Simplistic Collection and Labeling Practices Limit the Utility of Benchmark Datasets for Twitter Bot Detection

Chris Hays*
Massachusetts Institute of Technology
jhays@mit.edu

Zachary Schutzman*
Massachusetts Institute of Technology
zis@mit.edu

Manish Raghavan
Massachusetts Institute of Technology
mragh@mit.edu

Erin Walk
Massachusetts Institute of Technology
ewalk@mit.edu

Philipp Zimmer
Massachusetts Institute of Technology
philippz@mit.edu

ABSTRACT

Accurate bot detection is necessary for the safety and integrity of online platforms. It is also crucial for research on the influence of bots in elections, the spread of misinformation, and financial market manipulation. Platforms deploy infrastructure to flag or remove automated accounts, but their tools and data are not publicly available. Thus, the public must rely on third-party bot detection. These tools employ machine learning and often achieve near-perfect performance for classification on existing datasets, suggesting bot detection is accurate, reliable and fit for use in downstream applications. We provide evidence that this is not the case and show that high performance is attributable to limitations in dataset collection and labeling rather than sophistication of the tools. Specifically, we show that simple decision rules — shallow decision trees trained on a small number of features — achieve near-state-of-the-art performance on most available datasets and that bot detection datasets, even when combined together, do not generalize well to out-of-sample datasets. Our findings reveal that predictions are highly dependent on each dataset's collection and labeling procedures rather than fundamental differences between bots and humans. These results have important implications for both transparency in sampling and labeling procedures and potential biases in research using existing bot detection tools for pre-processing.

CCS CONCEPTS

• **Information Systems** → *Web applications*; • **Social and professional topics** → *User characteristics*; *Computing / technology policy*; • **Computing methodologies** → **Machine learning**.

*Both authors contributed equally to this research.

Code for the results presented is available on GitHub at github.com/johnchrishays/bot-detection and can be accessed with DOI doi.org/10.5281/zenodo.7196519. The authors would like to thank Sinan Aral, Dean Eckles, and Kiran Garimella for helpful research discussions, as well as the anonymous reviewers for helping improve the presentation of the results. E.W. is supported by the NSF Graduate Research Fellowship Program.

WWW '23, April 30–May 04, 2023, Austin, TX, USA

ACM ISBN 978-1-4503-9416-1/23/04.
https://doi.org/10.1145/3543507.3583214

KEYWORDS

Social media, bot detection

ACM Reference Format:
Chris Hays, Zachary Schutzman, Manish Raghavan, Erin Walk, and Philipp Zimmer. 2023. Simplistic Collection and Labeling Practices Limit the Utility of Benchmark Datasets for Twitter Bot Detection. In *Proceedings of the ACM Web Conference 2023 (WWW '23), April 30–May 04, 2023, Austin, TX, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3543507.3583214

1 INTRODUCTION

With the rise of online social media as an important means for connecting with others and sharing information, the influence of *bots*, or automated accounts, has become a topic of vital societal concern. Some bots are benign and serve content which is entertaining or directly enhances the accessibility of the site (e.g., by providing captions for otherwise uncaptioned videos on a platform), but many others engage in influence operations, the spread of misinformation, and harassment: fake followers boost some users' perceived popularity; spammers flood the site with advertisements for a political candidate or product; malicious automated accounts undermine the credibility of elections or inflame polarization. Bots have reportedly influenced the 2016 US Presidential Election [4, 36], the Brexit vote in the UK [3, 36], the spread of misinformation about COVID-19 [25] and financial markets [11, 52]. The ability (or inability) to accurately label such accounts could have a very real impact on elections and public health as well as public trust in institutions.

Platforms remove large numbers of accounts that they deem inauthentic, but they keep these removal systems secret and may be incentivized to misrepresent the influence or prevalence of bots. Indeed, bot detection was at the center of Elon Musk's negotiations to buy Twitter: Twitter claimed that less than five percent of its monetizable users are bots [66] while Musk claimed the number is much higher [51]. Because internal bot detection techniques are in general not made public, researchers, journalists, and the public at-large rely on researcher-developed tools to separate bots from genuine human users and understand the impact of bots on social phenomena.

Developing tools for bot detection on Twitter and other online social media platforms is an active area of research. Over the last decade, an abundance of user datasets have been collected for the purpose of enabling third-party bot detection. Tools trained on these datasets achieve high (sometimes nearly perfect) performance using expressive machine learning techniques such as ensembles of

random forests and deep neural networks, and hundreds or thousands of features such as profile metadata, engagement patterns, network characteristics, and tweet content and sentiment.

Crucially, resarchers frequently use bot detection as a *preprocessing* step to study social phenomena, to separate human users from bots and study phenomena related to one or both of humans and bots. This includes topic areas such as the spread of mis- or disinformation [6, 40, 53, 61–63, 67], elections [2, 4, 24, 41, 54, 64] and echo chambers [7] and published in premier venues for scientific research including *Science* [67], *Nature* [53] and *PNAS* [64]. For example, Broniatowski et al. [6] observed that bots eroded the population's trust in vaccinations, González-Bailón et al. [35] conclude bots share disproportionate amount of content during political protests and Vosoughi et al. [67] conclude that humans and bots spread fake news in different ways. The robustness and validity of these results depend on accurate and reliable bot detection.

Third-party bot detection tools are also easily accessible to and widely used by the public: the most recent version of Botometer [60] reportedly receives hundreds of thousands of daily queries to its public API [74] and BotSentinel [5] provides a browser extension and ways to conveniently block accounts classified as bots.

Is bot detection a solved problem? On its face, bot detection research seems to be a success story for machine learning: researchers have collected a variety of datasets for a well-defined classification task and expressive machine learning models like random forests and neural networks attain near-perfect performance on the data. Moreover, these methods have been widely adopted in both the academic literature and in public use. Bot detection tools are frequently trained on a combination of datasets, and researchers have argued that the existing approach can easily adapt to the short-comings of existing classifiers or evolution of more human-like bots by adding more datasets [60] or using even more complex techniques, like generative adversarial networks [9].

Even so, there are signs that bot detection tools are far from perfect. They may disagree with one another [47], prove unreliable over time [56], and rely on dubious labels [26, 27]. Here, we attempt to reconcile and systematically explain the apparent success of Twitter bot detection with what seem to be significant limitations.

Evaluating third-party bot detection datasets and tools is inherently challenging: the "ground truth" is unknown or inaccessible to the public, and the only window of insight we have into bots on Twitter is through the datasets themselves. However, this does not make evaluation impossible. We can still gain a better understanding of what these datasets tell us by closely analyzing the datasets and how they relate to one another.

Consider, for example, the dataset released by Cresci et al. [10] (`cresci-2017`), one of the most widely used in the academic literature. This dataset consists of a pool of genuine human users, collections of fake followers, and several types of 'spam bots': a diverse collection of accounts in this domain. The state-of-the-art model is a deep neural network using text data which achieves essentially perfect performance on this dataset [43]. However, a closer inspection revealed something surprising: we can achieve near-state-of-the-art performance using a classifier that asks a single yes/no question of the data. In fact, there are at least two different yes/no questions which nearly separate the human and bot classes. These classifiers are shown in the left and middle decision trees in Figure 1. As we discuss later, we believe the left tree is an artifact of convenience sampling from Cresci et al. [16], which concerns social sensing of natural disasters using Twitter. On the right of Figure 1 we show another high-performing classifier for another popular dataset: `caverlee-2011` published in [44]. Again, a small number of yes/no questions distinguishes humans from bots with high accuracy. These examples are not exceptional cases. As we will show, almost all of the other benchmark datasets we analyze admit high performance using very simple classifiers.

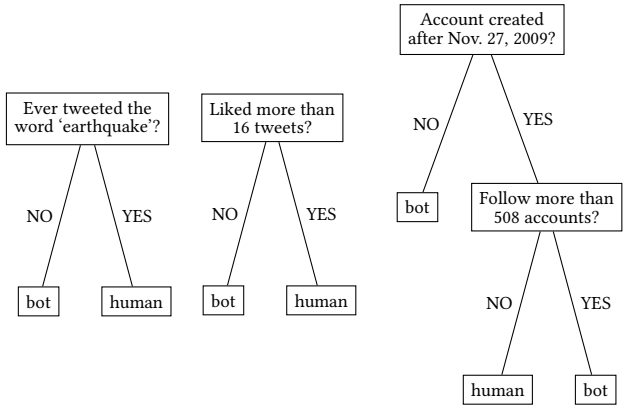

Figure 1: Two shallow decision trees for `cresci-2017` (left, middle) achieving accuracies of 0.98 and one for `caverlee-2011` (right) with an accuracy of 0.91.

How should we reconcile these results with our intuition that that bot detection is a difficult problem? On the one hand, it is possible that bot detection is inherently simpler than expected, and simple decision rules suffice. On the other, perhaps the datasets themselves fail to capture anywhere near the true complexity of bot detection. If this is the case, then while simple decision rules perform well in-sample, their performance will be significantly worse when deployed. We provide evidence to support the latter hypothesis across a wide range of Twitter bot detection datasets.

Our contributions. In this work, we carefully examine widely used datasets for Twitter bot detection and explore their limitations. First, we demonstrate that simple decision rules perform nearly as well as the state-of-the-art models on benchmark datasets. Thus, each dataset only provides predictive signals of limited complexity. Because our simple decision rules allow us to transparently inspect the reasons for our classifiers' high performance, we find that predictive signals in the datasets likely reflect particular collection and labeling procedures, i.e., the processes for collecting accounts from Twitter and assigning a human or bot label to each account.

Next, we examine combinations of datasets. Many bot detection tools combine datasets (see, e.g., [17, 37, 75]) and argue implicitly or explicitly that it is possible to cover the distribution of bots that appear on Twitter by doing so. Building on prior work [18, 60], we show that expressive machine learning models trained on one dataset do not perform well when tested on others and that models trained on all but one dataset perform poorly when evaluated on the held-out one. Information provided by a dataset does not generalize to others, suggesting that datasets are distributed according to

dissimilar distributions, which indicates different sampling (i.e., collection and labeling) procedures.

Finally, we consider whether imposing structural assumptions about the data, namely that each dataset contains bots from one of a small number of *types* (e.g., spam bots or fake followers) can yield greater generalization as the approaches in Sayyadiharikandeh et al. [60] and Dimitriadis et al. [17] suggest. We find that simple decision rules can accurately differentiate bots of each type from humans. Thus, each sample of bots of one type is itself of low informational complexity. We additionally show that, within accounts of a particular bot type, simple decision rules can identify from which dataset a given bot originates. Thus, datasets of a given bot type are drawn from very different distributions, again indicating different data collection procedures. Taken together, these results suggest that each individual dataset contains little information, predictive signals in each dataset are not informative for prediction on the others, and this is true even within datasets representing a particular type of bots. Therefore, existing datasets are unlikely to provide a representative or comprehensive sample of bots, and it is unlikely that classifiers trained on this data will perform well when deployed.

Beyond bot detection, our methodology — examining simple decision rules on datasets and measuring cross-dataset performance — may be useful for detecting simplistic data sampling and labeling processes in a range of machine learning applications: If datasets admit highly accurate simple decision rules, the datasets themselves have low informational complexity. If, additionally, expressive machine learning models trained on some datasets do not generalize to other datasets, the underlying system does not appear to be simple, and the datasets are unlikely to provide insight into the problem domain as a whole.

We also believe these findings have direct implications for future bot detection research both on Twitter and beyond: creators of bot detection datasets should transparently report and justify sampling and labeling procedures; researchers developing bot detection techniques should train and analyze simple, interpretable models alongside more expressive ones; and researchers using bot detection as a preprocessing step should consider how it may bias results.

2 BACKGROUND

Bot detection techniques. Researchers have used a range of cutting-edge machine learning techniques to bot detection across diverse types of data in order to improve classification. One approach applies random forests [32, 72] and ensembles of random forests that combine predictions from classifiers trained on subsets of data [17, 60]. Another popular approach leverages text data to apply large pre-trained language models [38] or models trained by the researchers themselves [28, 39, 43, 46, 48]. A third approach uses network data to train graph neural networks [1, 20, 23] or try to detect botnets from anomolous network structures [70]. Finally, a fourth approach seeks insight from other disciplines by using behavioral [30, 34] or biology-inspired techniques [13–15, 58]. In addition to novel predictive models, significant effort is spent deriving or exploring profile, text or network features that are likely to be informative for bot detection [39, 49]. All of the papers cited above rely on the benchmark datasets analyzed in our work.

Limitations of bot detection tools. Several papers have explored the limitations of bot detection techniques, but few provide evidence to explain these limitations. To the best of our knoweldge, our work is the first that traces the limitations of bot detection to simplistic sampling and labeling strategies. Martini et al. [47] compare three public tools for bot detection and finds significant disagreement of predictions across tools. Relatedly, Rauchfleisch and Kaiser [56] find that a single tool may produce varying results over time as a result of variation in account activity and Torusdağ et al. [65] created bots that can reliably evade existing bot detection frameworks. Elmas et al. [19] find that the qualitative observations of prior work, such as that bot accounts are typically recently created or are marked by a high volume of activity, do not hold on data collected for their paper and conclude that popular classifiers may not generalize. Gallwitz and Kreil [26, 27] manually identify individual accounts that are incorrectly labeled as 'bots' in popular datasets, noting a high prevalence of false positives and arguing that the labels which are typically taken as ground truth may have errors.

3 DATA & METHODS

In this section, we discuss the datasets we analyze and our criteria for including each in our analysis. Most benchmark datasets in the literature are aggregations of data collected across various contexts. The benchmark datasets we study are summarized in Table 1.

3.1 Dataset collection.

To collect a list of benchmark datasets, we searched Google Scholar for peer-reviewed papers related to bot detection and within the references of papers we found. We found a total of 58 papers using at least one of the datasets we included in our analysis, of which 22 had at least 50 citations on Google Scholar at the time of writing (while several had at least 500 citations) and 26 of which were published since 2020. In our analysis, we only include datasets that were used in multiple peer-reviewed bot detection papers reporting accuracy and F1 scores that we found in our search, although nearly all of the datasets were used in many more than two. Several of the datasets were accessed via the the Botometer Bot Repository.[1] For the rest of the datasets, we reached out to the author(s) of the associated paper to request access to the original data (for `twibot-2020` and `yang-2013`) or found public access to the data online (in the cases of `caverlee-2011` and `pan-2019`).

We also received augmented data for `gilani-2017` as was used in the original work [30–32] from the authors, though a reduced feature set is available on the Bot Repository. For `gilani-2017` and `caverlee-2011`, the original data provided by the authors [32, 44] contained at least 35% more users than are included in the Bot Repository; we use the larger, original datasets in our results. For the `astroturf` and `varol-2017` datasets published on the Bot Repository, the data only came as a list of user identifiers. Due to the amount of time that has passed since their origination, we did not rehydrate that data or use it in our analysis.

Features. All of the datasets include profile characteristics, which typically include screen name, number of tweets, number of followers, number of following, number of favorites, language, location, timezone, number of Twitter lists on which the user is included, and

[1] https://botometer.osome.iu.edu/bot-repository/datasets.html

others. Additionally, several datasets include a corpus of tweets by each of the users in the dataset. Network relations and associated following/followers behavior are occasionally recorded.

Annotation methods. Ascertaining 'ground truth' labels for bot detection is a challenging task. In most datasets, humans, either the authors of the paper or hired crowdworkers, assigned a 'bot' or 'human' label to each account manually. Previous work has found human annotators have a high level of agreement with each other [32], and accounts for which there is not enough agreement are sometimes excluded from the datasets [22]. Others used heuristics or relied on external sources (e.g., celebrity accounts [celebrity-2019] or accounts that tweeted links from public blacklists [yang-2013]) to assign them. The quality of the hand- and heuristic-labeled datasets depends crucially on the implicit assumption that *humans* are very good at the classification task, and neither the datasets themselves nor the broader literature provide robust evidence that this is the case. To the contrary, recent evidence suggests human annotators are systematically biased towards believing opinion-incongruent accounts are bots [69, 71]. Similarly, there are accounts for which neither bot nor human labels may be appropriate, such as semi-automated accounts or accounts that represent an institutional entity like a corporation or a university [8]. Nevertheless, since other work assumes labels in the data are ground truth and since better annotation methods are not available, we make the same assumption.

3.2 Dataset descriptions.

The datasets which we considered fall into two categories: component datasets, which consist of a single class (human or bot) of accounts, and composite datasets, which consist of a combination of component datasets. Each of the 28 datasets is described briefly below. Unless otherwise specified, the authors of the associated paper hand-labeled the dataset.

social-spambots-1 [10] are spam accounts used during the 2014 Roman mayoral election to promote a particular candidate. social-spambots-2 [10] are spammers who promoted the TalNts app using the hashtag #TALNTS. social-spambots-3 [10] contains accounts which spammed links to products on Amazon, both genuine links to products as well as malicious URLs. traditional-spambots-yang [72] are accounts spamming known malicious links, collected by crawling the Twitter network. genuine-accounts-yang [72] are accounts which did not tweet a malicious link, taken from the same crawling process as traditional-spambots-yang. traditional-spambots-2 [10] includes accounts that share malicious URLs and accounts that repeatedly tag those sharing such content. traditional-spambots-3 [10] and traditional-spambots-4 [10] are accounts spamming job offers. pronbots-2019 [73] are Twitter bots infrequently tweeting links to pornographic sites. elezioni-2015 [12] are manually labeled Italian language accounts that used the hashtag #elezioni2013. political-bots-2019 [73] were collected and identified by Josh Russell (@josh_emerson) to be automated accounts run by a single individual to amplify right-wing influence in the U.S. midterm-2018 [75] includes accounts which used relevant hashtags such as #2018midterms during the 2018 US elections. stock-2018 [11] are accounts with high volumes of tweets that tagged stock microblogs with cashtags. genuine-accounts-cresci [10] is purportedly a random sample of human Twitter users, confirmed to be genuine by their response to a natural language question. These are the accounts that all tweeted "earthquake" mentioned in Section 1 and discussed in Section 4. twibot-2020 [22] was collected by crawling the Twitter network using well-known users as seeds. The accounts were manually labeled by hired crowdworkers. gilani-2017 [32] contains accounts sampled from Twitter's streaming API.

Table 1: Publicly available benchmark datasets for bot detection.

Benchmark dataset	Ref.	Humans	Bots
twibot-2020	[22]	3632	4646
feedback-2019	[73]	380	139
pan-2019	[55]	2060	2060
rtbust-2019	[49]	340	353
midterm-2018	[75]	8092	42446
stock-2018	[11]	6174	7102
cresci-2017	[10]	3474	10894
gilani-2017	[32]	1939	1492
cresci-2015	[12]	1957	3351
yang-2013	[72]	10000	1000
caverlee-2011	[44]	19276	22223

rtbust-2019 [49] contains manually labeled accounts subsampled from all accounts which retweeted Italian tweets during the data collection period. fake-followers-2015 [10] and vendor-purchased-2019 [73] are fake follower accounts purchased from different Twitter online markets. caverlee-2011 [44] were collected via honeypot Twitter accounts and researchers used a human-in-the-loop automated process to label bot and human accounts. celebrity-2019 [73] are manually collected verified celebrity accounts. the-fake-project-2015 [12] consists of accounts which followed @TheFakeProject and successfully completed a CAPTCHA. botwiki-2019 [75] is a list of self-identified benign Twitter bots, for example automated accounts that post generative art or tweet world holidays. feedback-2019 [73] is a collection of about 500 accounts which users of Botometer flagged as being incorrectly labeled by that tool.

Several of the datasets we study are combinations of the above components. cresci-2015 [12] includes the-fake-project-2015, elezioni-2015, and fake-followers-2015. cresci-2017 [10] is composed of fake-followers-2015, genuine-accounts-cresci, the three social-spambots datasets, and the four traditional-spambots datasets. yang-2013 [72] has bots from traditional-spambots-yang and humans from genuine-accounts-yang. pan-2019 [55] includes all of the components of cresci-2015, cresci-2017, varol-2017, plus caverlee-2011 and an additional collection of manually annotated bots and humans not found in any of these. This dataset additionally includes tweet data not present in the original components.

3.3 Methods.

Simple decision rules. While sophisticated machine learning models are able to learn complicated relationships between patterns in input data and their labels, their flexibility generally comes at the

cost of transparency and interpretability. We choose to instantiate 'simple decision rules' as shallow decision trees because their transparency allows us to easily examine exactly why each data point is assigned a label. Similar analyses are significantly more difficult or infeasible with the complex and opaque models predominantly used in bot detection. Researchers have used now-standard explainable machine learning tools like LIME [57] and SHAP [45] to bot detection models [12, 42, 75]. However, none of these can demonstrate as we do that the underlying datasets admit simple, high-performing classifiers that rely on a small number of features. Other simple machine learning models like linear regression, k-means, or a nearest neighbors classifier may be able to provide similar interpretability to shallow decision trees, but the choice of the exact method is not important for our analysis.

We use `scikit-learn`'s implementation of binary decision trees,[2] which are trained recursively on numerical data by choosing a feature-threshold pair (represented by a node) that best splits the data into two groups by class and then learning a decision tree on each group separately. In our case, after a fixed recursive depth (corresponding to tree depth), the classifier outputs a label corresponding to that of the majority of examples in the group; these are leaves of the tree. We only consider trees of depth four or less to ensure the trees can be readily inspected and to avoid overfitting. See Figure 1 for several examples of shallow decision trees trained on benchmark datasets.

Performance metrics. The most commonly reported metrics used in the literature are accuracy and the F1 score. Accuracy is defined as the fraction of examples labeled correctly. When a dataset is not balanced between classes, the accuracy may be misleading since a naive model can achieve a high accuracy by always predicting the majority class. The F1 score in binary classification is the harmonic mean of the model's precision and recall. In our context, low F1 score indicates a classifier that either does not detect a high proportion of the bots or incorrectly labels a large fraction of the humans. The F1 score does not incorporate the number of true negatives, i.e., humans correctly labeled as humans, which is potentially misleading in contexts where bots outnumber humans.

Though the two metrics complement each other, both depend on the proportions of humans and bots in the data. For these reasons, it is hard to compare accuracy and F1 score results across models and datasets with different proportions of bots and humans. To provide additional clarity and comparability, we report the balanced accuracies (bal. acc.) of our classifiers, or the arithmetic mean of the true positive rate and the true negative rate. Balanced accuracy is a less useful metric when one has prior knowledge about the relative proportions of bots and humans in the context where the classifier will be deployed.

4 RESULTS

In this section we present and discuss the results of experiments run on the datasets identified in Section 3. In Section 4.1, we establish that simple decision rules, instantiated as shallow decision trees, yield near-state-of-the-art performance when trained and evaluated on these benchmark datasets and that the simple decision rules are suggestive of sampling and labeling procedures. In Section 4.2, we show that the information contained in one dataset is not informative for classification on other datasets, in other words classifiers trained on one dataset do not generalize to other datasets. Building on prior work [18], we next establish that training a classifier on all of the datasets but one and testing on the held out one yields performance not much better than random guessing. Predictably, both of these results are weaker in the cases where the held out dataset shares some data with the training dataset(s).

In Section 4.3, we assess an assumption made among popular bot detection tools in the literature [17, 18, 60]: each dataset of bots represents one of a small number of types of bots, like spam bots or fake followers. This assumption underpins the approach of training a series of specialized classifiers to detect each bot type and then combining their outputs to provide an overall prediction. We find it is indeed possible to build simple classifiers that perform well on differentiating one type of bot from humans consistent with prior work using more sophisticated models [60].

However, in Section 4.4, we provide evidence that datasets within a given bot type are not drawn from similar distributions; simple decision rules can also differentiate bots within the same type from different datasets. This implies that rather than each dataset being broadly representative of all or a part of the respective subspace containing that type of bot, these component datasets are drawn from narrow and easily separable regions of the sample space, meaning that the signals from each dataset are strongly influenced by sampling and labeling procedures. We conclude that we should not expect that the collection of more — even many more — datasets using similar simplistic sampling and labeling strategies will result in significantly more generalizable classifiers.

4.1 Decision trees on component datasets.

In Table 2, we summarize the performance of our decision trees against that of the state-of-the-art classifier on each dataset. For each dataset, we train a tree of each depth from one to four and report the accuracy and F1 score for the shallowest tree that achieves test accuracy and F1 score within 2.5 percent of best-performing tree; in other words, we favor shallower trees when performance is similar across depths.

In training and testing our models, we use five-fold cross-validation and report the results accordingly. However, `twibot-2020` comes with a train/test split, which we use instead for comparability with prior work. We searched the literature to find the state-of-the-art performance on each dataset. In the cases of `midterm-2018` and `stock-2018`, we could not find papers reporting results on these datasets alone, so we omit entries for the state-of-the-art from the table. (In the literature, e.g., [17, 37, 50, 60, 68], these datasets are frequently used in combination with other ones.) Where a paper reported the results from multiple models or from different test sets, we recorded the maximum score achieved for each metric to make our analysis as conservative as possible. We only compare the performance of our shallow decision trees against state-of-the-art performance when the state of the art includes the same types of features we include. Thus, if a classifier from the literature is trained on profile features like the number of followers or number of tweets, but not text features, we use only profile features in our model. In many cases, the full set of features that were used to

[2] https://scikit-learn.org/stable/modules/tree.html

develop state-of-the-art models were not publicly available. Despite this, for almost all of the datasets, our simple decision rules perform nearly as well as the state-of-the-art. For all datasets except `rtbust-2019`, accuracy is within ten percentage points of the state of the art and all of those but `gilani-2017` and `caverlee-2011` are within five percentage points of the state of the art. For most datasets, the F1 score is similarly close to the state of the art.

For the "earthquake" classifier on `cresci-2017` described in Section 1 and Fig. 1, a cursory inspection of several of the human-labeled accounts on Twitter reveals that they tweeted the word "earthquake," after which an automated account replied, asking for more information about their situation. This points to the human accounts being a sample of convenience from the authors' previous work on detecting natural disasters using social media [16], although the paper publishing this dataset described the data as "a random sample of genuine (human-operated) accounts" [10].

Other datasets yield classifiers which are similarly suggestive of their sampling and labeling procedures. The third tree in Figure 1 shows that the account creation date is an important feature in distinguishing bots from humans in this dataset. This may result from the authors targeting the collection of *spam* accounts on Twitter, for about eight months starting in December 2009. Active spam accounts may have high turnover on the platform if they are reported by users or the platform targets them for suspension, so very few spambots in the dataset were created more than a month before data collection began. The originators of the dataset make a similar observation [44]. For `twibot-20`, the depth one decision tree reported in Table 2 checks if the user is verified or not, and achieves nearly state-of-the-art performance on just this one feature. This may be an artifact of *both* the sampling and labeling strategy. The authors collected accounts by starting from well-known (verified) seed users and collecting the network around those users using a breadth-first traversal [22], and we expect many verified users to follow each other. Accounts were labeled by crowdworkers and discarded if annotators did not sufficiently agree with each other, but verified accounts were automatically labeled human and so they were not at risk of being excluded. The other datasets we consider yield similar analyses. Shallow decision trees reveal that simple models, suggestive of sampling and labeling heuristics, are powerful predictors.

The cases where our models underperform relative to the state of the art are informative: the state-of-the-art model for `rtbust-2019` uses the time between a user's retweets, which we did not have access to. The bots in the dataset were identified by "suspicious" temporal retweeting patterns, so a simple decision rule with access to inter-tweet time may yield much higher performance. `feedback-2019` is a small dataset collected from accounts reported to be misclassified by an earlier version of Botometer, and this is a complex sampling mechanism that may be hard to capture with our simple decision rules. For `yang-2013`, the F1 score is dragged down by low recall (the percentage of bots that were classified as bots), which may be a result of the classifier biasing predictions toward the human label since the dataset itself is more than 90% human. When a classifier is trained on a balanced subsample of this data, the F1 score of a depth four decision tree is within 3% of the state of the art.

Table 2: Performance of our shallow decision trees (SDT) versus state-of-the-art (SOTA) on benchmark datasets.

Dataset	SDT Acc./F1/bal. acc.	Depth	SOTA	SDT - SOTA Acc./F1
`twibot-2020`	0.82/0.86/0.80	1	[20]	-0.05/-0.03
`feedback-2019`	0.80/0.55/0.69	3	[37]	-0.01/-0.15
`rtbust-2019`	0.71/0.73/0.71	4	[49]	-0.22/-0.14
`pan-2019`	0.92/0.91/0.92	2	[21]	-0.03/-0.04
`midterm-2018`	0.97/0.98/0.95	1	[34]	-0.01/ —
`stock-2018`	0.80/0.83/0.80	3	—	— / —
`cresci-2017`	0.98/0.98/0.97	1	[43]	-0.02/-0.02
`gilani-2017`	0.77/0.72/0.76	3	[33]	-0.09/-0.11
`cresci-2015`	0.98/0.98/0.98	3	[12]	-0.01/-0.01
`yang-2013`	0.96/0.71/0.79	4	[72]	-0.03/-0.19
`caverlee-2011`	0.91/0.91/0.90	2	[44]	-0.08/-0.07

We stress that these results are not intended to suggest that simple decision rules make useful classifiers for bot detection, as they can be in other domains [59]; instead, they reveal that bot detection classifiers are limited by the simple sampling and labeling procedures used to construct datasets. If we believed that bot detection is a simple, low-dimensional classification task, we might accept that simple classifiers suffice in this domain, but intuitively we do not expect this to be the case. In what follows, we support this intuition by considering how classifiers *generalize* across datasets.

4.2 Cross-dataset generalization.

Heuristic data collection and labeling practices can be useful when the sample space is also easy to describe. If the sample space is simple, we should expect that classifiers trained on one dataset perform well on others. We present evidence here that this is not the case, by showing that classifiers that perform well on a given dataset typically do not significantly outperform random guessing when tested on each of the others, even when using more expressive models. Similarly, we find that classifiers trained on all but one dataset and tested on the held-out dataset do not perform significantly better than random guessing in most cases. From this, we conclude that the best predictors for separating human and bot users are not consistent across datasets.

Train on one, test on another. For each dataset, we train a random forest using `scikit-learn`'s default parameter settings with 100 trees on each dataset and evaluate test performance on each of the others, using an 80%-20% train-test split. We restrict the feature sets for all training and testing data to the counts of account followers, following, number of tweets and lists each user is on since these features are common to nearly all datasets. This allows us to compare each classifier on a consistent feature set. `yang-2013` and `caverlee-2011` do not contain information on Twitter lists, so we did not include that feature in classifiers trained or tested on those datasets. When we instead used the pairwise intersection of the features available for each train-test combination, the results were qualitatively the same.

For this experiment, we use random forests rather than shallow decision trees to limit the extent to which the poor performance of the models can be attributed insufficient expressiveness, though we did find similar results when using shallow decision trees. Further,

many papers in the literature use random forests to achieve state-of-the-art performance (see, e.g., [32, 72]). Although it is possible that more expressive models, like neural networks, could achieve better cross-dataset performance than random forests for this experiment, we believe this to be implausible, since we can already capture much of the predictive signal in these datasets with simple decision rules. The results are summarized in Table 3, where each row corresponds to the dataset used for training and each column to the dataset used for testing. We see qualitatively similar results for accuracy and F1 scores, but omit those tables for brevity.

In Table 3, the diagonal entries largely reproduce the experiment from Section 4.1 using random forests, showing the unsurprising fact that these can fit each dataset well. Because we use a restricted feature set, the performance of on-diagonal entries is sometimes lower in Table 3 than the performance reported in Table 2. Most of the entries off of the main diagonal show model performance no better than (weighted) random guessing or assigning all examples to a single class.

In a few cases, we see balanced accuracies significantly higher than 0.5. This is the case for classifiers tested on `cresci-2017` and `cresci-2015` as well as to a lesser extent for `midterm-2018` and `yang-2013`. These numbers may be explained by overlap between the datasets, as is the case between `cresci-2017` and `yang-2013` noted in Section 3, or similar collection and labeling strategies conducted by the same research group, such as is the case with `midterm-2019`, `cresci-2017` and `cresci-2015`. Notably, training on `yang-2013` yeilds poor performance when testing the classifier on other datasets, but other datasets yield classifiers that perform well on `yang-2013`, suggesting that it is a dataset that is in some sense easy to test on and not very useful for training classifiers. In some cases, we see significantly worse than 0.5 balanced accuracy. This may arise from the distributions of users being very different across these datasets. A model trained on a dataset where human accounts have very low activity and bots have very high activity will perform very poorly when evaluated on a dataset where these patterns are reversed. The generally unimpressive performance of most of these classifiers suggests that the collections of accounts in each dataset, both bot and human, sit in different parts of the universe of possible accounts.

Leave-one-dataset-out. Following [18, 60], we also train random forest classifiers on all but one dataset and test on the held-out dataset. We report the results in Table 4. In most cases, the signal contained in all but one dataset is not sufficient to predict the labels on the held out dataset. For the cases where this is not true, the datasets share some data in common or were collected and labeled using similar strategies. These results suggest that the best predictors in each dataset are different, and therefore that these benchmark datasets, even when combined, do not generalize to perform well on the others. In the rest of this section, we explore generalization across datasets further, examining whether making assumptions about the types of bots that exist in each dataset can be used to build generalizable predictors.

4.3 Generalization using a bot taxonomy.

We observe in the previous analysis that classifiers trained on one or more benchmark datasets do not give good performance on the others. However, previous work [17, 60] has argued that different datasets may contain different types of bots—e.g., simple bots, spammers, fake followers, self-declared, political bots, and other bots (a catch-all category)—which could account for poor out-of-sample performance in cases where the bot types are different. The need to combine data from different populations is common in prediction problems. For example, to make predictions on patient data from adult and pediatric hospitals, it is necessary to combine data across age groups in ways that account for population differences. Similarly, bot detection may require differentiating between and combining data from qualitatively different types of bots.

Following Botometer [60], we define combined bot-type datasets as follows. `simple` consists of the bot accounts from `caverlee-2011`; `spammers` includes the social or traditional spambots datasets and `pronbots-2019`; `fake-followers` consists of `fake-followers-2015` and `vendor-purchased-2019`; `other-bots` consists of `feedback-2019` and `rtbust-2019`; `financial-bots` consists of `stock-2018`. We do not have access to data for the roughly 500 accounts in `astroturf-2020`, included in `political-bots` in [60].

Table 3: Balanced accuracy of random forests trained on the row-indexed dataset and tested on the column-indexed one.

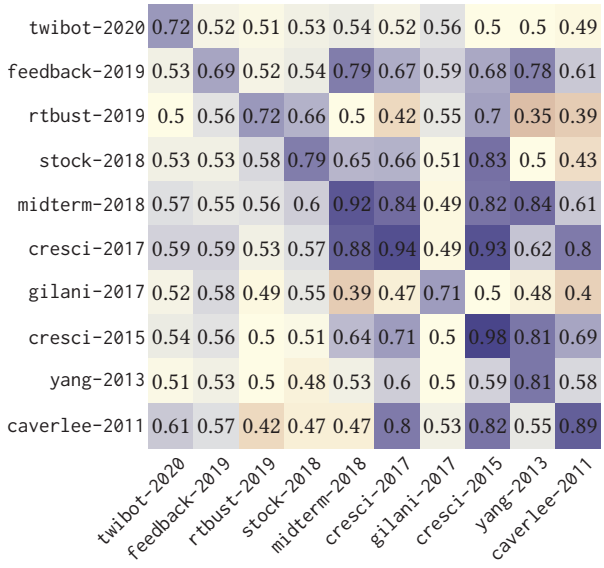

Table 4: Leave-one-dataset-out on benchmark datasets.

Dataset left out	In-sample Acc./F1/bal. acc.	Out-of-sample Acc./F1/bal. acc.
`twibot-2020`	0.80/0.84/0.78	0.52/0.44/0.55
`feedback-2019`	0.78/0.82/0.77	0.64/0.37/0.56
`rtbust-2019`	0.78/0.82/0.76	0.52/0.31/0.53
`midterm-2018`	0.78/0.82/0.77	0.77/0.85/0.77
`stock-2018`	0.76/0.81/0.75	0.55/0.49/0.56
`cresci-2017`	0.77/0.82/0.75	0.83/0.88/0.84
`gilani-2017`	0.79/0.84/0.78	0.58/0.22/0.52
`cresci-2015`	0.81/0.86/0.78	0.87/0.90/0.83
`yang-2013`	0.84/0.88/0.83	0.32/0.21/0.62
`caverlee-2011`	0.71/0.71/0.71	0.56/0.56/0.57

Table 5: Distinguishing each bot type from humans.

Bot type	Acc/F1	Depth
simple	0.86/0.86	3
spammers	0.89/0.89	3
fake-followers	0.94/0.94	4
self-declared	0.88/0.87	3
political-bots	1.00/1.00	3
other-bots	0.76/0.74	2
financial-bots	0.73/0.75	4

In Table 5, we show that these datasets admit accurate yet simple classifiers for detecting a bot type against a sample of human accounts, indicating as in Section 4.1 that the predictive signal contained in each bot type-human dataset is also simple. For each type of bot, we take equal-sized random samples of human accounts and accounts from the bot type and learn a shallow decision tree. We report accuracy and F1, along with the depth that achieves the stated test performance for an 80%-20% train/test split. Since we balance the datasets, accuracy and balanced accuracy are equivalent and we do not report the latter. As before, we favor shallower decision trees when performance is comparable across depths.

Across the bot types, the simple decision rules achieve high accuracy against a baseline of 0.5 (random guessing or a naive classifier) on this task, summarized in Table 5. The performance on political-bots is perfect, perhaps as a result of the small size of this bot type in our data. We see only modest performance improvements by using a random forest rather than a simple decision rule and we omit the details of those results here.

We next learn classifiers that predict which dataset an account comes from within bot types in order to understand whether accounts from different datasets in the same category are substantively similar to one another.

4.4 Distinguishing datasets within bot types.

Continuing with the taxonomic viewpoint, here we address whether these datasets can be used to train models that generalize well *only within a single class of bots*. We provide evidence towards the negative by showing that it is possible to distinguish accounts from different datasets within a bot type. Even when restricted to the subspace of spam bots, for instance, the constituent datasets in that class are sampled in such distinct ways that a classifier can easily distinguish them. We also observe a similar phenomenon with the human accounts, indicating that the strategies used to gather and label human accounts also results in samples being drawn from different parts of that sample space. In Table 6, we report accuracy and balanced accuracy for the task of predicting which dataset an account comes from for accounts within a given type. Rather than a binary prediction task of bot or not, this is a multiclass prediction task where each account is labeled with the dataset it comes from, omiting the classes that consist of only one dataset. We train shallow decision trees to predict these labels, summarized in Table 6. Again we see minor performance improvements by using random forest classifiers, but prefer the decision trees for their interpretability.

For humans, we have access to six datasets, so a naive or random classifier should achieve a balanced accuracy of less than 0.17.

Table 6: Distinguishing an account's dataset by type.

Type	Acc./bal. acc.	Depth	Num. datasets
humans	0.67/0.43	3	6
spammers	0.97/0.75	4	7
fake followers	0.97/0.94	1	2
other bots	0.91/0.77	2	3

However, we achieve accuracy greater than 0.65 and balanced accuracy of greater than 0.4, indicating that this classifier can identify which dataset a human account belongs to with probability significantly better than random chance. For spammers, we achieve balanced accuracy of 0.75 and nearly perfect accuracy on seven datasets, against a naive baseline of less than 0.15. The classifier here gains predictive power from artifacts of the sampling strategies used to collect the data. As examples, most of the accounts in pronbots-2019 liked several tweets whereas other spammers did not; accounts in social-spambots-1 sent many tweets in Italian whereas other spammers used English or only tweeted URLs; still other datasets can be separated by basic account activity features like the number of accounts followed.

5 CONCLUSION & DISCUSSION

In this paper, we provided evidence that Twitter bot detection datasets are limited by their simple collection strategies, explaining their poor generalization. These findings suggest that a limiting factor in advancing bot detection research is a lack of availability of robust, high quality data. If accounts are improperly discarded for being likely bots because they have not liked enough tweets or their account was created at the wrong time or they were accessed from a particular kind of device, that introduces errors into a downstream analysis and if any of those features correlate with the topic of interest, these errors may bias the conclusions of that analysis.

This work also highlights the broader issue of how opaque machine learning techniques may obscure certain flaws in the underlying data. While we examine the Twitter bot detection ecosystem in this work, our methods and recommendations should apply broadly to the study of any online social media platform and to applied machine learning research wherever datasets are reused across contexts. We encourage researchers who originate and publish datasets to be explicit about their sampling and labeling procedures, perhaps using existing documentation tools [29]. Researchers and engineers building bot detection tools should carefully examine their training data, possibly via the use of simple models like shallow decision trees, to ensure the data captures the complexity of the space the more expressive model attempts to describe. Finally, those using bot detection as a preprocessing step should carefully consider how errors might propagate through this process and what kinds of biases might be present in their analysis as a result. Given how seriously the academic community and general public treat the problem of bot detection on social media, we also hope that the platforms themselves can facilitate work in this area by providing rich and robust data with high quality ground truth labels.

REFERENCES

[1] Eiman Alothali, Motamen Salih, Kadhim Hayawi, and Hany Alashwal. 2022. Bot-MGAT: A Transfer Learning Model Based on a Multi-View Graph Attention Network to Detect Social Bots. *Applied Sciences* 16 (2022). https://doi.org/10.3390/app12168117

[2] Adam Badawy, Emilio Ferrara, and Kristina Lerman. 2018. Analyzing the Digital Traces of Political Manipulation: The 2016 Russian Interference Twitter Campaign. In *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*. 258–265. https://doi.org/10.1109/ASONAM.2018.8508646

[3] Marco T. Bastos and Dan Mercea. 2019. The Brexit Botnet and User-Generated Hyperpartisan News. *Social Science Computer Review* 37, 1 (2019), 38–54. https://doi.org/10.1177/0894439317734157 arXiv:https://doi.org/10.1177/0894439317734157

[4] Alessandro Bessi and Emilio Ferrara. 2016. Social bots distort the 2016 US Presidential election online discussion. *First Monday* 21, 11 (2016). https://doi.org/10.5210/fm.v21i11.7090

[5] Christopher Bouzy. 2018. Bot sentinel, Platform developed to detect and track political bots, trollbots, and untrustworthy accounts. https://botsentinel.com/

[6] David A. Broniatowski, Amelia M. Jamison, SiHua Qi, Lulwah AlKulaib, Tao Chen, Adrian Benton, Sandra C. Quinn, and Mark Dredze. 2018. Weaponized Health Communication: Twitter Bots and Russian Trolls Amplify the Vaccine Debate. *American Journal of Public Health* 108, 10 (2018), 1378–1384. https://doi.org/10.2105/AJPH.2018.304567 arXiv:https://doi.org/10.2105/AJPH.2018.304567 PMID: 30138075.

[7] Daejin Choi, Selin Chun, Hyunchul Oh, Jinyoung Han, and Ted Kwon. 2020. Rumor Propagation is Amplified by Echo Chambers in Social Media. *Scientific Reports* 10, 310 (2020). https://doi.org/10.1038/s41598-019-57272-3

[8] Zi Chu, Steven Gianvecchio, Haining Wang, and Sushil Jajodia. 2012. Detecting Automation of Twitter Accounts: Are You a Human, Bot, or Cyborg? *IEEE Transactions on Dependable and Secure Computing* 9, 6 (2012), 811–824. https://doi.org/10.1109/TDSC.2012.75

[9] Stefano Cresci. 2020. A Decade of Social Bot Detection. *Commun. ACM* 63, 10 (sep 2020), 72–83. https://doi.org/10.1145/3409116

[10] Stefano Cresci, Roberto Di Pietro, Marinella Petrocchi, Angelo Spognardi, and Maurizio Tesconi. 2017. The paradigm-shift of social spambots: Evidence, theories, and tools for the arms race.. In *26th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 963–972. https://doi.org/10.1145/3041021.3055135

[11] Stefano Cresci, Fabrizio Lillo, Daniele Regoli, Serena Tardelli, and Maurizio Tesconi. 2019. $ FAKE: Evidence of Spam and Bot Activity in Stock Microblogs on Twitter. (2019).

[12] Stefano Cresci, Roberto Di Pietro, Marinella Petrocchi, Angelo Spognardi, and Maurizio Tesconi. 2015. Fame for sale: Efficient detection of fake Twitter followers. *Decision Support Systems* 80 (2015), 56–71. https://doi.org/10.1016/j.dss.2015.09.003

[13] Stefano Cresci, Roberto Di Pietro, Marinella Petrocchi, Angelo Spognardi, and Maurizio Tesconi. 2016. DNA-Inspired Online Behavioral Modeling and Its Application to Spambot Detection. *IEEE Intelligent Systems* 31, 5 (2016), 58–64. https://doi.org/10.1109/MIS.2016.29

[14] Stefano Cresci, Roberto Di Pietro, Marinella Petrocchi, Angelo Spognardi, and Maurizio Tesconi. 2017. Exploiting digital DNA for the analysis of similarities in Twitter behaviour. In *IEEE International Conference on Data Science and Advanced Analytics (DSAA)*. 686–695. https://doi.org/10.1109/DSAA.2017.57

[15] Stefano Cresci, Roberto Di Pietro, Marinella Petrocchi, Angelo Spognardi, and Maurizio Tesconi. 2018. Social Fingerprinting: Detection of Spambot Groups Through DNA-Inspired Behavioral Modeling. *IEEE Transactions on Dependable and Secure Computing* 15, 4 (2018), 561–576. https://doi.org/10.1109/TDSC.2017.2681672

[16] Stefano Cresci, Maurizio Tesconi, Andrea Cimino, and Felice Dell'Orletta. 2015. A Linguistically-Driven Approach to Cross-Event Damage Assessment of Natural Disasters from Social Media Messages. In *Proceedings of the 24th International Conference on World Wide Web* (Florence, Italy) *(WWW '15 Companion)*. Association for Computing Machinery, New York, NY, USA, 1195–1200. https://doi.org/10.1145/2740908.2741722

[17] Ilias Dimitriadis, Konstantinos Georgiou, and Athena Vakali. 2021. Social Botomics: A Systematic Ensemble ML Approach for Explainable and Multi-Class Bot Detection. *Applied Sciences* 11, 21 (2021). https://doi.org/10.3390/app11219857

[18] Juan Echeverria, Emiliano De Cristofaro, Nicolas Kourtellis, Ilias Leontiadis, Gianluca Stringhini, and Shi Zhou. 2018. LOBO: Evaluation of Generalization Deficiencies in Twitter Bot Classifiers. In *34th Annual Computer Security Applications Conference*. Association for Computing Machinery, 137–146. https://doi.org/10.1145/3274694.3274738

[19] Tuğrulcan Elmas, Rebekah Overdorf, and Karl Aberer. 2022. Characterizing Retweet Bots: The Case of Black Market Accounts. In *Proceedings of the International AAAI Conference on Web and Social Media*, Vol. 16. 171–182.

[20] Shangbin Feng, Zhaoxuan Tan, and Minnan Luo Rui L and. 2022. Heterogeneity-aware Twitter Bot Detection with Relational Graph Transformers. In *AAAI Conference on Artificial Intelligence*. Association for the Advancement of Artificial Intelligence, 3977–3985. https://doi.org/10.1609/aaai.v36i4.20314

[21] Shangbin Feng, Herun Wan, Ningnan Wang, Jundong Li, and Minnan Luo. 2021. SATAR: A Self-supervised Approach to Twitter Account Representation Learning and its Application in Bot Detection. In *30th ACM International Conference on Information & Knowledge Management (CIKM)*. Association for Computing Machinery, 3808–3817. https://doi.org/10.1145/3459637.3481949

[22] Shangbin Feng, Herun Wan, Ningnan Wang, Jundong Li, and Minnan Luo. 2021. TwiBot-20: A Comprehensive Twitter Bot Detection Benchmark. (2021), 4485–4494.

[23] Shangbin Feng, Herun Wan, Ningnan Wang, and Minnan Luo. 2021. BotRGCN: Twitter Bot Detection with Relational Graph Convolutional Networks. In *2021 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*. Association for Computing Machinery, 236–239. https://doi.org/10.1145/3487351.3488336

[24] Emilio Ferrara. 2017. Disinformation and Social Bot Operations in the Run Up to the 2017 French Presidential Election. *First Monday* 22, 8 (2017). https://doi.org/10.5210/fm.v22i18.8005

[25] Emilio Ferrara. 2020. What types of COVID-19 conspiracies are populated by Twitter bots? *First Monday* (05 2020). https://doi.org/10.5210/fm.v25i6.10633

[26] Florian Gallwitz and Michael Kreil. 2021. The Rise and Fall of 'Social Bot' Research. https://ssrn.com/abstract=3814191

[27] Florian Gallwitz and Michael Kreil. 2022. Investigating the Validity of Botometer-based Social Bot Studies. arXiv:2207.11474 [cs.SI]

[28] Andres Garcia-Silva, Cristian Berrio, and José Manuel Gómez-Pérez. 2019. An Empirical Study on Pre-trained Embeddings and Language Models for Bot Detection. In *4th Workshop on Representation Learning for NLP (RepL4NLP)*. Association for Computational Linguistics, 148–155. https://doi.org/10.18653/v1/W19-4317

[29] Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé Iii, and Kate Crawford. 2021. Datasheets for datasets. *Commun. ACM* 64, 12 (2021), 86–92.

[30] Zafar Gilani, Reza Farahbakhsh, Gareth Tyson, and Jon Crowcroft. 2019. A Large-scale Behavioural Analysis of Bots and Humans on Twitter. *ACM Transactions on the Web* 13, 1 (2019). https://doi.org/10.1145/3298789

[31] Zafar Gilani, Reza Farahbakhsh, Gareth Tyson, Liang Wang, and Jon Crowcroft. 2017. Of Bots and Humans (on Twitter). In *IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*. Association for Computing Machinery, 349–354. https://doi.org/10.1145/3110025.3110090

[32] Zafar Gilani, Ekaterina Kochmar, and Jon Crowcroft. 2017. Classification of Twitter Accounts into Automated Agents and Human Users. In *2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*. 489–496.

[33] Zafar Gilani, Ekaterina Kochmar, and Jon Crowcroft. 2020. Classification of twitter accounts into automated agents and human users. In *IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*. 489–496. https://doi.org/10.1145/3110025.3110091

[34] Salvatore Giorgi, Lyle Ungar, and H. Andrew Schwartz. 2021. Characterizing Social Spambots by their Human Traits. In *The Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*. 5148–5158. https://doi.org/10.18653/v1/2021.findings-acl.457

[35] Sandra González-Bailón, Valeria d'Andrea, Deen Freelon, and Manlio De Domenico. 2022. The advantage of the right in social media news sharing. *PNAS Nexus* 1, 3 (07 2022). https://doi.org/10.1093/pnasnexus/pgac137 arXiv:https://academic.oup.com/pnasnexus/article-pdf/1/3/pgac137/45484944/pgac137.pdf pgac137.

[36] Yuriy Gorodnichenko, Tho Pham, and Oleksandr Talavera. 2018. *Social Media, Sentiment and Public Opinions: Evidence from #Brexit and #USElection*. NBER Working Papers 24631. National Bureau of Economic Research, Inc. https://ideas.repec.org/p/nbr/nberwo/24631.html

[37] Qinglang Guo, Haiyong Xie, Yangyang Li, Wen Ma, and Chao Zhang. 2022. Social Bots Detection via Fusing BERT and Graph Convolutional Networks. *Symmetry* 14, 1 (2022). https://doi.org/10.3390/sym14010030

[38] Maryam Heidari and James H Jones. 2020. Using BERT to Extract Topic-Independent Sentiment Features for Social Media Bot Detection. In *11th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*. 542–547. https://doi.org/10.1109/UEMCON51285.2020.9298158

[39] Loukas Ilias and Ioanna Roussaki. 2021. Detecting malicious activity in Twitter using deep learning techniques. *Applied Soft Computing* 107 (2021).

[40] S. Mo Jang, Tieming Geng, Jo-Yun Queenie Li, Ruofan Xia, Chin-Tser Huang, Hwalbin Kim, and Jijun Tang. 2018. A computational approach for examining the roots and spreading patterns of fake news: Evolution tree analysis. *Computers in Human Behavior* 84 (2018), 103–113. https://doi.org/10.1016/j.chb.2018.02.032

[41] Tobias Keller and Ulrike Klinger. 2018. Social Bots in Election Campaigns: Theoretical, Empirical, and Methodological Implications. *Political Communication* 36, 1 (2018), 171–189. https://doi.org/10.1080/10584609.2018.1526238

[42] Maria Kouvela, Ilias Dimitriadis, and Athena Vakali. 2020. Bot-Detective: An Explainable Twitter Bot Detection Service with Crowdsourcing Functionalities. In *Proceedings of the 12th International Conference on Management of Digital EcoSystems* (Virtual Event, United Arab Emirates) *(MEDES '20)*. Association for Computing Machinery, New York, NY, USA, 55–63. https://doi.org/10.1145/3415958.3433075

[43] Sneha Kudugunta and Emilio Ferrara. 2018. Deep neural networks for bot detection. *Information Sciences* 467 (2018), 312–322. https://doi.org/10.1016/j.ins.2018.08.019

[44] Kyumin Lee, Brian Eoff, and James Caverlee. 2011. A Long-Term Study of Content Polluters on Twitter. In *Fifth International AAAI Conference on Weblogs and Social Media*. Association for the Advancement of Artificial Intelligence, 185–192.

[45] Scott M Lundberg and Su-In Lee. 2017. A Unified Approach to Interpreting Model Predictions. In *Advances in Neural Information Processing Systems 30*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Eds.). Curran Associates, Inc., 4765–4774. http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf

[46] Linhao Luo, Xiaofeng Zhang, Xiaofei Yang, and Weihuang Yang. 2019. Deepbot: A Deep Neural Network based approach for Detecting Twitter Bots. *IOP Conference Series: Materials Science and Engineering* 719, 1 (2019). https://doi.org/10.1088/1757-899x/719/1/012063

[47] Franziska Martini, Paul Samula, Tobias R Keller, and Ulrike Klinger. 2021. Bot, or not? Comparing three methods for detecting social bots in five political discourses. *Big Data & Society* 8, 2 (2021), 20539517211033566. https://doi.org/10.1177/20539517211033566 arXiv:https://doi.org/10.1177/20539517211033566

[48] David Martín-Gutiérrez, Gustavo Hernández-Peñaloza, Alberto Belmonte Hernández, Alicia Lozano-Diez, and Federico Álvarez. 2021. A Deep Learning Approach for Robust Detection of Bots in Twitter Using Transformers. *IEEE Access* 9 (2021), 54591–54601. https://doi.org/10.1109/ACCESS.2021.3068659

[49] Michele Mazza, Stefano Cresci, Marco Avvenuti, Walter Quattrociocchi, and Maurizio Tesconi. 2019. RTbust: Exploiting Temporal Patterns for Botnet Detection on Twitter. In *10th ACM Conference on Web Science*. Association for Computing Machinery, 183–192. https://doi.org/10.1145/3292522.3326015

[50] Guanyi Mou and Kyumin Lee. 2020. Malicious bot detection in online social networks: arming handcrafted features with deep learning. In *International Conference on Social Informatics*. Springer, 220–236.

[51] Elon Musk. 2022. Bot Percentage Thread. https://twitter.com/elonmusk/status/1555950698252181507

[52] Leonardo Nizzoli, Serena Tardelli, Marco Avvenuti, Stefano Cresci, Maurizio Tesconi, and Emilio Ferrara. 2020. Charting the Landscape of Online Cryptocurrency Manipulation. *IEEE Access* 8 (2020), 113230–113245. https://doi.org/10.1109/ACCESS.2020.3003370

[53] Gordon Pennycook, Ziv Epstein, Mohsen Mosleh, Antonio A. Arechar, Dean Eckles, and David G. Rand. 2021. Shifting attention to accuracy can reduce misinformation online. *Nature* 592, 6380 (2021), 590–595. https://doi.org/10.1038/s41586-021-03344-2

[54] Francesco Pierri, Alessandro Artoni, and Stefano Ceri. 2020. Investigating Italian disinformation spreading on Twitter in the context of 2019 European elections. *PloS one* 15, 1 (2020), e0227821. https://doi.org/10.1371/journal.pone.0227821

[55] Francisco Rangel and Paolo Rosso. 2015. Overview of the 7th Author Profiling Task at PAN 2019: Bots and Gender Profiling in Twitter. In *CLEF Evaluation Labs and Workshop Working Notes Papers*.

[56] Adrian Rauchfleisch and Jonas Kaiser. 2020. The False positive problem of automatic bot detection in social science research. *PLOS ONE* 15, 10 (10 2020), 1–20. https://doi.org/10.1371/journal.pone.0241045

[57] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (San Francisco, California, USA) *(KDD '16)*. Association for Computing Machinery, New York, NY, USA, 1135–1144. https://doi.org/10.1145/2939672.2939778

[58] Luigi Rovito, Lorenzo Bonin, Luca Manzoni, and Andrea De Lorenzo. 2022. An Evolutionary Computation Approach for Twitter Bot Detection. *Applied Sciences* 12, 12 (2022). https://doi.org/10.3390/app12125915

[59] Cynthia Rudin. 2019. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence* 1, 5 (01 May 2019), 206–215. https://doi.org/10.1038/s42256-019-0048-x

[60] Mohsen Sayyadiharikandeh, Onur Varol, Kai-Cheng Yang, Alessandro Flammini, and Filippo Menczer. 2020. Detection of Novel Social Bots by Ensembles of Specialized Classifiers. In *29th ACM International Conference on Information & Knowledge Management*. Association for Computing Machinery, 2725–2732. https://doi.org/10.1145/3340531.3412698

[61] Chengcheng Shao, Giovanni Luca Ciampaglia, Onur Varol, Kai-Cheng Yang, Alessandro Flammini, and Filippo Menczer. 2018. The spread of low-credibility content by social bots. *Nature Communications* 9, 4787 (2018), 1–9. https://doi.org/10.1038/s41467-018-06930-7

[62] Chengcheng Shao, Pik-Mai Hui, Lei Wang, Xinwen Jiang, Alessandro Flammini, Filippo Menczer, and Giovanni Luca Ciampaglia. 2018. Anatomy of an online misinformation network. *PloS one* 13, 4 (2018), e0196087. https://doi.org/10.1371/journal.pone.0196087

[63] Kai Shu, Deepak Mahudeswaran, Suhang Wang, Dongwon Lee, and Huan Liu. 2020. FakeNewsNet: A Data Repository with News Content, Social Context, and Spatiotemporal Information for Studying Fake News on Social Media. *Big Data* 8, 3 (2020), 171–188. https://doi.org/10.1089/big.2020.0062

[64] Massimo Stella, Emilio Ferrara, and Manlio De Domenico. 2018. Bots increase exposure to negative and inflammatory content in online social systems. *Proceedings of the National Academy of Sciences* 115, 49 (2018), 12435–12440. https://doi.org/10.1073/pnas.1803470115

[65] M. Buğra Torusdağ, Mucahid Kutlu, and Ali Aydın Selçuk. 2020. Are We Secure from Bots? Investigating Vulnerabilities of Botometer. In *2020 5th International Conference on Computer Science and Engineering (UBMK)*. 343–348. https://doi.org/10.1109/UBMK50275.2020.9219433

[66] Twitter. 2021. FORM 10-K. https://www.sec.gov/Archives/edgar/data/1418091/000141809121000031/twtr-20201231.htm

[67] Soroush Vosoughi, Deb Roy, and Sinan Aral. 2018. The spread of true and false news online. *Science* 359, 6380 (2018), 1146–1151. https://doi.org/10.1126/science.aap9559 arXiv:https://www.science.org/doi/pdf/10.1126/science.aap9559

[68] Xiujuan Wang, Qianqian Zheng, Kangfeng Zheng, Yi Sui, Siwei Cao, and Yutong Shi. 2021. Detecting social media bots with variational autoencoder and k-nearest neighbor. *Applied Sciences* 11, 12 (2021), 5482.

[69] Magdalena Wischnewski, Rebecca Bernemann, Thao Ngo, and Nicole Krämer. 2021. Disagree? You Must Be a Bot! How Beliefs Shape Twitter Profile Perceptions. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 160, 11 pages. https://doi.org/10.1145/3411764.3445109

[70] Liang Wu, Xia Hu, Fred Morstatter, and Huan Liu. 2017. Adaptive Spammer Detection with Sparse Group Modeling. In *Eleventh International AAAI Conference on Web and Social Media (ICWSM)*. Association for the Advancement of Artificial Intelligence, 319–326.

[71] Harry Yaojun Yan, Kai-Cheng Yang, Filippo Menczer, and James Shanahan. 2021. Asymmetrical perceptions of partisan political bots. *New Media & Society* 23, 10 (2021), 3016–3037. https://doi.org/10.1177/1461444820942744 arXiv:https://doi.org/10.1177/1461444820942744

[72] Chao Yang, Robert Harkreader, and Guofei Gu. 2013. Empirical evaluation and new design for fighting evolving twitter spammers. *IEEE Transactions on Information Forensics and Security* 8, 8 (2013), 1280–1293. https://doi.org/10.1109/TIFS.2013.2267732

[73] Kai-Cheng Yang, Onur Varol, Clayton A. Davis, Emilio Ferrara, Alessandro Flammini, , and Filippo Menczer. 2019. Arming the public with artificial intelligence to counter social bots. *Human Behavior and Emerging Technologies* 1 (2019), 48–68. https://doi.org/10.1002/hbe2.115

[74] Kai-Cheng Yang, Emilio Ferrara, and Filippo Menczer. 2022. Botometer 101: Social bot practicum for computational social scientists. *arXiv preprint arXiv:2201.01608* (2022).

[75] Kai-Cheng Yang, Onur Varol, Pik-Mai Hui, and Filippo Menczer. 2020. Scalable and Generalizable Social Bot Detection through Data Selection. In *AAAI Conference on Artificial Intelligence*. Association for the Advancement of Artificial Intelligence, 1096–1103. https://doi.org/10.1609/aaai.v34i01.5460