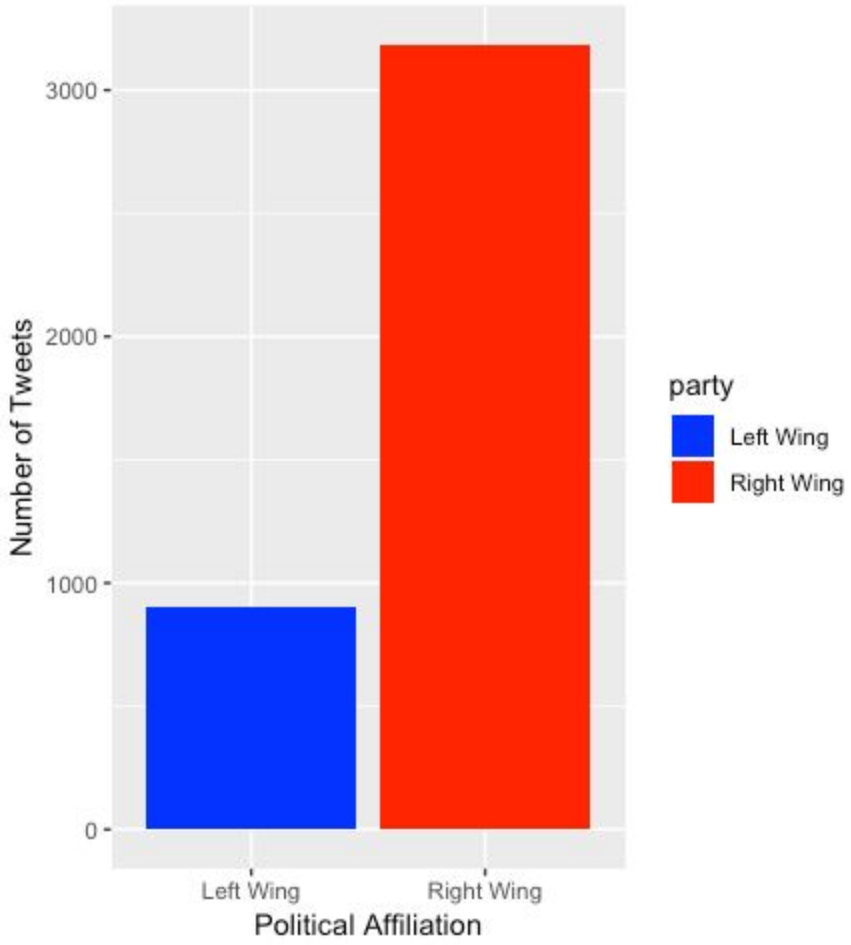
of each Senate candidate and their respective states in tweets from suspicious accounts. In addition, we track the amount of activity between battleground and non-battleground races. Battleground races are defined as the Senate elections in Arizona, Florida, Indiana, Missouri, Montana, Nevada, North Dakota, Tennessee, and Texas.

Our graphs and findings are based on data that has currently been collected and processed. As of December 4th, the programs have filtered through more than 60,000 tweets to find bots. Among the hundreds of thousands of tweets pertaining to the midterms in the week prior to the election, we then extracted only the tweets generated by the trollbot accounts we detected.

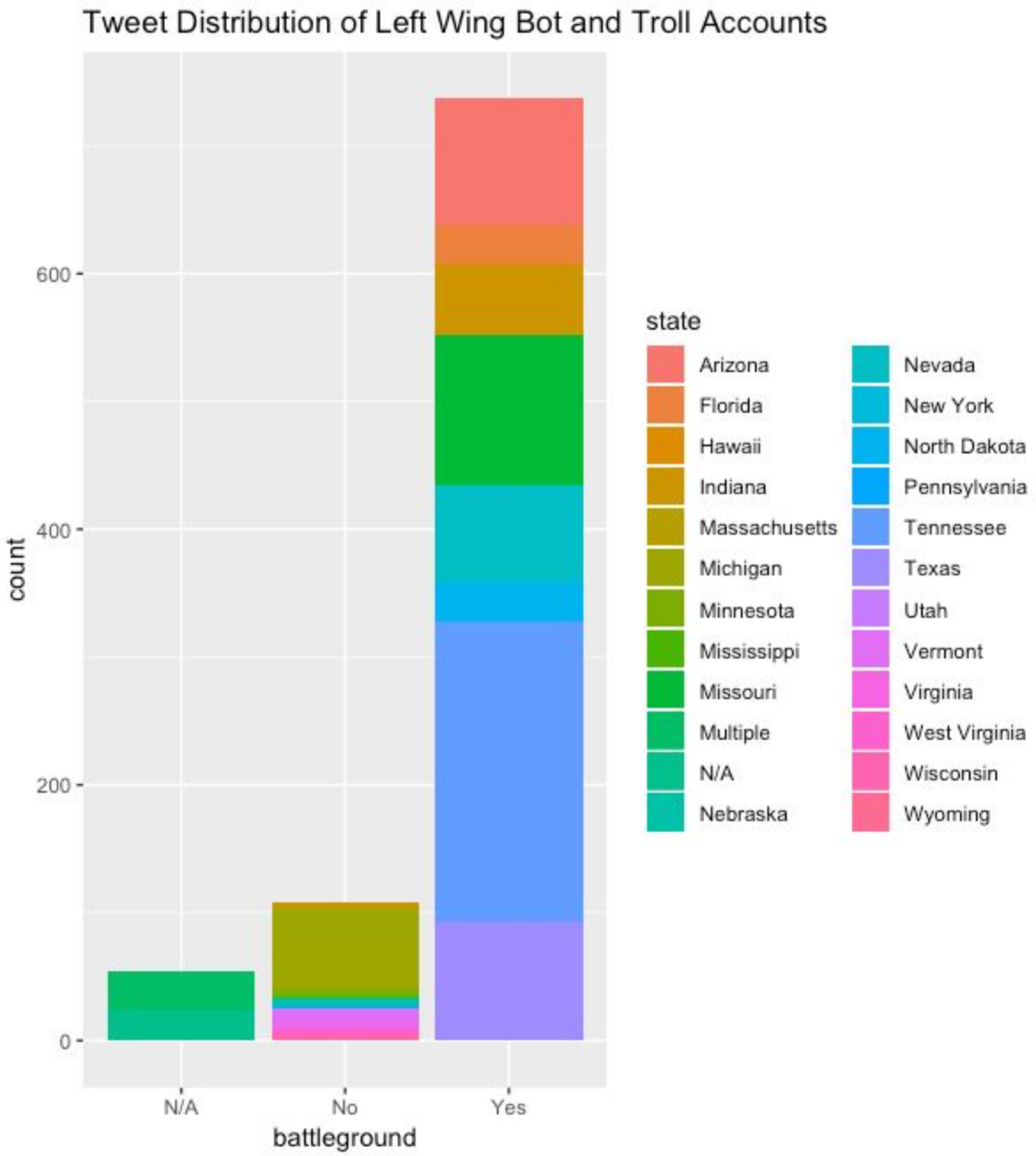


The Python and R programs have currently found 456 Twitter bots and propaganda accounts posting political propaganda on the social media website. Out of those 456 trollbots, 37 of these accounts are posting information favoring Democratic races and candidates. A much greater majority of the bot and propaganda accounts pump out right-wing misinformation. This observation seems to support the trend from the 2016

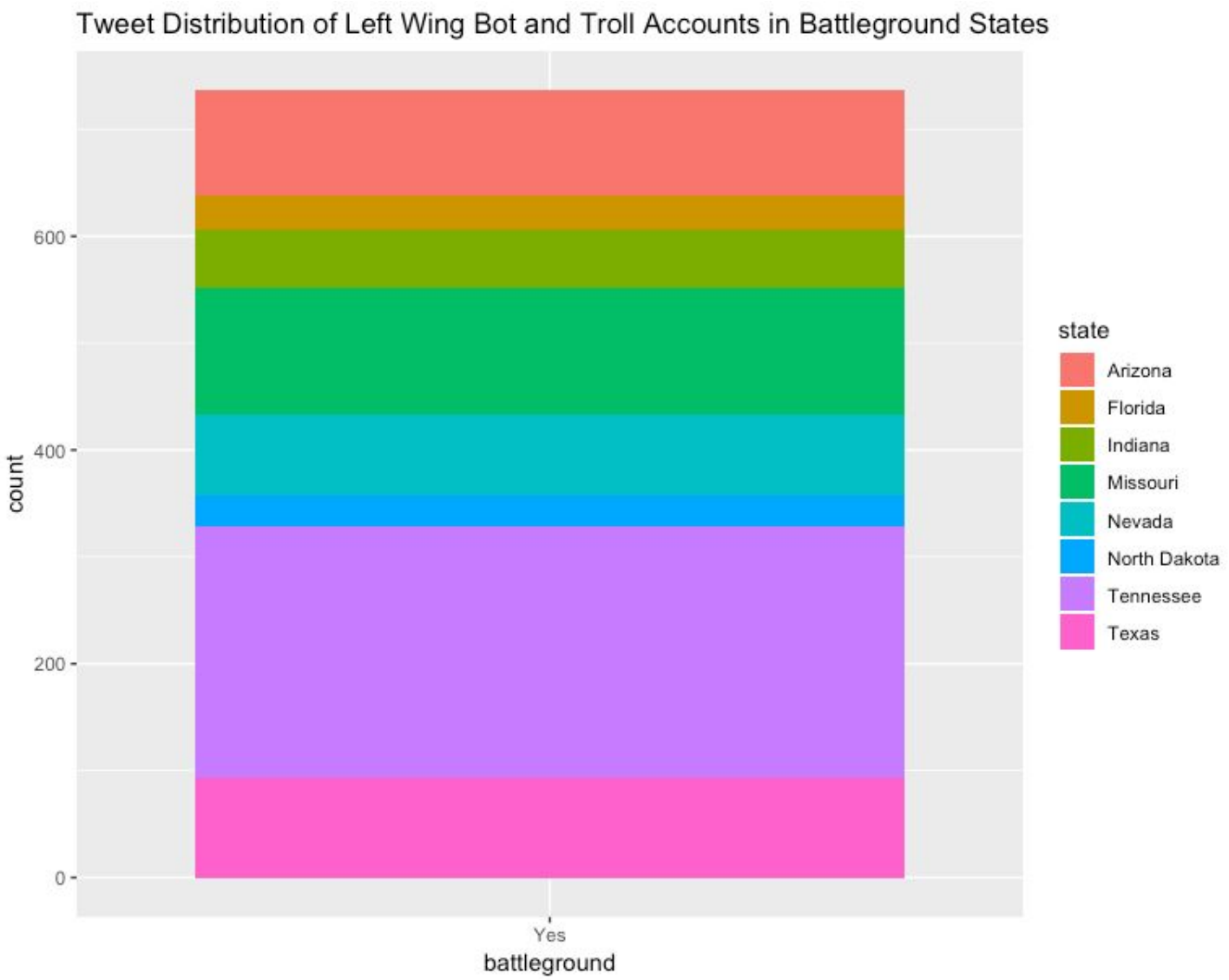
presidential elections that right-wing trollbot activity was more frequent than left-wing ones (Ryan 2018). However, when we analyze our current dataset more closely to look at the number of tweets generated by left- and right- wing accounts, the division between Twitter trollbot activity for the Democrats and Republicans diminishes slightly.



Out of the 4,083 trollbot tweets examined by our program, 899 tweets came from Democratic trollbots. Therefore, eight percent of bot and troll accounts tweeted almost a quarter of the propaganda and misinformation found. Despite lagging behind right-wing trolls in terms of the number of accounts, the left wing troll bots generated far more content per account.



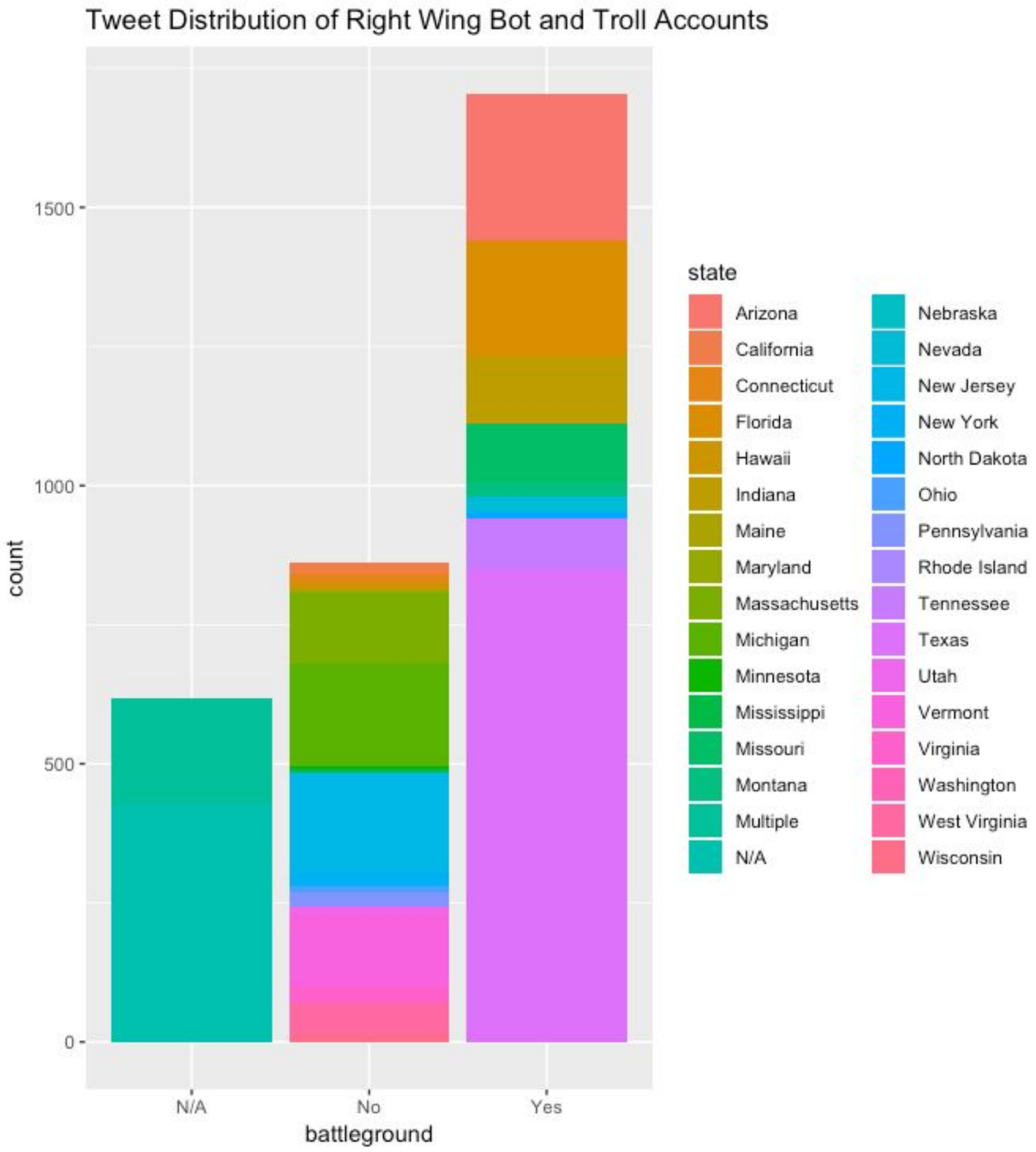
Left-leaning accounts focused almost entirely on tweeting content pertaining to battleground races, with more than eighty percent of tweets containing content relevant to the Senate races in Arizona, Florida, Indiana, Missouri, Montana, Nevada, North Dakota, Tennessee, and Texas. In addition, they did not focus on posting generic content that is not relevant to Senate races, or tweet about multiple races at once (listed under N/A). Interestingly, left-wing trollbots posted more tweets about Elizabeth Warren and the race in Massachusetts than battleground races in Florida, Indiana, and North Dakota, which were seats held by Democrats but lost to the Republican Party in the 2018 midterm elections.



As seen more easily in this graph, left wing accounts attempted to play an offensive strategy in battleground states, with more tweets being generated for seats they wanted to gain from the Republicans (Arizona, Nevada, Tennessee and Texas) than for seats they needed to defend against (Florida, Indiana, Missouri, and North Dakota). The controllers of the left wing propaganda accounts believed that the trollbots would better be utilized for converting Republican voters and getting Democrat voters to the polls in red states.

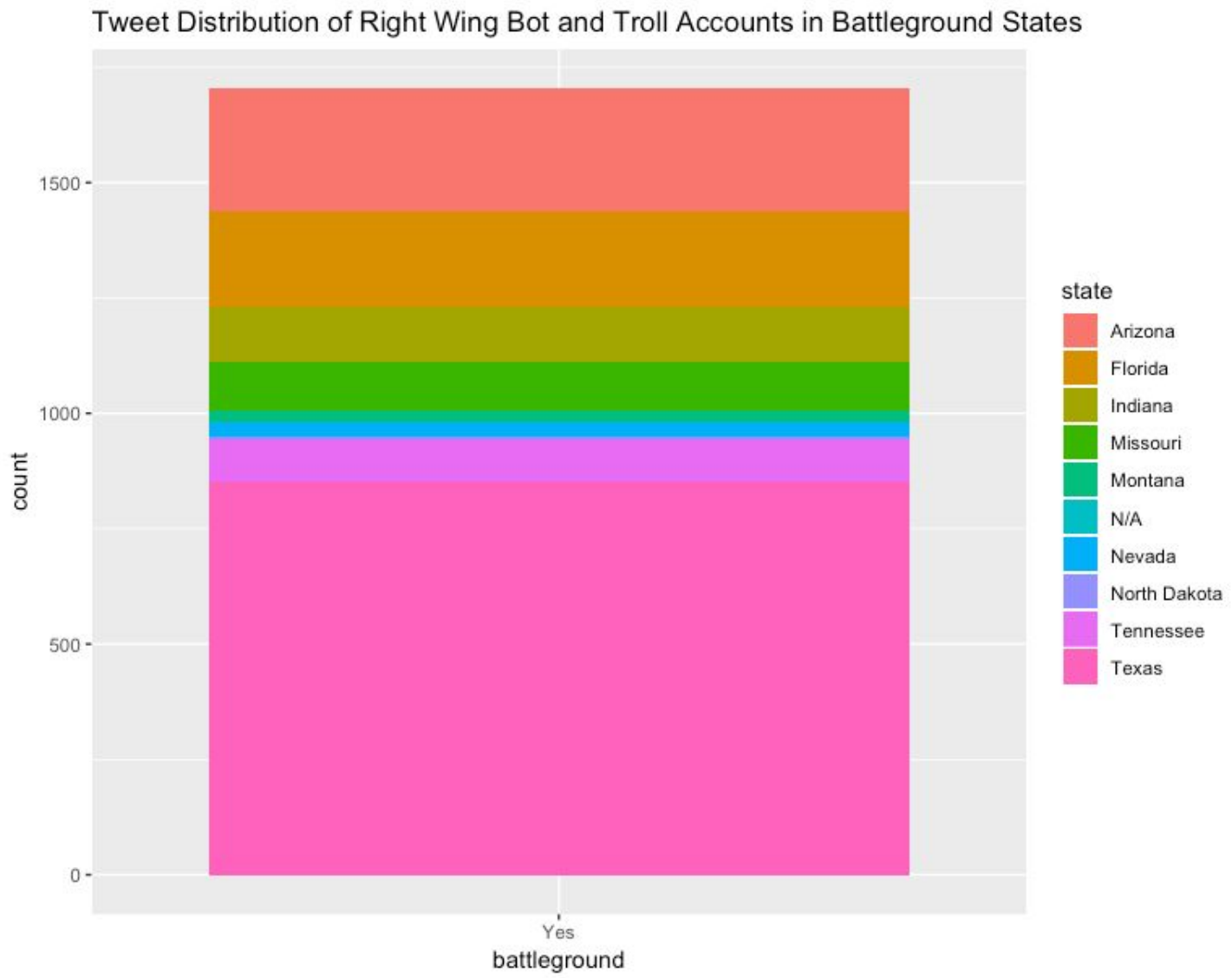
On the other hand, right-wing troll bots used a different strategy. Their distribution of activity between battleground and non-battleground states is more uniform. In addition, many tweets contained multiple hashtags rather than candidate mentions, such as #MAGA and #KAG (hence classified under N/A), mass retweets and

image tweets, or conversations between trollbots. As shown in the following graphs, these activities were not undertaken by left-wing accounts.



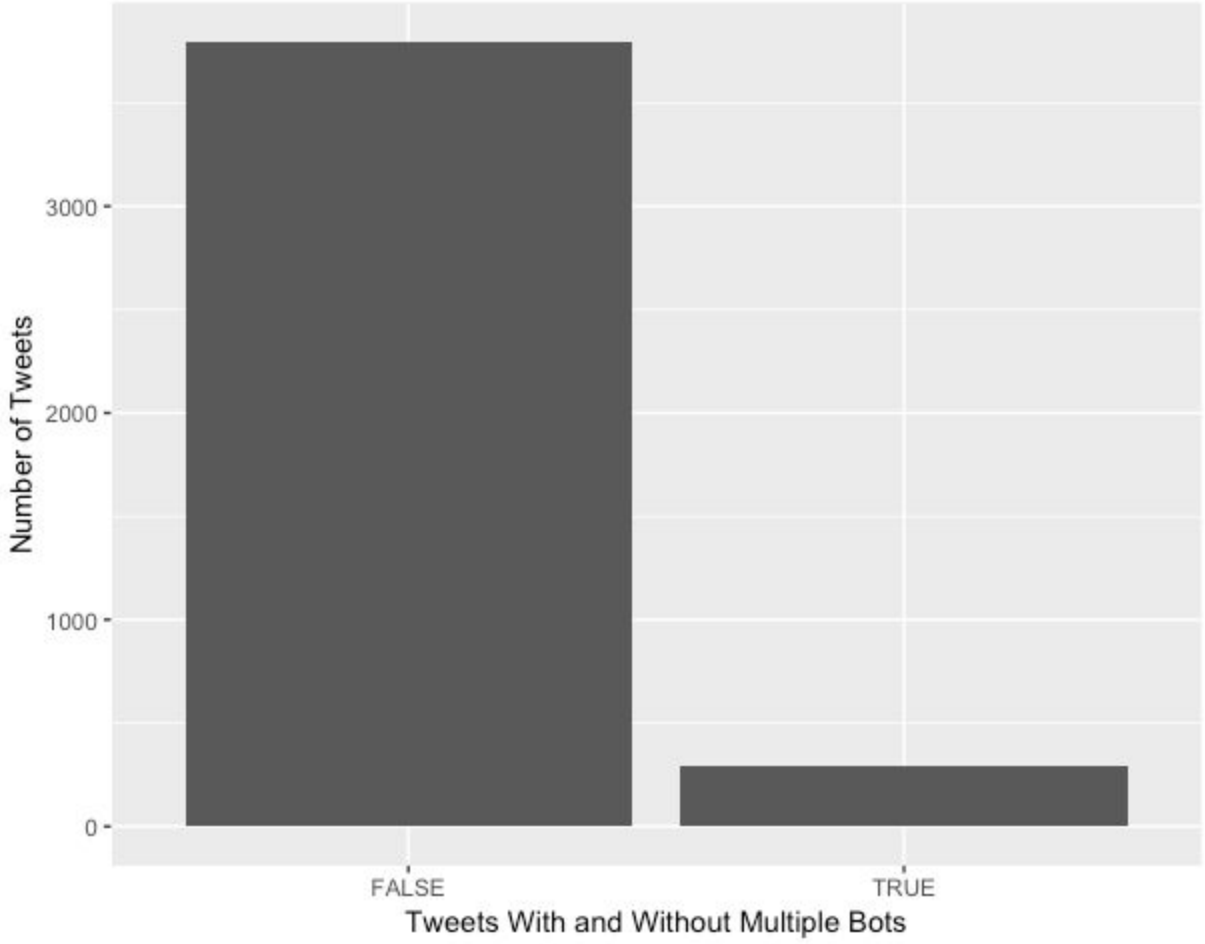
Right-wing trollbots posted heavily in Texas. The amount of right-wing tweets pertaining to the Texas race were almost as many as the those on twenty six non-battleground states combined. In addition, right-wing trollbots were also more likely to tweet about multiple races and candidates at once. Similar to left-wing

trollbots, right-wing accounts placed heavy emphasis on the Massachusetts race with Senator Warren, with more Massachusetts-related content generated than for battleground races in Indiana, Missouri, Nevada, North Dakota, and Tennessee.

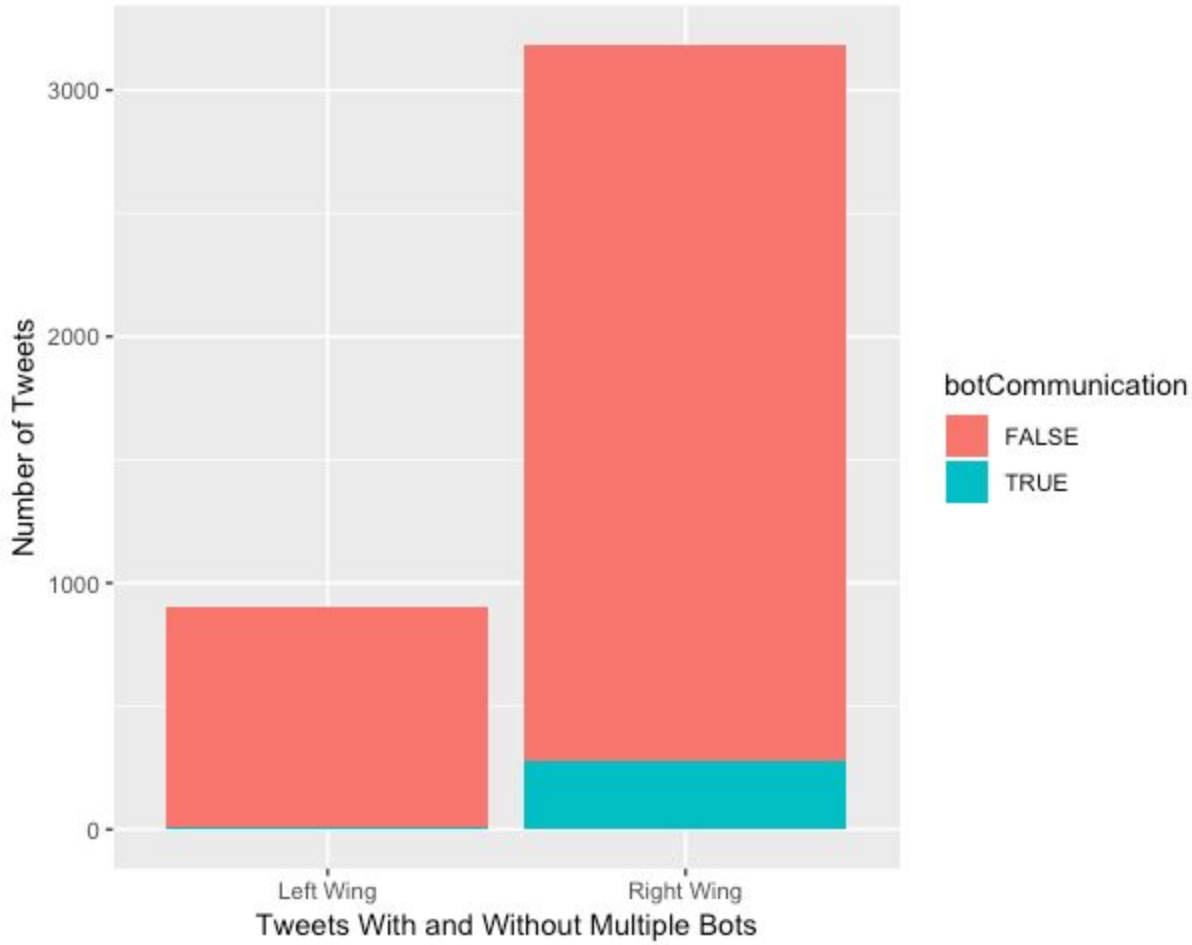


Unlike their left-wing counterparts, right-wing accounts employed a more diverse strategy. A large portion of tweets supported Senator Ted Cruz in the Texas Senate race. Apart from Texas, where Republicans were defending from Representative Beto O'Rourke and the Democrats, and Arizona (where Republicans failed in fending off Representative Kyrsten Sinema), right-wing accounts played an offensive strategy, placing content in states that Republicans were trying to take from Democrats. However, while some of the

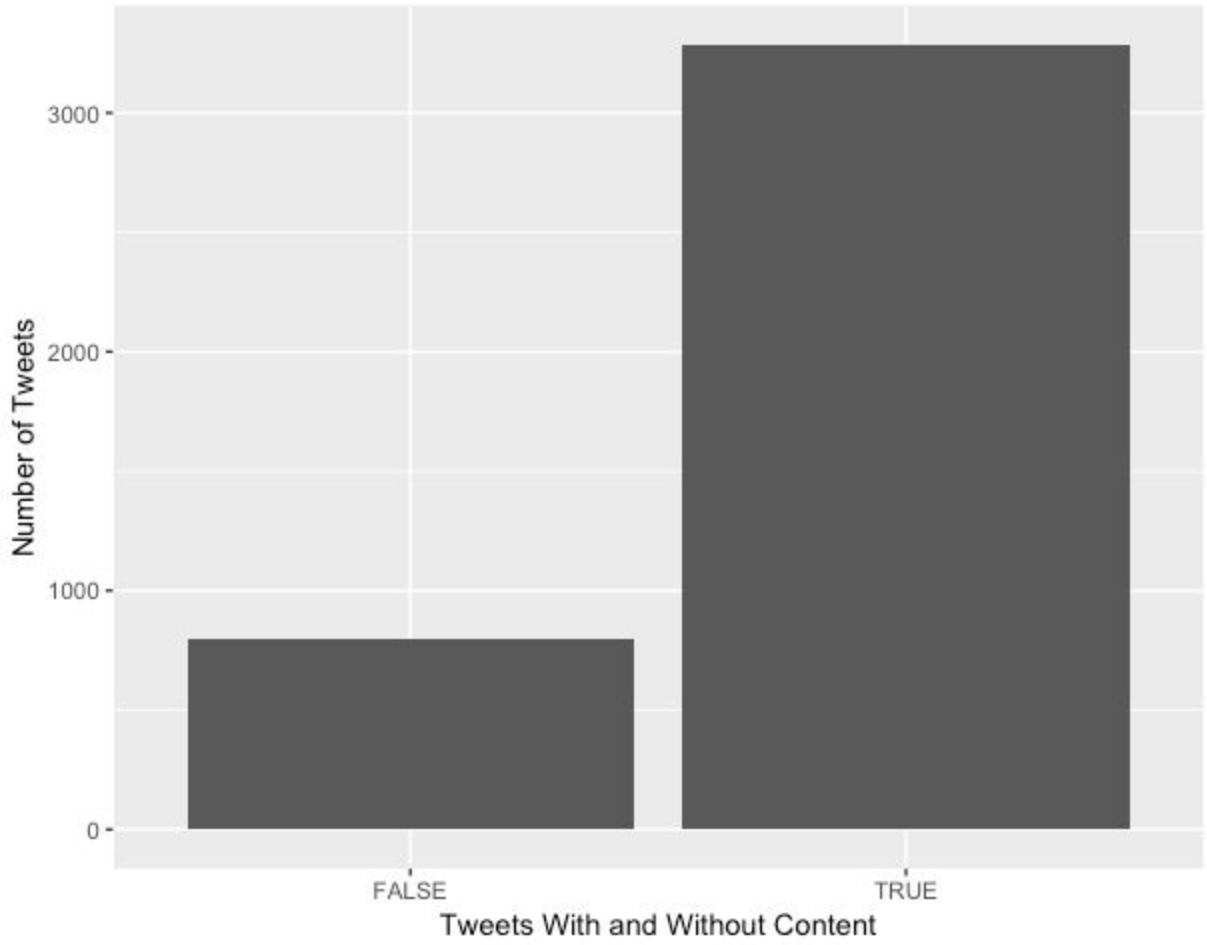
battleground tweets pertain to Democrat-held seats, most tweets are used to defend Republican holds. Right-wing accounts generally used a defense-first, offensive-second strategy to generate content.



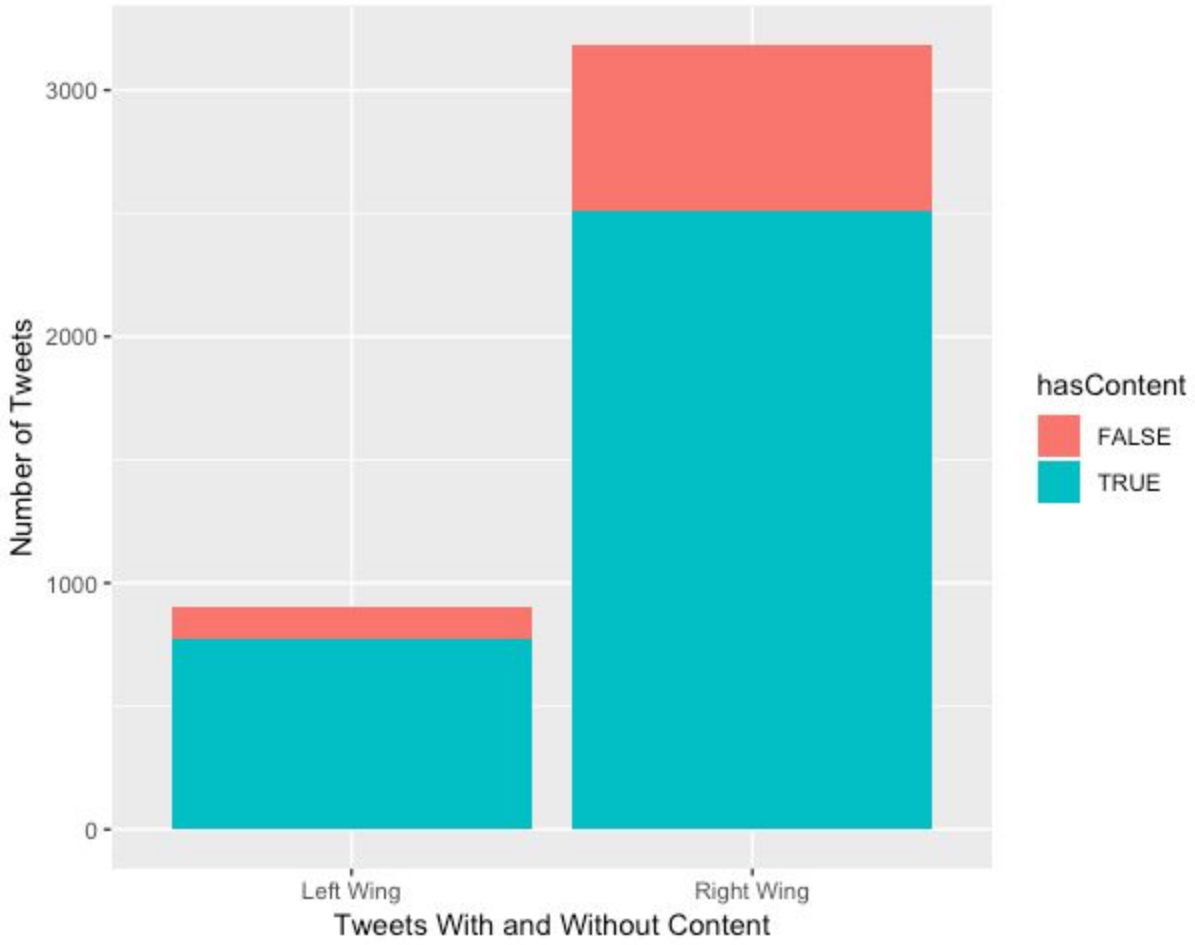
On a more abstract level, tweets were generally found to not include bot interactions, which we are defining as tweets by trollbots interacting with each other through replying to other trollbots' tweets to generate conversations. Only 250 or so tweets were found to have significant interactions between multiple trollbot accounts.



However, all of the tweets that had interaction between multiple bot accounts were right-wing related accounts. Another difference between left-wing and right-wing trollbots is that none of the left-leaning accounts interacted with each other. In contrast, eight percent of all right-leaning content were direct, complex interactions between multiple right-wing accounts.



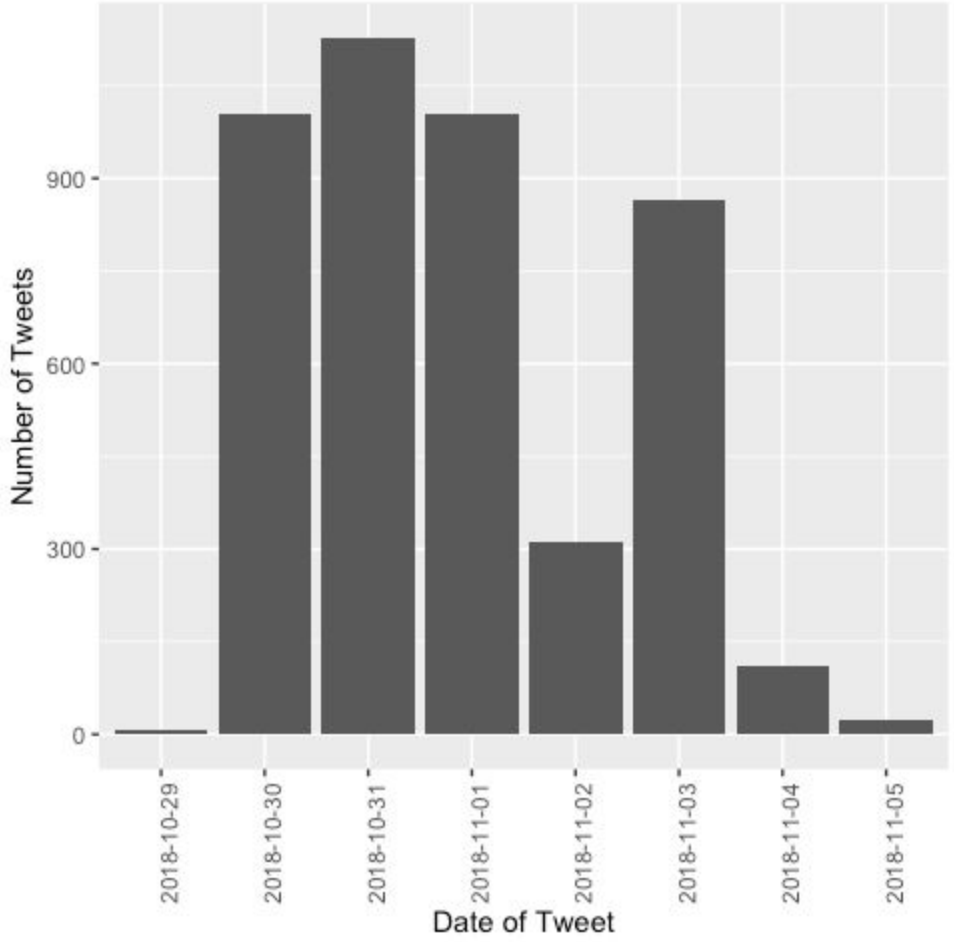
Political trollbot accounts also were found to generate more than simple text-based tweets pertaining to the midterm elections. Monitored accounts focused mainly on tweeting images, articles, and retweets alongside generating text content . Therefore, political trollbots create and push more content than mere text messages in their tweets, and thus use images, articles, videos, and other tweets to spread misinformation.



Both left- and right-wing accounts created a majority of content that are not just text. Text-only content was a mere 16 percent of total content from left-wing accounts, while taking up 23 percent of right-wing content.

In addition to the content-breakdown graphs above, our R program also measured content uniqueness from both left- and right- leaning accounts. Left-wing accounts were found to be more independent, less repetitive and less robotic from each other than right-wing ones. 127 out of 899 tweets from left-wing trollbots had identical text as another tweet, while 187 out of 899 tweets had the same content as another tweet from a left-wing trollbot. Therefore, up to twenty percent of content from left-wing trollbots were reposted either by the same or a different trollbot. This rate, however, sharply increases among right-wing trollbots. Our code concluded that 1112 out of 3184 right-wing tweets had the same text as another right-wing tweet, while 1145 out of 3184 posted the same content . Between 35 and 36 percent of all content from right wing trollbots

accounts were repeated multiple times either on the same account or on a different right-wing trollbot account. This much higher rate, almost double that of the rate of left wing trollbot content repetition, leads our team to conclude that the network of right-wing trollbots is more complex in nature than the left wing counterpart.



Interestingly, the number of tweets generated by accounts did not increase sharply as the election became closer. The week prior to the election had the most tweets, with activity diminishing on the Friday before the election, rising sharply on Saturday, November 3rd, before dying back down to little activity two days before the election. This trend was inconsistent with our team's hypothesis, as we believed that trollbot accounts would pump misinformation as greater rates in the two days before the election. Other than October 29th (which is an outlier in our graph because we began our analysis late that evening), the two days before the midterm election actually had the least trollbot activity. We believe this is because trollbots were targeting early voters and attempting to increase early voting turnout.

Our paper has found multiple significant trends regarding Twitter bot and troll accounts and their activity in the 2018 midterm elections. While there are still more right-wing trollbot accounts than left-wing ones, the left-leaning accounts generated more content per account than their right wing counterparts. In addition, both left- and right-wing bots tweeted heavily about battleground races instead of the non-battleground races. There was more content generated for the nine battleground states combined than for 28 non-battleground states combined from both sides. Notably, both left- and right-leaning accounts tweeted a decent amount of content relevant to Elizabeth Warren and the Massachusetts race. Bots of both ideologies tweeted more about Warren than some battleground races, even though Senator Warren was all but certain to win her election.

Furthermore, left-wing trollbots had more of an offensive strategy than their right-wing counterparts. Left-wing activity was concentrated in elections where Democrats were trying to take Senate seats from Republicans. Other than Missouri, the battleground states held by Republicans all generated more left-wing content than battleground seats with Democratic incumbents. Right-wing trollbot accounts used a more defensive approach that concentrated heavily on Texas. There were almost as many right-wing tweets about Texas than about all of the non-battleground states put together. Outside of Texas, right-wing accounts used a more offensive approach in attempts to sway voters in Democratic-held battleground states. Right-wing accounts are also much more likely to tweet about multiple races at once or tweet generic statements about their party.

Right-wing and left-wing trollbots also slightly differed in the type of content they generated. Left-wing trollbots relied almost exclusively on tweets with articles and images. However, right-wing trollbots also relied heavily on tweets with articles and images rather than just plain text, albeit at a smaller rate than their left wing counterparts.

Bot interaction amongst right-wing trollbots and left-wing trollbots indicated that the network of right-wing trollbots is much more vast and complex. All of the tweets containing multiple bots had right-wing content and no left-wing content. Multiple trollbot posts made up eight percent of right-leaning tweets found by our study. In addition, at 35 percent, right-wing content has a higher likelihood of repetition than left-leaning

ones, which had a rate of up to 20 percent. Therefore, we believe that the left-wing accounts are more likely to be controlled by people, while the right wing accounts consist of both actual people and bots.

Lastly, current results from our study showed the hypothesis of increasing trollbot activity as the midterm election neared to be incorrect. The number of tweets found was somewhat consistent during the first three days of the study, before sharply decreasing the Friday before the election, increasing to similar levels of activity the next day, and then sharply decreasing on both the Sunday and Monday before the midterm elections. Although the ratio of left-wing to right-wing accounts and the content generated for each ideology was similar to the 2016 election, the trollbot accounts used a different strategy than the 2016 election cycle by greatly limiting the content produced as the midterm elections approached.

Sources

Emilio Ferrara, Onur Varol, Clayton Davis, Filippo Menczer, and Alessandro Flammini. 2016. The rise of social bots. Commun. ACM 59, 7 (June 2016), 96-104. DOI: https://doi.org/10.1145/2818717

Bhat, Ash, and Rohan Phadte. "An Analysis of Propaganda Bots on Twitter." Medium, RobHat Labs, 31 Oct. 2017, www.medium.com/@robhat/an-analysis-of-propaganda-bots-on-twitter-7b7ec57256ae.

Smith, Ryan. "Right-Wing Twitter Propaganda Bots Pumping up Jeanne Ives Campaign for Governor, Study Finds." Chicago Reader, 14 Mar. 2018, www.chicagoreader.com/Bleader/archives/2018/03/14/right-wing-twitter-propaganda-bots-pumping-up-jeanne-ives-campaign-for-governor-study-finds.

White Jr., Lamar (LamarWhiteJr). "At 2 in the morning, hundreds of Ted Cruz supporters all woke up to issue the same exact statement on Twitter. Nothing suspicious about this. Nope. These are all just proud, patriotic American citizens, all deep in the heart of Texas." 13 September 2018, 12:20 a.m. Tweet.

Silverman, Craig. "This Analysis Shows How Viral Fake Election News Stories Outperformed Real News On Facebook." BuzzFeed News, BuzzFeed News, 16 Nov. 2016, www.buzzfeednews.com/article/craigsilverman/viral-fake-election-news-outperformed-real-news-on-facebook.

Abeshouse, Bob. "Troll Factories, Bots and Fake News: Inside the Wild West of Social Media." Al Jazeera, Al Jazeera, 8 Feb. 2018, www.aljazeera.com/blogs/americas/2018/02/troll-factories-bots-fake-news-wild-west-social-media-180207061815575.html.