Big Data
Volume 5 Number 4, 2017

DOI: 10.1089/big.2017.0038

ORIGINAL ARTICLE

Detecting Bots on Russian Political Twitter

Denis Stukal[1,2] Sergey Sanovich[1,2] Richard Bonneau[2–4] and Joshua A. Tucker[1–3,*]

Abstract

Automated and semiautomated Twitter accounts, bots, have recently gained significant public attention due to their potential interference in the political realm. In this study, we develop a methodology for detecting bots on Twitter using an ensemble of classifiers and apply it to study bot activity within political discussions in the Russian Twittersphere. We focus on the interval from February 2014 to December 2015, an especially consequential period in Russian politics. Among accounts actively Tweeting about Russian politics, we find that on the majority of days, the proportion of Tweets produced by bots exceeds 50%. We reveal bot characteristics that distinguish them from humans in this corpus, and find that the software platform used for Tweeting is among the best predictors of bots. Finally, we find suggestive evidence that one prominent activity that bots were involved in on Russian political Twitter is the spread of news stories and promotion of media who produce them.

Keywords: bot detection; ensemble methods; machine learning; Russia; Twitter

Introduction

Social media platforms have been playing an increasingly important role in public opinion formation in democratic and nondemocratic countries alike. Often providing a platform for political learning and organizing, social media can foster more open political systems and bring better political outcomes.[1,2] However, recent developments, including developments in stable democracies like the United States, show that social media can facilitate polarization and the spread of misinformation.[3] Social media accounts whose content is generated by a computer program rather than a human being (also called bots) can be particularly detrimental to the social media ecology as meaningful discussion between peers can easily fade in the presence of artificially boosted stories and opinions.[4,5]

Although not all Twitter bots are created for malicious purposes,[6] even benign ones can eventually have negative social effects.[7] Their effect can be especially devastating in nondemocratic regimes, as social media often provide the only opportunity for the opposition to mobilize supporters and the public to access alternative opinions and unfiltered news.[8,9] Eliminating these opportunities by flooding the social media commons with a barrage of distracting or purely fake news comprises an important misinformation/censorship strategy for many modern autocracies.[10–12] Besides suppressing online opposition, bots can also be used for propaganda purposes, for example, to help official political messages reach broader audiences at home and abroad or to influence what messages are widely distributed or trending.

Propaganda has always been an integral part of politics, from early Mesopotamia through this day.[13,14] Modern nondemocratic regimes have been especially active purveyors of propaganda.[15–17] However, the recent rise of social media creates a plethora of new, and yet understudied, ways to automate the dissemination of propaganda and information using bots and paid trolls. In this article, we help narrow this gap by focusing on the task of detecting and analyzing the activity of bots in the Russian political Twittersphere from 2014 to 2015 as a case study. Our interest in automated accounts in Russia, and the Russian speaking Twitterverse, is motivated by a sharp and well-documented increase in both the domestic and international online activity of Russian hackers, trolls, and bots.[18–23]

As we are interested in studying both how bots were used during specific political events and how bot activity changes over time, we develop a methodology that enables us to study account activity retrospectively,

[1]*Department of Politics, New York University, New York, New York.*
[2]*Social Media and Political Participation (SMaPP) Lab, New York University, New York, New York.*
[3]*New York University Center for Data Science, New York University, New York, New York.*
[4]*Flatiron Institute, Simons Foundation, New York, New York.*

*Address correspondence to: *Joshua A. Tucker, Department of Politics, New York University, 19 West 4th Street, New York, New York 10012*, E-mail: joshua.tucker@nyu.edu

including accounts that were deleted or suspended in the course of data collection. This is a key difference that distinguishes our approach from publicly available tools (e.g., BotOrNot[24]) that allow users to detect bots only among live Twitter accounts. In addition, while we focus on the Russian Twittersphere in this article, we aim to build a tool that could be applicable in other contexts too, and thus refrain from using text-related features for bot identification purposes.

We make a novel methodological contribution by proposing the unanimous voting rule of supervised classifiers as the principle for building an ensemble classifier for bot detection purposes. In other words, we propose a bot detection ensemble algorithm that classifies an account as a bot only if all of its component algorithms do the same. We show that our approach produces a tool with extremely high precision and sufficiently high recall (or, in other words, is extremely accurate when labeling an account as a bot and is also highly accurate in finding bots among those that are in the sample). We also identify the most informative account features and use them to show how bots differ from humans on Twitter.

Substantively, we find that bots represent a very large portion of the Russian political Twittersphere, that contrary to received wisdom[25] most bots in our collection do not Tweet much more often than humans (at least about politics), and that the most common use of bots in the period of time we examined in Russian political Twitter was to share news headlines, although not necessarily links to the news stories. This in turn suggests the possibility that a primary use of bots in Russia from 2014 to 2015 was to attempt to manipulate search rankings.

Previous Research

The use of bots in political campaigns and propaganda dates back to at least the mid-2000s, when they were used in the United States for linking politicians' names to unflattering stories about them in search engine results.[26] However, it was only in recent years that both the scope[27] and sophistication[28] of bot activity on Twitter increased to such an extent that it began to draw serious attention from political actors, including state actors. The official Twitter estimates released in 2016 claim that around 8.5% of Twitter users are bots.[29] The U.S. Defense Advanced Research Projects Agency launched its own bot detection program in 2015.[29]

A variety of approaches have been proposed to detect bots. One class of approaches involves using unsupervised machine learning methods for bot detection. These unsupervised methods rely on the fact that bots are automated programs, and thus should have a much higher resemblance to each other than humans. Cresci et al.[30] propose to use a sequential data mining technique based on coding account activities over time with sequences of symbols, each representing a different type of activity, and finding the longest common substring within accounts' strings. Chavoshi et al.[31] apply Dynamic Time Warping distance for time series of accounts' Tweeting activity to identify "correlated" accounts in real time. Although these approaches are promising for the purpose of bot detection on Twitter, they are not fully applicable to topical datasets collected through keyword searches from the streaming API that allow storing only Tweets meeting certain criteria and omit others.

An alternative approach is to employ human coding to create a training set of Twitter accounts that can then be used for training a supervised learning algorithm. Ratkiewicz et al.[32] used Adaptive Boosting and support vector machines (SVMs) to discriminate between information distributed by humans and bots. Chu et al.[33] collected a set of 6000 accounts from human, cyborg, and bot categories to build a Random Forest classifier. They applied their classifier to a dataset of 500,000 users and found that 53.2% of users were humans, whereas 36.2% were cyborgs and 10.5% were bots. Oentaryo et al.[6] used a set of 1600 training accounts to build a number of classifiers and found the most discriminative features.

We build on the research that uses supervised learning and develop a bot detection tool that is able to identify bots with very high levels of precision, while maintaining high recall (formally, precision is the probability that an account predicted to be a bot is actually a bot (i.e., no false positives); recall is the probability that an account that is actually a bot will be classified as such (i.e., no false negatives)). We then apply our method to a comprehensive dataset, including almost 2 years of data collected from Russian political Twitter. This allows us to characterize the population of accounts Tweeting about major topics in Russia politics during a highly consequential period of recent Russian history.

Methodology

As our goal is to analyze bot activity in the political segment of Russian Twitter at specific and substantively important points in time, we are unable to make use of bot detection algorithms readily available online (including BotOrNot), since they do not allow the user to study account activity retrospectively or analyze deleted or suspended accounts. Instead, we develop

a bot detection methodology based on dynamic account characteristics that can be measured every time a given account Tweets, and thus is applicable to a dataset of stored Tweets and metadata regardless of whether corresponding accounts are still available or active. Such an approach is useful for both retrospective studies and the analysis of the evolution of Twitter accounts and communities.

Our bot detection methodology uses supervised learning methods to distinguish bots from other account types. Initially, we coded a sample of ~1000 accounts that we used to develop a 13-category account typology of bots and humans on Twitter (see details in Sanovich et al.[8]). Using these categories, we created a detailed coding manual describing the distinguishing characteristics of each account type and trained 50 coders—all undergraduate social sciences students from Moscow—who are native Russian speakers familiar with Twitter.

The training process included two to three trial rounds of supervised coding where our coders label around 20 accounts we select for them. These preselected accounts include both cases where account types are easy to identify (e.g., accounts with a default background picture and no user description that only Retweet posts) and more difficult accounts that require careful investigation. Our coders were instructed to pay attention to a number of account characteristics, including a link to a different social network that might contain a more detailed biographical information, a set of personal pictures in the user's media collection, and interactions with other users that exhibit meaningful human reactions or emotions. In the rare cases where an account label was still unclear to the coders, we instructed them to err on the side of the account belonging to a human being. This is the first in a series of steps we take to increase the confidence that bots we identify as such are indeed bots, not humans.

After the training rounds are over, we randomly split the coders into groups of five people, each of whom is unaware of the other members of their group and is instructed to work independently. All workers from a group are coding the same set of accounts (on average, 50 accounts per week). Overall, the coding took 15 weeks, although not all coders joined the project at the same time, and not all groups worked simultaneously. In total, 3130 accounts were coded, but, as we explain in detail below in this section, we provide our bot detection algorithm with a high-quality training set of 1068 accounts that are coded with a high intercoder reliability and have at least 10 Tweets in our collection.

The limitations of this approach should be noted. The decision to focus on this training set of 1068 accounts with high intercoder reliability implies that the harder and more difficult to detect bots may be invisible to our algorithm. No means of bot detection is perfect and all involve trade-offs; we chose to minimize the risk of incorrectly characterizing a human account as a bot (i.e., a false positive) at the potential cost of missing classifying some accounts that are bots as such. Thus, the proposed algorithm should be understood as a tool that advantages identifying a set of accounts that are indeed bots with a higher degree of certainty, at the expense of potentially missing some bots within a given sample.

Although our overall coding framework can be used to break down Twitter accounts into 13 different subtypes that we identify in other work,[8] our focus in this article is simply on whether we can separate out bots from humans. Thus, we collapse all subtypes into two broad categories: all types of bots and the rest. Since coders are much more likely to disagree on the subtype of bot as opposed to whether an account is operated by a human or not, this further increases our confidence in the results.

To ensure both long-term reproducibility and the coders' ability to access and code accounts from our collection (regardless of whether they are still available online), we create artificial account snapshots that reproduce Twitter account pages at the time of data collection. An account snapshot contains all the Tweets from that account in our collection. If a Tweet is still available online at the time of coding, it is shown exactly the same way as on Twitter; otherwise it is shown as plain text. We also use metadata (the user picture, nickname, description, background picture, and number of friends and followers, among others) in snapshots to provide coders with as familiar a Twitter experience as possible.*

The labeling of accounts is, of course, subject to limitations derived from the imperfect human ability to distinguish between robots and human beings.[34] To reduce potential bias from human error, we compute Hamming distances[35] for each pair of coders within each group in every round, and remove the results produced by the most dissimilar coder (in the rare case of a tie, we randomly select the coder to be removed). If three out of four remaining coders assign an identical category to an account, we label it with that category in the labeled

*An example of a Twitter snapshot used is available at: www.denisstukal.com/uploads/8/4/7/0/84708866/snapshot_example.png.

set. Otherwise, we mark it as unclear and remove from the labeled set. This approach guarantees that the intercoder reliability for all accounts in the labeled set is at least 75%. It is also relatively robust to human errors, since it ignores coding decisions from the most extreme coders and allows one coder to disagree with the rest without affecting the final result. An added benefit of this approach is that it does not affect the outcome if coders agree on the labels they assign to accounts.

The overall workflow comprises a number of steps. First, we collect Tweets with the Tweet- and user-level information available through the Twitter API. We store all Tweets that mention a political word or hashtag from a large list of keywords (Appendix A). This approach to data collection allows us to build a comprehensive dataset of accounts Tweeting about politics in Russia.

Second, we extract a large number of features using metadata and data on the account Tweeting activity (see details in *Data* section).

Then, we select a random sample of accounts and use human coders to create a labeled set by categorizing selected accounts into account types as already described. To reiterate, although the hardest cases on which human coders disagree the most end up being invisible for the classifier, this has the advantage of ensuring that the training set approximates the ground truth as close as possible and allows us to proceed with a conservative approach to bot detection that emphasizes precision over recall.

As the coding proceeded, we discovered that taking a random sample of accounts produced a highly imbalanced labeled set with a large prevalence of bots. To attenuate the pitfalls associated with analyzing imbalanced datasets, such as poor performance of evaluation metrics and problems with class separability,[36–38] we use additional adaptive sampling by training a ridge logistic regression with our feature set, predicting account types for the unlabeled set, and augmenting the labeled set with a new random sample of 300 instances that are predicted to be non-bots with a probability of at least 0.7. Our motivation was to increase the number of non-bot instances to produce a more representative sample from this class and thus create a more generalizable classifier.

At the fourth step, we produce balanced training and test sets. Even though we employ oversampling to increase the number of non-bots in the labeled set, it is still highly imbalanced with $N_b^{lab} \gg N_{nb}^{lab}$, where N_b^{lab} and N_{nb}^{lab} refer to the number of labeled bots and non-bots, respectively. Thus, we sample 50 bots and non-bots into the test set and use the remaining $N_{nb}^{tr} = N_{nb}^{lab} - 50$ non-bots in the training set, adding to them a random sample of labeled bots of the same size. Thus, our training sample size is $2N_{nb}^{tr}$, which includes only a subset of all labeled bots. To avoid capitalizing on chance, we repeat this procedure 10 times, thus obtaining 10 classifiers that are independently trained on 10 different and balanced training sets. We assess classifiers' performance by averaging over the performance values obtained on each of the test sets.

Next, we train four classifiers (logistic regression with ridge regularization, gradient boosted tree, SVM with radial basis function kernel, and Stagewise Additive Modeling using Multiclass exponential loss function) using fivefold cross-validation on each training set. Thus, each of the 10 test sets remains unseen by the classifiers trained on the respective training set.

Since our main substantive goal is to study patterns of bot activity, false positives will be more detrimental to our planned study than false negatives. Thus, we continue to adopt a conservative approach to bot detection, building an ensemble classifier based on a voting rule. Previous research on the use of these types of procedures in political science and economics has explored various voting rules.[39,40] Although the most common types of rules include plurality[41] and majority,[42] theoretical and experimental results do not provide definitive guidelines about the choice of a voting rule,[43] which motivates us to use the most conservative rule: unanimous voting. Thus, the voting rule that we employ in this study predicts an account to be a bot if all classifiers predict it to be a bot. This helps us minimize the chances of having human accounts in the set of predicted bots.

The next step is to evaluate each classifier on the data it did not see in the process of training, which we repeat 10 times. We then average model performance across the 10 test sets to produce our final estimates.

We then apply the ensemble classifiers trained on 10 different training sets to the collection of Twitter accounts and obtain the population of political bots on Russian Twitter. To produce the final classification, we employ two approaches. First, we use majority rule: if 6 out of 10 ensembles predict an account to be a bot, we label it that way (the same rule applies also to non-bots). Second, as a robustness check, we use a conservative approach with the unanimous rule again and claim that an account is a bot if all training samples predict it to be a bot. Finally, we analyze patterns of bot activity over time using both classification rules.

Data

We collect data through the Twitter Streaming API. Our dataset consists of a collection of Tweets (along with associated metadata) that feature words or hashtags from a list of 86 politically relevant keywords and hashtags related to Russian politics (Appendix A). To both develop a highly accurate bot detection tool and assess its ability to identify bots despite the possible increase in their sophistication over time, we collect data for two different periods of time. As we show in the Empirical Results section, Twitter accounts exhibit a consistent behavior throughout both periods, which allows us to pool both datasets and build a classifier using all available data. We first collected data between February 6 and October 1, 2014. This part of our data collection includes almost 18 million Tweets from around 3.8 million Twitter accounts. Then, we collected data between January 30 and December 31, 2015, thus including an additional 18 million Tweets from 1.5 million accounts. These two periods cover a tumultuous period in Russian politics, including, but not limited to, the Russian annexation of Crimea, conflict in Eastern Ukraine, and the murder of a Russian opposition leader in front of the Kremlin.

As we are interested in detecting bots that are active in the Russian political Twittersphere, we restrict our attention to accounts that Tweet in Russian and chose Russian in the account language settings (hereafter referred to as the "account interface language"). The information about an account interface language is part of the metadata associated with collected Tweets. Conditional on having multiple Tweets from an account in our collection, we are able to track changes in its metadata over time, including changes in its interface language. We define an account to be Russian if it has a Russian interface for at least 75% of Tweets in our collection. Since the accounts that Tweet about politics only occasionally are not of substantive interest for us, we further restrict our analysis to accounts with at least 10 Tweets in either period of our collection. The distribution of the number of accounts and Tweets for different cutoff values is reported in Table 1.

Overall, the subset of data we focus on in this study includes 230,000 Russian accounts with around 15 million Tweets. The labeled set consists of 1068 accounts coded by human coders as already described. Among these labeled accounts, there are 249 non-bots. We randomly select 50 accounts for the test set and use the remaining 199 accounts for the training set, then add same numbers of labeled bots, and repeat this process 10 times, as described in the Methodology section.

Table 1. Russian accounts with different thresholds

Threshold	*Period 1*	*Period 2*	*Both periods*
1	880,000 (7,770,000)	800,000 (10,910,000)	1,400,000 (18,650,000)
3	430,000 (7,190,000)	340,000 (10,320,000)	680,000 (17,710,000)
10	100,000 (5,520,000)	130,000 (9,310,000)	230,000 (15,400,00)
50	22,000 (3,960,000)	31,000 (7,380,000)	50,000 (11,900,000)
100	11,000 (3,230,000)	16,000 (6,330,000)	27,000 (10,320,000)
500	1000 (1,160,000)	3000 (3,590,000)	4500 (5,730,000)
1000	280 (660,000)	1100 (2,240,000)	1800 (3,870,000)
5000	15 (240,000)	30 (420,000)	66 (830,000)

Main entries are numbers of Russian accounts with at least k Tweets in our collection for a given period, where k is the threshold value. The number of Tweets from these accounts are in parentheses. Rounded numbers are reported. Period 1 runs from February 6 to October 1, 2014. Period 2 is between January 30 and December 31, 2015. Numbers for "Both periods" are not sums of the numbers for the two previous columns because first, accounts in the two periods overlap, and, second, an account with, for example, less than 50 Tweets in two periods separately can have more than 50 Tweets when pooling data from the two periods. *Source:* Authors' calculations based on data collected from Twitter API.

To detect bots, we use a set of 42 features that either are included with the metadata accompanying each Tweet when downloaded from the API or characterize account Tweeting activity (Appendix B for the complete list of features). To make our method applicable beyond the Russian case for future research, we do not use any text-related features for bot detection purposes.

Empirical Results

Appendix Table C1 (located in Appendix C) presents the average cross-validated parameters for the trained classifiers across 10 training sets, while Table 2 shows the average performance metrics for the trained classifiers. The latter table reveals that all four methods have a relatively high precision with slightly lower recall levels. However, as we aim to minimize the probability of false positives, we adopt a conservative approach to bot detection and apply an ensemble of classifiers based on the unanimity voting rule: only those accounts that are predicted to be bots by all classifiers are eventually

Table 2. Performance metrics over 10 test sets

	Precision	*Recall*	*Specificity*
Ridge logistic regression	0.92	0.87	0.92
SVM (RBF kernel)	0.90	0.84	0.90
XGBoost	0.87	0.91	0.87
SAMME	0.92	0.89	0.92
Ensemble (unanimous voting: bots)	0.99	0.77	0.99
Ensemble (unanimous voting: non-bots)	0.93	0.79	0.94

Entries are performance metrics averaged over 10 test sets. $Precision = \widehat{Pr}(bot|bot)$. $Recall = \widehat{Pr}(bot|bot)$. $Specificity = \widehat{Pr}(nonbot|nonbot)$.

Source: Authors' calculations based on data collected from Twitter API.

RBF, radial basis function; SAMME, Stagewise Additive Modeling using Multiclass exponential loss function; SVM, support vector machine.

labeled as bots. Such an approach allows us to achieve precision in excess of 0.95, while keeping recall at a lower, but still satisfactorily high, level, ensuring that the bot activity patterns that we do detect are indeed those of bots.

To aggregate the results from the ensemble classifier across 10 training sets, we employ two approaches. First, we apply the majority rule on top of unanimous ensembles ("majority unanimous rule") and label an account as a bot if at least half of the ensembles have it coded as a bot. Second, as a robustness check, we use the unanimous rule again ("double unanimous" rule, whereby the account is only coded as a bot if all of the ensembles have it coded as a bot), which ensures the highest possible robustness, but risks underestimating the proportion of bots.

As one can observe from Figure 1, bots that are detected by the majority voting of our ensemble classifiers were largely dormant (or nonexistent) until March 2014, when the bot proportion exploded and hovered over 40% of Tweeting accounts per day, which coincided in time with the Crimean status referendum that resulted in the Russian annexation of the peninsula. This proportion kept growing steadily and in late 2014 exceeded 80%. In the second period under study, in 2015, the daily bot percentage mostly remained between 50% and 80%. We find similar patterns for the proportions of Tweets posted by bots.

As expected, we get a more conservative estimate of the bot proportion when we apply the unanimous-unanimous rule (Appendix C, Appendix Fig. C1), although the pattern of activity remains the same: a dramatic increase in March 2014, a gradual growth in 2014, and stabilization of daily Tweet and bot percentages in 2015. Overall, Spearman rank correlation coefficients between sequences of daily percentages from the majority–unanimous and double unanimous rules are 0.87 for account shares and 0.77 for Tweet percentages, thus confirming the overall consistency of the revealed patterns.

Interestingly, Figure 2, which provides more details on the number of Tweets per account for bots and non-bots identified using the majority rule, illustrates that bots do not necessarily Tweet more frequently than humans or official accounts. Although the daily ratio of the number of Tweets to the number of Tweeting accounts is typically larger for bots than for humans, this is not always the case. Indeed, the ratio ranges from 1.07 to 5.20 for bots, and lies between 1 and 7.85 for non-bots. Thus, even when bots Tweet more than non-bots, the difference in ratios is not as dramatic as one could expect, given the fact that bots are automated programs and could be easily set up to Tweet as often as their creators would like. Although beyond the scope of this article, it is worth noting that two potential explanations for bots not Tweeting all that often would be either to try to avoid detection by Twitter or because the bot activity is tied to something that does not happen too frequently. The latter explanation could be consistent with bots that are programmed to Tweet the headline of a newly published newspaper article in some prearranged category. Of course, it is also possible that the bots are indeed Tweeting more frequently, but not about topics that allow these Tweets to be captured by our search criteria; the nature of our research design simply does not permit us to rule out this possibility.

If bots do not have dramatic differences from non-bots in terms of how often they Tweet about politics, how are they different? We extract feature importance and respective coefficients from the ridge logistic regression models used in our classifier. Table 3 presents the signs of model coefficients averaged across 10 training sets for the 20 most informative features. As is immediately clear from the table, the choice of the software platform for Tweeting is the strongest predictor. Posting Tweets from the Web or using the Tweet Button is a strong indicator of a bot. On the other hand, predictably, Tweeting from mobile phones (with the surprising exception of Android tablets) as well as having Tweets geolocated is indicative of not being a bot. Besides choosing specific software platforms for Tweeting, bot creators also program them to Retweet more than an average human. Again, such behavior could be consistent with trying to impact rankings for news headlines.

Another peculiarity of bots is the content of their Tweets. Even a cursory look at the most common Tweets posted on days with the largest spikes reveals how bots are used for propaganda and counter-propaganda in Russia. The largest spike in bots' activity happened on February 28, 2015, the day following the killing of Boris Nemtsov, a prominent Russian opposition politician and former Deputy Prime Minister, in front of the Kremlin. The two most frequent Tweets bots posted that day were "Boris Nemtsov is killed in Moscow" (398 times) and "Opposition will organize a funeral march on March 1" (290 times). Many other Tweets were either Retweets or other news related to the killing. One can find a similar pattern of bots Tweeting news

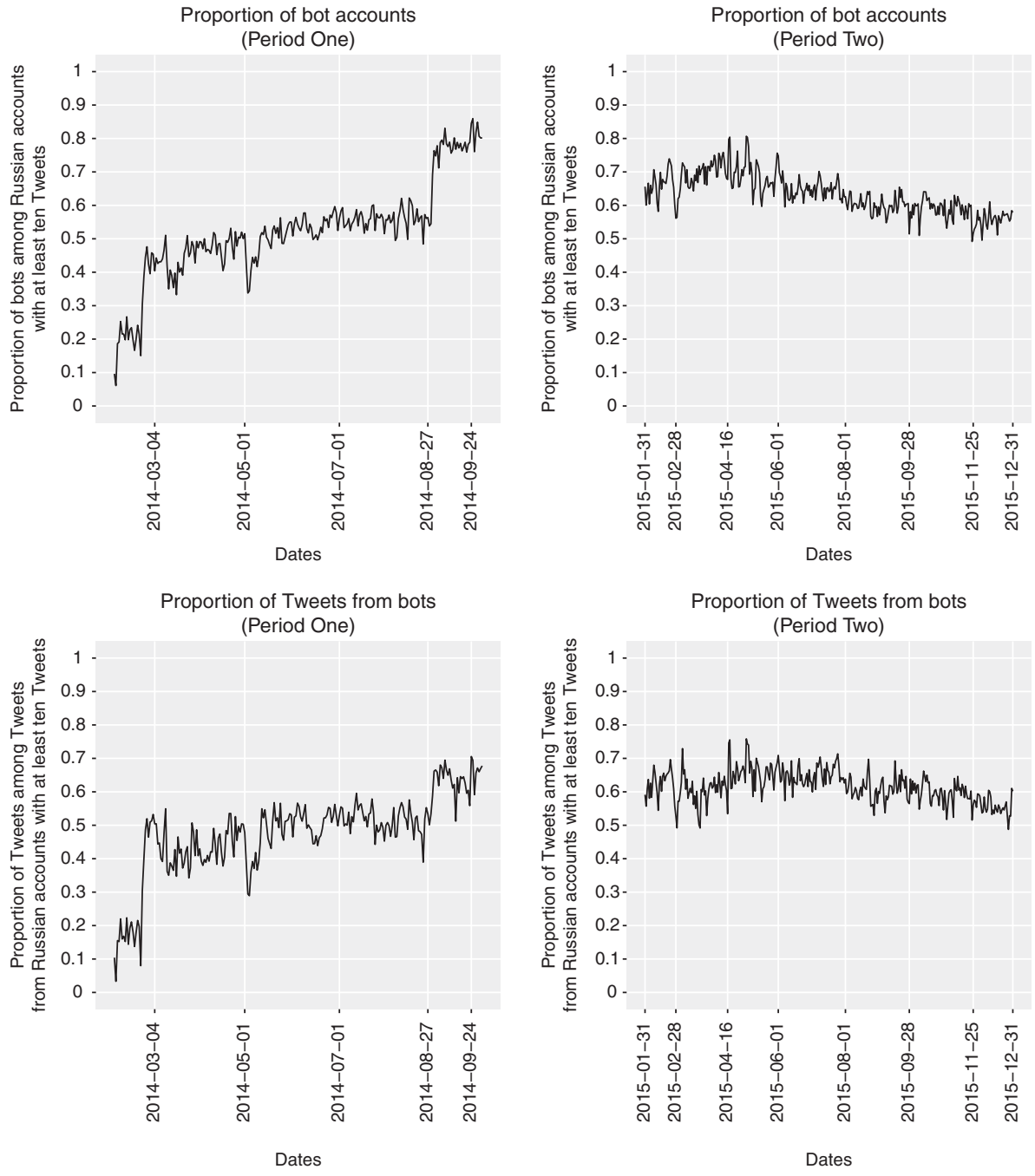


FIG. 1. Daily proportions of bot accounts and Tweets (majority rule). *Source:* Authors' calculations based on data collected from Twitter API.

headlines on other days with spikes. For example, on November 25, 2015, the day after a Turkish fighter jet shot down a Russian SU-24, both real and fake news about the incident were Tweeted and Retweeted by bots: "Reuters: United States believe that SU-24 was shot down in the air over Syrian territory" and "RT @lifenews_ru: At the G20 summit, Obama approved Erdogan's plan to shoot down a Russian SU-24." Another popular Tweet that day was the news that V. Putin and D. Medvedev opened the Boris Yeltsin Center in Russia.

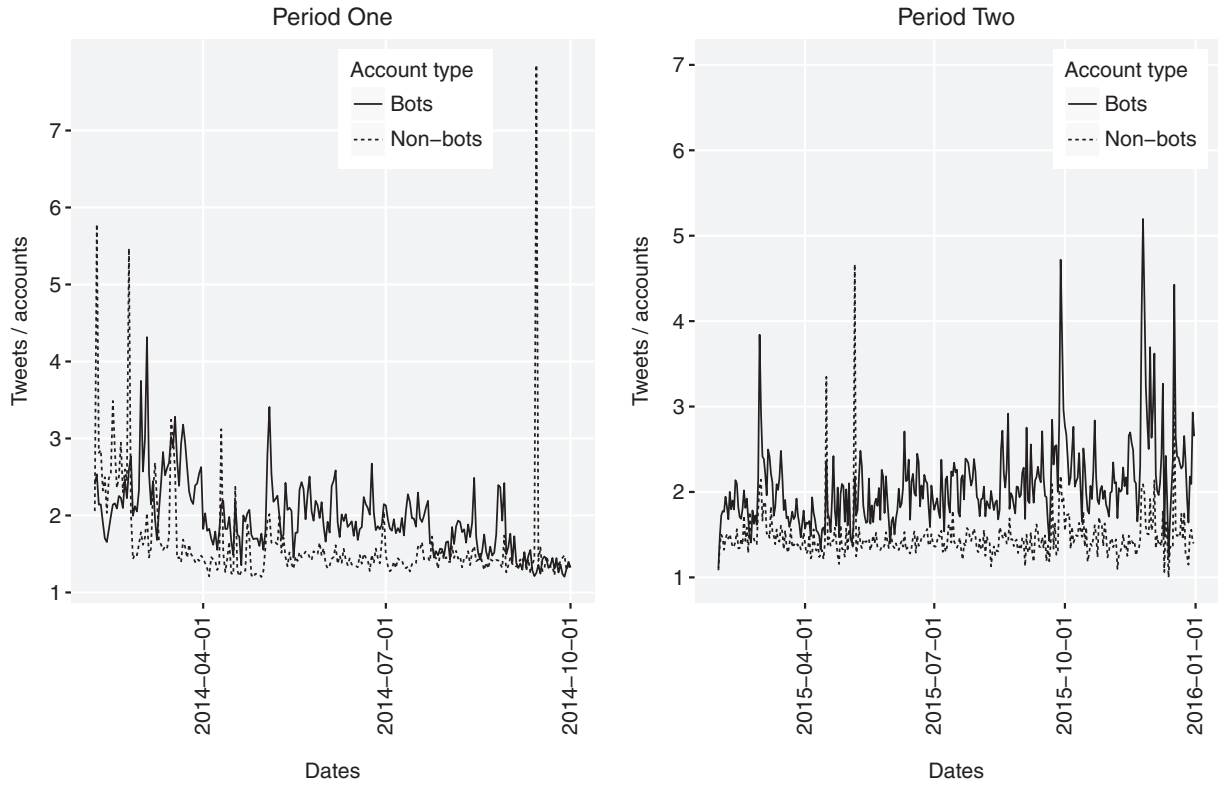


FIG. 2. Daily Tweets-to-accounts ratio for bots and non-bots. *Source:* Authors' calculations based on data collected from Twitter API.

Extension: Temporary Bots?

In this section, we consider one possible complication to our approach to identifying bots, which is the possibility that accounts could function as bots in one time period, but then be controlled by a human in another time period. Twitter has tools for detecting and blocking accounts that exhibit an unnatural behavior. Thus, bot managers should either constantly create new bots to replace old or banned accounts, and/or use tools that can trick Twitter algorithms. One such trick is apparently the use of intermittent human Tweets that distort the patterns of bot activity and make a bot account more human like. Another approach could involve provisionally renting (or borrowing) a human account for posting automatically generated Tweets. This latter type of bot accounts would arguably be among the hardest to detect, since their classification depends on the time of data collection. Indeed, if an account was changing hands between human and machine control, then our classification of the account would depend heavily on the period of time during which we collect our data.

Although a full-fledged investigation of the extent to which the latter approach is used and how exactly it might be implemented is beyond the scope of this article, we present in this study, an exploratory analysis based on comparing classification results for the accounts that were active both in periods 1 (February–October 2014) and 2 (February–December 2015) of our data collection. To do so, we take the classifier from the previous section—the one trained on the full data set—and then we run it separately on the data from periods 1 and 2. That is, we computed all features as if we only had the data for period 1 and applied our classifier to unlabeled accounts in period 1, and then repeat the process for period 2. The assumption in this study is that the classifier is correctly measuring the state of the world, so that if the account switches from being a bot to not being a bot across periods, we are assuming this means that it has actually changed from being controlled by an algorithm in one period to being controlled by a human in the other. Of course, we know that some of these changes could be

Table 3. Top 20 features (ridge logistic regression)

Feature	*Coefficient Sign*[a]	*Feature Importance*[b]
Platform: Twitter for Websites	+	1.3
Platform: Tweet Button	+	5.0
Platform: Twitter for iPhone	−	5.9
Platform: dlvr.it	+	7.2
Platform: Twitter for Android Tablets	+	7.2
% of Retweets	+	8.9
% of Tweets with a hashtag	−	9.0
Platform: Twitter for Android	−	9.9
Platform: Twitter for iPad	−	10.3
Platform: Mobile Web (M2)	−	10.8
Geo-enabled Tweets (binary)	−	11.0
Platform: Mobile Web (M5)	−	11.3
Platform: Twitter website	−	13.4
Politicalness[c]	−	14.0
Platform: Twitterfeed	+	14.7
Change of default profile image (binary)	+	15.0
% of Tweets at somebody	−	16.2
Platform: ifttt.com	−	17.8
Platform: Twitter Web Client	−	18.6
Entropy of platform use	−	18.7

The DV is the probability of being a bot. Positive coefficients mean that *ceteris paribus* bots tend to have higher values of a given feature than non-bots. All nonbinary features that are not proportions were standardized to make coefficients comparable. Features ranked by coefficients absolute value. All ranks averaged over 10 training sets. Signs refer to signs of averaged coefficients. Explained variance (deviance) ranges from 0.62 to 0.72 across models, and its average is 0.66.

Source: Authors' calculations based on data collected from Twitter API.

[a]Entries are signs of ridge logistic regression coefficients averaged over 10 training sets.

[b]Entries are average ranks of features across 10 training sets. The most informative feature has rank 1.

[c]The number of Tweets from an account in our collection over the estimated total number of Tweets that account posted during data collection.

due to the inherent error in the classifiers. Table 4 shows the results.

As can be observed from Table 4, out of almost 93,000 accounts that were active in both periods (which we define as at least 3 Tweets in each period, in addition to having at least 10 Tweets in total), over 62,600 (67%) were labeled as bots in at least one period. Out of these identified bots, 75% were categorized as bots in both periods. Only as few as 145 accounts that appeared as bots in one period were labeled as non-bots in the other. The remaining accounts can be labeled as unclear, since they are assigned to different categories on different training sets. Taken together, this is not evidence that would be consistent with widespread switching of the control of accounts on Twitter between bots and humans.

Table 4. Account type switches

	Period 1			
	Bots	*Unclear*	*Non-bots*	*Totals*
Period 2				
Bots	46,977	7241	83	54,301
Unclear	8284	24,730	2122	35,136
Non-bots	62	2052	1298	3412
Totals	55,323	34,023	3503	92,849

This table presents the evidence of accounts switching between operating as a bot and a non-bot. The ensemble was trained on joint data from both periods. Predictions made for periods 1 and 2 separately. The number in the top left cell is the number of accounts predicted to be bots both in periods 1 and 2 by the classifier trained on joint data. Bots and non-bots are labels predicted by the ensemble classifier using majority rule to aggregate predictions from 10 training sets. Unclear refers to accounts that did not get either the bot or non-bot label.

Source: Authors' calculations based on data collected from Twitter API.

Conclusion

The growing penetration of social media into different aspects of human life, including politics, creates new opportunities for political participation and learning, ensuring democratic accountability, and an open exchange of ideas. However, it also creates new avenues for propaganda, especially in nondemocratic settings. This article explores the case of the Russian political Twittersphere to study the political use of bots—a new and growing tool employed to sway public opinion and help or hamper political mobilization.

We develop methodology for detecting bots on Twitter that allows us not only to identify bots among currently active accounts but also to conduct a retrospective analysis, uncovering the dynamics of the use of bots over time. We propose an ensemble of classifiers based on a unanimous voting rule that allows us to detect bots with almost perfect precision (99%) and a sufficiently high recall (over 75%). This method provides a conservative estimate of the spread of bots among all Russian accounts that Tweeted at least 10 times on politically related themes (or, more precisely, had Tweets that contained politically related keywords) in 2014–2015 and shows that the daily proportion of bots among actively Tweeting Russian accounts in our collection reached as high as 85% during that time. This work reveals a very high presence of bots in the active Russian political Twittersphere, significantly higher that Twitter's estimate of the presence of bots on Twitter overall.

Contrary to the received wisdom,[25] we show that bots do not necessarily Tweet more than humans (although some bot accounts do), even though they are automated programs that could potentially be able to produce vast numbers of Tweets. We conjecture that bots are limited in their Tweeting activity both by Twitter bot detection algorithms and their creators' desire to avoid, as much as possible, revealing that the Tweets are not actually produced by human beings.

The comparative analysis of human and bot Twitter accounts shows that the best predictor of bot activity is

the software platform used for Tweeting. Bots are much more likely to use online platforms, whereas humans often use mobile devices. However, humans and bots are not dramatically different from each other on a number of other features that characterize their Tweeting activity and "habit." This reveals a relatively high level of bots' sophistication, whose account metadata and Tweeting patterns can be similar to those of humans, thus making it harder to identify bots.

A key substantive finding that requires further research relates to the use of bots to spread information. We find that the most common type of bot in our hand-coded data is one that Tweets news headlines without links to the original source of the news. This suggests that an important strategy in the use of bots for the purposes of propaganda might be to promote specific news stories and news media in the rankings of search engines. We also find that although many bots spread proregime information, there may also be antiregime bots that either disseminate information about opposition activities or criticize and deride the regime. Inferring the political position of detected bots, however, presents its own set of technical and machine learning challenges. In future research, we plan to develop sentiment analysis tools to link to not only identify bots in political Twitter but also to classify their political orientation.

Another issue that requires further investigation is classifiers' lifespan. Although we show in Table 4 that accounts do not often change their type by switching between being a bot account and a human account, further research is required to understand how often one needs to retrain bot-detecting classifiers. Table 5 presents one preliminary assessment by comparing predictions for period 2 from classifiers trained on both periods (i.e., the classifier presented previously in this article) and on a new classifier trained using data from period 1 only; as in Table 4, we again need to restrict our results to accounts active in both periods to conduct the analysis.

Table 5. Classifier robustness: prediction for period 2

	Trained on both periods			
	Bots	***Unclear***	***Non-bots***	***Totals***
Trained on period 1				
Bots	45,230	4589	0	49,819
Unclear	9071	29,123	1343	39,537
Non-bots	0	1424	2069	3493
Totals	54,301	35,136	3412	92,849

This table presents evidence of the classifier robustness to changes in the time span of the analysis: Would the classifier produce similar results if trained on data from a shorter period of time? We compare predictions for period 2 from the classifier trained on the whole collection versus trained on data from period 1 only. The top left number is the number of accounts predicted to be bots in period 2 by both classifiers. Bots and non-bots are labels predicted by the ensemble classifier using majority rule to aggregate predictions from 10 training sets. Unclear refers to accounts that did not get either the bot or non-bot label.

Source: Authors' calculations based on data collected from Twitter API.

So one way to interpret this table is by answering the question of how well would we have done predicting which accounts were bots in period 2 if we had simply reused a classifier we had previously trained using only data from period 1.[†] As the table illustrates, over 80% of accounts get identical labels from both classifiers, suggesting that there would certainly be some drop-off in accuracy from using a 2014 trained classifier to find bots in Russian political Twitter in 2015, but perhaps not as large as we might have feared. Nevertheless, further research should consider this question carefully, especially as we consider using classifiers with increasingly longer periods of time between the training set and the unlabeled data, to say nothing of detecting bots using a classifier trained in one country to label accounts in another country. To reiterate, the features in our classifier were chosen so as not to make them Russia specific, but any such application of our classifier outside of Russia should be carefully verified. Another aspect of the proposed methodology that requires further exploration is the robustness of the classifier: would it produce the same results if trained on a data set with more Tweets from the same accounts? Recall that our classifier was trained only on the "political" Tweets that entered our collection from using keyword searches. While this approach has many positive features in terms of replication, as we have discussed previously, it is possible that going back and collecting all of these accounts', other Tweets would result in a more accurate classifier. Finally, the proposed bot detection methodology can only be as accurate as the training set used for supervised learning; further research needs to be conducted to improve our understanding of how good and consistent human coders are in detecting different types of bots.

While much work remains to be done and many more questions to be answered, the methods developed in this article represent an important first step in the ability to detect the presence and activity of bots in the political Twittersphere. Also, as recent developments have aptly shown, this is a task that stands to

[†]Note that the reference category here is the classifier trained on data from both periods, which we are again assuming produces the "correct" state of the world for this exercise.

become only all the more urgent in the coming years, and especially so in the case of Russian bots.[‡]

Acknowledgments

We are extremely grateful to Pablo Barberá, Neal Beck, Rita Kamalova, Sean Kates, Megan Metzger, Jonathan Nagler, Jennifer Pan, Duncan Penfold-Brown, Margaret Roberts, and Anastasiia Shukhova for their feedback and valuable suggestions. The data were collected by the New York University Social Media and Political Participation (SMaPP) laboratory (https://wp.nyu.edu/smapp/), of which Bonneau and Tucker are codirectors along with John T. Jost and Jonathan Nagler. The SMaPP laboratory is supported by the INSPIRE program of the National Science Foundation (Award SES-1248077), the New York University Global Institute for Advanced Study, the Moore-Sloan Data Science Environment, Dean Thomas Carew's Research Investment Fund at New York University, and the John S. and James L. Knight Foundation. We are also grateful to our volunteer coders from the Higher School of Economics (Moscow, Russia): Ivan Aleksandrov, Valeria Babayan, Maret Bochaeva, Daria Bushina, Viktoria Dimova, Yulia Gavrilova, Anastasia Gergel, Tatyana Glushkova, Alexandra Goncharova, Egor Ilin, Christina Ilina, Aleksandra Izyumova, Artem Kolganov, Nikita Konyukhov, Yulia Korneeva, Maria Kuz, Alena Kuznetsova, Nikita Lata, Alina Lyutikova, Maria Makarova, Kamila Malikova, Elena Malysheva, Polina Malyutina, Ekaterina Mikhaylova, Dmitry Muravyov, Mikhail Murzin, Pavel Myslovskiy, Timur Naushirvanov, Maxim Novokreschenov, Veronika Pankina, Anastasia Parshina, Dmitrii Prodanov, Anastasia Rodygina, Zlata Sergeeva, Rais Shaidullin, Maria Sidorova, Viktor Sinitsyn, Anna Skosyreva, Timur Slavgorodskiy-Kazanets, Anna Sokol, Elizaveta Sokovnina, Georgy Tarasenko, Oksana Tiulpinova, Azizbek Tulaganov, Natalia Vasilenok, Anna Velikanova, Alena Volodkina, Alexey Volokhovich, Anna Zaychik, and Kirill Ziborov.

Authors' Contributions

Stukal, Sanovich, Bonneau, and Tucker designed the study. Stukal developed the code for all the analyses and prepared the first draft of the article. Stukal recruited and Sanovich trained human coders. Both Sanovich and Stukal supervised human coders. Bonneau and Tucker oversaw the data collection process. All the authors participated in revising and editing of the article.

Author Disclosure Statement

No competing financial interests exist.

References

1. Howard P, Hussain M. Democracy's fourth wave? Digital media and the Arab Spring. Oxford, UK: Oxford University Press, 2013.
2. Barber P. How social media reduces mass political polarization. Evidence from Germany, Spain, and the U.S. In: Presented at the Annual Meeting of the American Political Science Association, San Francisco, CA, 2015.
3. Thompson A. Journalists and Trump voters live in separate online bubbles, MIT analysis shows. Vice News, December 2016.
4. Kollanyi B, Howard P, Woolley S. Bots and automation over Twitter during the U.S. election. COMPROP Data Memo, November 2016.
5. Bessi A, Ferrara E. Social bots distort the 2016 U.S. Presidential election online discussion. First Monday. 2016;21. Available online at: http://firstmonday.org/article/view/7090/5653
6. Oentaryo R, Murdopo A, Prasetyo P, Lim E-P. On profiling bots in social media. Soc Informat. 2016;10046:92–109.
7. Ford H, Dubois E, Puschmann C. Keeping Ottawa honest one Tweet at a time? Politicians, Journalists, Wikipedians and their Twitter bots. Int J Commun. 2016;10:4891–4914.
8. Sanovich S, Stukal D, Tucker J. Turning the virtual tables: Government strategies for addressing online opposition with an application to Russia. Comparative Politics 2018.
9. Woolley S. Automating power: Social bot interference in global politics. First Monday. 2016;21. Available online at: http://firstmonday.org/article/view/6161/5300
10. Guriev S, Treisman D. How modern dictators survive: Cooptation, censorship, propaganda, and repression. 2015.
11. Munger K, Bonneau R, Jost J, et al. Elites Tweet to get feet off the streets: Measuring regime social media strategies during protest. 2017.
12. Sanovich S. Computational propaganda in Russia: The origins of digital misinformation. 2017.
13. Lasswell H, Lerner D, Hans S. Propaganda and communication in world history: Volume I. The symbolic instrument in early times. Honolulu, HI: The University Press of Hawaii, 1979.
14. Lasswell H, Lerner D, Hans S. Propaganda and communication in world history: Volume II. Emergence of public opinion in the west. Honolulu, HI: The University Press of Hawaii, 1980.
15. Kallis A. Nazi propaganda and the Second World War. London: Palgrave Macmillan, 2005.
16. Berkhoff K. Motherland in danger: Soviet propaganda during World War II. Cambridge: Harvard University Press, 2012.
17. Welch D. Nazi propaganda: The power and the limitations. London: Routledge, 2014.
18. Chen A. The agency. The New York Times, June 2, 2015.
19. Snegovaya M. Putin's Information Warfare in Ukraine: Soviet Origins of Russia's Hybrid Warfare. Washington, DC: Institute for the Study of War, 2015.
20. Pomerantsev P. Inside the Kremlin's hall of mirrors. The Guardian, April 9, 2015.
21. Applebaum A, Lucas E. The danger of Russian disinformation. The Washington Post, May 6, 2016.
22. Ananyev M, Sobolev A. Fantastic beasts and whether they matter: Do Internet "Trolls" influence political conversations in Russia? In: Presented at Midwest Political Science Association, Chicago, IL, April 2017.
23. Remnick D, Yaffa J, Osnos E. Trump, Putin, and the New Cold War. The New Yorker (Annals of Diplomacy), March 2017.
24. Davis C, Varol O, Ferrara E, et al. BotOrNot: A system to evaluate social bots. In: Proceedings of the 25th International Conference on World Wide Web Companion, Montréal, Québec, Canada, April 11–15, 2016. pp. 273–274.
25. Nimmo B. #BotSpot: Twelve Ways to Spot a Bot. DFRLab. 2017. Available online at: https://medium.com/dfrlab/botspot-twelve-ways-to-spot-a-bot-aedc7d9c110c

[‡]See as just a couple of many recent examples: www.nytimes.com/2017/09/27/technology/twitter-russia-election.html and https://www.nytimes.com/2017/09/28/us/politics/twitter-russia-interference-2016-election-investigation.html

26. Zeller T, Jr. Gaming the search engine, in a political season. The New York Times, November 6, 2006.
27. Dickerson J, Kagan V, Subrahmanian VS. Using sentiment to detect bots on Twitter: Are humans more opinionated than bots? In: Proceedings of the 2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), Beijing, China, August 17–20, 2014. pp. 620–627.
28. Cresci S, Pietro RD, Petrocchi M, et al. 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, Perth, Australia, April 3–7, 2017. pp. 963–972.
29. Subrahmanian VS, Azaria A, Durst S, et al. The DARPA Twitter bot challenge. Computer. 2016;49:38–46.
30. Cresci S, Pietro RD, Petrocchi M, et al. DNA-inspired online behavioral modeling and its application to spambot detection. IEEE Intell Syst. 2016;31:58–64.
31. Chavoshi N, Hamooni H, Mueen A. Identifying correlated bots in Twitter. Soc Informat. 2016;10047:14–21.
32. Ratkiewicz J, Conover M, Meiss M, et al. Detecting and tracking political abuse in social media. In: Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media (CWSM), July 17–21, 2011, Barcelona, Spain, pp. 297–304.
33. Chu Z, Gianvecchio S, Wang H, Jajodia S. Detecting automation of Twitter accounts: Are you a human, bot, or cyborg? IEEE Trans Depend Sec Comput. 2012;9:811–824.
34. Warwick K, Shah H. Human misidentification in Turing tests. J Exp Theor Artif Intell. 2015;27:123–135.
35. Choi S-S, Cha S-H, Tappert C. A survey of binary similarity and distance measures. J Syst Cybern Inform. 2010;8:43–48.
36. Weiss G. Mining with rarity: A unifying framework. ACM SIGKDD Explor Newslett. 2004;6:7–19.
37. Weiss G, Provost F. Learning when training data are costly: The effect of class distribution on tree induction. J Artif Intell Res. 2003;19: 315–354.
38. Sun Y, Wong A, Kamel M. Classification of imbalanced data: A review. Int J Pattern Recogn Artif Intell. 2009;23:687–719.
39. Kuncheva L. A theoretical study on six classifier fusion strategies. IEEE Trans Pattern Anal Mach Intell. 2002;24:281–286.
40. van Erp M, Vuurpijl L, Schomaker L. An overview and comparison of voting methods for pattern recognition. In: Proceedings of the Eighth International Workshop on Frontiers in Handwriting Recognition (IWFHR), Ontario, Canada, August 6–8, 2002. pp. 195–200.
41. Lin X, Yacoub S, Burns J, Simske S. Performance analysis of pattern classifier combination by plurality voting. Pattern Recogn Lett. 2003;24: 1959–1969.
42. Lam L, Suen C. Application of majority voting to pattern recognition: An analysis of its behavior and performance. IEEE Trans Syst Man Cybernet A. 1997;27:553.
43. Kuncheva L, Rodrguez J. A weighted voting framework for classifiers ensembles. Knowl Inf Syst. 2014;38:259–275.
44. Zhu J, Hastie T, Rosset S, Zou H. Multi-class AdaBoost. Stat Interface. 2009; 2:349–360.

Cite this article as: Stukal D, Sanovich S, Bonneau R, Tucker JA (2017) Detecting bots on Russian political Twitter. *Big Data* 5:4, 310–324, DOI: 10.1089/big.2017.0038.

Abbreviations Used

RBF = radial basis function
SAMME = Stagewise Additive Modeling using Multiclass exponential loss function
SMaPP = Social Media and Political Participation
SVM = support vector machine

Appendix A. Keywords and Hashtags for Collecting Twitter Data

This list uses transliteration. A Cyrillic equivalent was used when "Cyril." is given in parentheses.

- medvedev (Cyril.)
- dukhovniyeskrepy (Cyril.)
- putinvor (Cyril.)
- putinakh (Cyril.)
- pzhiv (Cyril.)
- Strategiya31 (Cyril.)
- triumfalnaya (Cyril.)
- bolotnaya (Cyril.)
- oppozicia (Cyril.)
- gorozhaneprotiv (Cyril.)
- surkovskayapropaganda (Cyril.)
- navalniy (Cyril.)
- zanavalnogo (Cyril.)
- komandanavalnogo (Cyril.)
- suvkirove (Cyril.)
- PussyRiot
- PussyRiot (Cyril.)
- tolokonnikova (Cyril.)
- narodniyskhod (Cyril.)
- sdnempobedi (Cyril.)
- #sochi2014
- #sochi
- #putinsgames
- #sochi (Cyril.)
- #vitishko (Cyril.)
- #schitaemvmeste (Cyril.)
- #sochifail
- #sochi2014problems
- golodovka (Cyril.)
- MinutaNeMolchaniya (Cyril.)
- zhalkiy (Cyril.)
- puti (Cyril.)
- spasiboputinuzaeto (Cyril.)
- priamayaliniya (Cyril.)
- partiyazhulikovivorov (Cyril.)
- edro (Cyril.)

- 6maya (Cyril.)
- sobyaninnashmer (Cyril.)
- marshmillionov (Cyril.)
- zachestniyevibory (Cyril.)
- bolotnoyedelo (Cyril.)
- 6may
- svobodupolitzaklyuchennym (Cyril.)
- svoboduuznikam6maya (Cyril.)
- rosuznik (Cyril.)
- odinzavsekh (Cyril.)
- vsezaodnogo (Cyril.)
- rasserzhennye (Cyril.)
- chestniyevybory (Cyril.)
- udaltsov (Cyril.)
- vysurkovskayapropaganda (Cyril.)
- 37godvernulsya (Cyril.)
- DMP (Cyril.)
- privet37god (Cyril.)
- krovaviyrezhim (Cyril.)
- kirovles (Cyril.)
- tolokno (Cyril.)
- biryulevo (Cyril.)
- khvatitkormitkavkaz (Cyril.)
- khvatitvinitkavkaz (Cyril.)
- russkiymarsh (Cyril.)
- Sochi2014 (Cyril.)
- #olimpiada (Cyril.)
- #olimpiyskayazachistka (Cyril.)
- #sochiproblems
- Odessa (Cyril.)
- #Nemtsov
- #Nemtsov (Cyril.)
- nemtsov (Cyril.)
- savchenko (Cyril.)
- #FreeSavchenko
- #Putinkiller
- maidan (Cyril.)
- maidaner (Cyril.)
- maidanutiy (Cyril.)
- #PutinUmer (Cyril.)
- #MinutaNeMolchaniya (Cyril.)
- Su24
- Su-24 (Cyril.)
- Su24 (Cyril.)
- #samolet (Cyril.)
- #RussianJet
- #ExpelTurkeyFromNATO
- #Russianplane
- #Erdogan
- #Latakia

Appendix B. Features for Classification

Metadata used for classification includes 15 features:

1. default profile image (binary feature)
2. default background image (binary feature)
3. change in the default profile image over time (binary feature)
4. change in the default background image over time (binary feature)
5. digits other than dates in screen name (binary feature)
6. more than one word in name (binary feature)
7. the number of characters in user description
8. no user description (binary feature)
9. change in user description over time (binary feature)
10. the average followers-to-friends ratio (set to zero if an account has no friends)
11. no friends (binary feature)
12. user location specified in user profile (binary feature)
13. the maximum number of Tweets the user favored over the total number of her Tweets
14. account has geotagged Tweets (binary feature)
15. change in the binary feature for the account's geotagged Tweets over time (binary feature)

The 27 Tweeting characteristics used for classification include the following:

1. average number of hashtags per Tweet
2. standard deviation of the number of hashtags per Tweet
3. percentage of Tweets with hashtags
4. percentage of Retweets
5. entropy of inter-Tweets time intervals in seconds
6. entropy of the software platform used for Tweeting
7. number of Tweets in our collection over the estimated total number of Tweets the account posted during our data collection
8. number of different Retweeted accounts divided by the total number of Retweets
9. percentage of directed Tweets with the "@" symbol
10. percentage of Tweets with an URL
11. average number of URLs per Tweet
12. maximum number of URLs per Tweet
13. standard deviation of number of URLs per Tweet
14. total number of Tweets over the number of days the account existed in our collection (computed as the difference in days between the last and the first Tweet from that account in the collection)

15. Twitter website used for Tweeting at least once
16. Twitter for Websites used for Tweeting at least once
17. Twitter Web Client used for Tweeting at least once
18. Mobile Web (M2) used for Tweeting at least once
19. Mobile Web (M5) used for Tweeting at least once
20. Tweet Button used for Tweeting at least once
21. Twitter for iPhone used for Tweeting at least once
22. Twitter for Android used for Tweeting at least once
23. Twitter for Android Tablets used for Tweeting at least once
24. Twitter for iPad used for Tweeting at least once
25. http://dlvr.it used for Tweeting at least once
26. http://twitterfeed.com used for Tweeting at least once
27. http://ifttt.com used for Tweeting at least once

Appendix C. Additional Figures and Robustness Checks

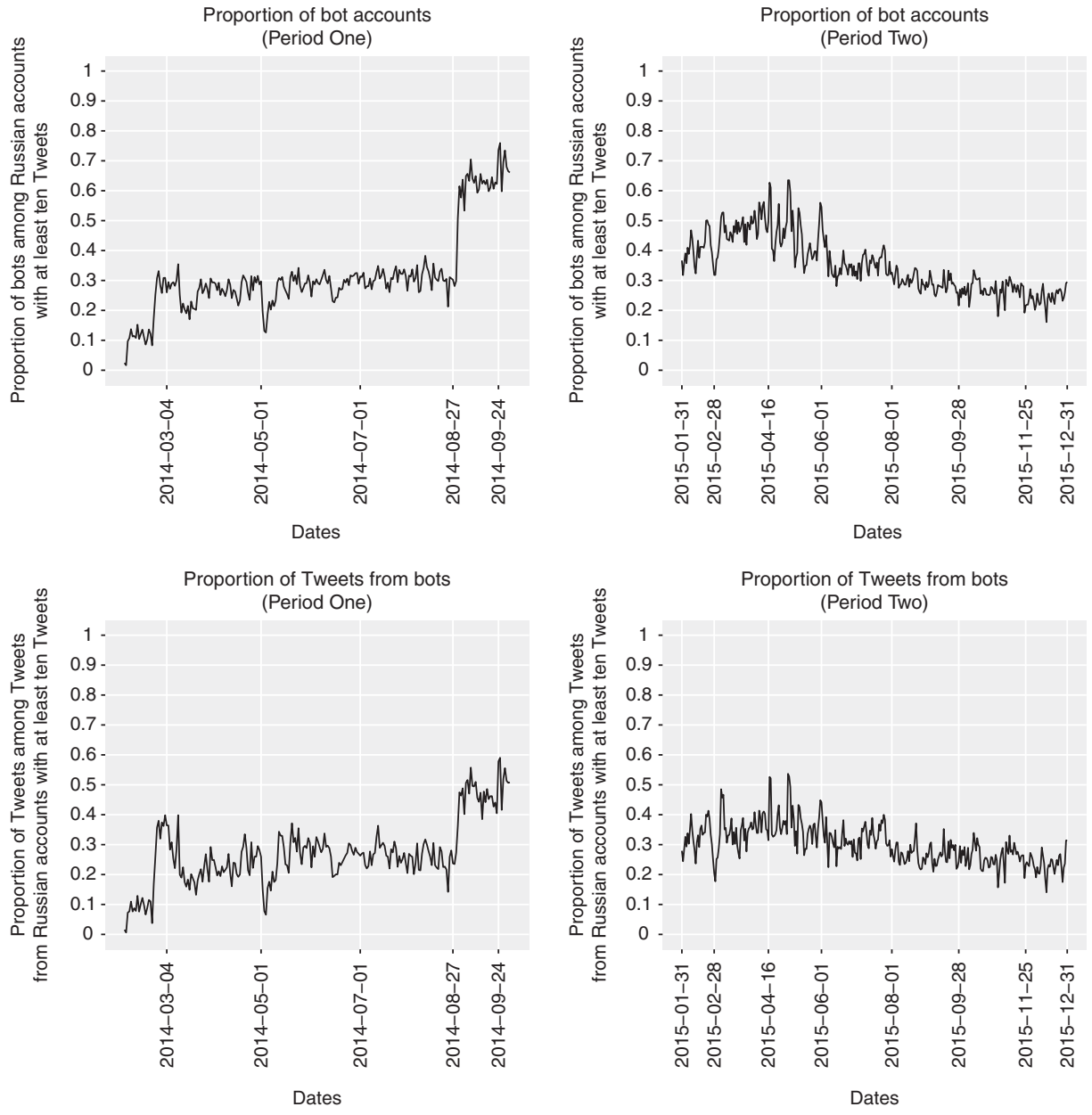
Appendix Table C1. Model parameters across 10 training sets

Parameter	*SVM (RBF kernel)*	*Logistic regression (ridge)*	*SAMME*	*XGBoost*
C	2.8 (1.44)			
σ	0.9 (1.26)			
α				1.6 (0.96)
λ		0.03 (0.004)		2.2 (0.78)
M			96 (45.51)	
Min			8.5 (2.42)	
η				0.76 (0.16)
Depth				20.4 (2.84)
Nround				15 (4.08)

Note: Main entries are average model parameters. Standard errors across 10 training sets are in parentheses. SAMME stands for Stagewise Additive Modeling using Multiclass exponential loss function.[44] All computations made in R (v.3.3.1) on x86_64-centos-linux-gnu. Package versions: SVM (kernlab, v.0.9-24), ridge logistic regression (glmnet, v.2.0-5), SAMME (adabag v.4.1), XGBoost (xgboost, v.0.4-4).

Source: Authors' calculations based on data collected from Twitter API.

α, elastic net mixing parameter; η, learning rate (scaling parameter for the contribution of each tree); λ, penalty term; σ, inverse kernel width; C, cost of constraints violation; Depth, maximum tree depth; M, number of boosting iterations; Min, minimum number of obs in a node; Nround, passes on the data; RBF, radial basis function; SVM, support vector machine.



APPENDIX FIG. C1. Dynamics of bot activity (unanimous rule). *Source:* Authors' calculations based on data collected from Twitter API.