Beyond Following: Augmenting Bot Detection with the Integration of Behavioral Patterns

Sebastian Reiche[1(✉)], Sarel Cohen[2], Kirill Simonov[1], and Tobias Friedrich[1]

[1] Hasso Plattner Institute, University of Potsdam, Potsdam, Germany
sebastian.reiche@student.hpi.de, {kirill.simonov,tobias.friedrich}@hpi.de
[2] The Academic College of Tel Aviv-Yaffo, Tel Aviv, Israel
sarelco@mta.ac.il

Abstract. Social media platforms like Twitter revolutionized online communication. But this new era of interaction has brought with it a challenge—the widespread presence and influence of bot accounts. These bots are rapidly evolving, making traditional detection methods increasingly ineffective and allowing malicious actors to influence public discourse. While existing bot detection methods report high performance, such results might actually be connected to shortcomings in dataset collection and labeling practices, rather than reflecting their true ability to detect bots, casting doubt on their true reliability. Our study introduces higher-order behavior-based relations, including Co-Retweet, and Co-Hashtag, derived from the TwiBot-22 dataset. By leveraging these new relations in the BotRGCN architecture, we shift the emphasis from isolated accounts to coordinated group dynamics, making it more challenging for bot developers to evade detection. This strategy not only acknowledges the limitations and inherent biases presented in existing bot detection techniques, but also presents a way to address them. Our experiments support this approach as a promising way forward to tackle challenges in bot detection.

Keywords: bot detection · graph neural network · relation enhancement

1 Introduction

Social networks like Twitter — currently in the process of rebranding to X — have become an integral part of our social lives. They revolutionized the way we communicate online, shape public discourse, and provide access to the latest news and opinions. One major issue within social networks is the prevalence of bot accounts, which have been known to influence public opinion, especially in critical areas like politics or financial markets [2]. It is notoriously hard to estimate the true extent of the presense of bots on social media platforms, and platforms may be incentivized to misrepresent them, as it could negatively impact

H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2023, SCI 1141, pp. 250–259, 2024.
https://doi.org/10.1007/978-3-031-53468-3_21

revenue[1]. In 2017, Varol et al. estimated that bots may make up to 15% of all Twitter accounts [13]. In another study, Cresci et al. analyzed Twitter dicussions concerning the US stock market, and concluded that up to 71% of the engaged users might be bots [4].

Furthermore, bots seem to become more sophisticated over time [2,6], a phenomenon often referred to as *bot evolution*. This term describes the adversarial cycle in which newer bots evade increasingly more sophisticated bot detection measures, by becoming progressively indistinguishable from real humans. An illustrative example of this effect are the results reported in early 2017 by Cresci et al. [3]. In this experiment, the users were tasked to tell bots apart from legitimate users, only being able to correctly identify newer bots with a 24% accuracy, compared to 91% on older bots. Cresci [2] points out that bot detection methods must be able to distinguish between genuine users and bots, who disguise as genuine users through stolen profile pictures and neutral messages. This complexity has been further intensified by the advancement of artificial intelligence, particularly generative AI, which makes it more difficult to separate individual bot accounts from genuine users. The increasing difficulty in distinguishing between human-written and AI-generated text underscores the complexity of the issue. This is highlighted by OpenAI's decision to disable their AI classifier as of July 2023 due to low rate of accuracy in distinguishing between AI-generated and human-generated content.[2]

In response to these challenges with feature-based methods, graph-based methods are emerging as an alternative, due to their proven effectiveness in recognizing coordinated, synchronized activities [6]. By leveraging these techniques it is not only possible to study how users interact with content, but also how they interact with other users. The rationale behind these approaches stems from the assumption that human-guided and authentic activities typically display more variability than their automated, inauthentic counterparts. This emphasizes the need to move beyond analyzing individual accounts to focusing on patterns of suspicious coordination within groups.

However, research by Elmas et al. [5] on retweet bots, utilizing data from services previously purchased on black market sites, discovered discrepancies in common assumptions about bot characteristics. This included, but was not limited to, areas of *volume of activity*, *diversity*, *following and followers* and *temporality*. They illustrated that bots may emerge from compromised accounts, acting as bots only for certain period of time, and did not find a single case of one bot following another one. Such insights should prompt researchers to critically assess, whether the metrics used to evaluate the performance of bot detection methods are in fact contributing to improving downstream applications. Hays et al. [8] argued that this is currently not the case for Twitter bot detection tools, attributing high performance to simplistic collection and labeling practices of the datasets employed. Separately, Martini et al. [10] observed that different methods

[1] https://storage.courtlistener.com/recap/gov.uscourts.cand.330648/gov.uscourts.cand.330648.257.0.pdf.

[2] https://openai.com/blog/new-ai-classifier-for-indicating-ai-written-text.

yield remarkably different results in comparison. This implies that current tools may not be ready for downstream usage and may result in the misclassification of many users [11].

With the heightened difficulty in identifying individual bot accounts, we focus our efforts on group activities and their coordinated behavior patterns. Our work is in line with trends in recent research that focuses more on actions and behavior of groups of accounts rather than on the classification of individual accounts [1].

We investigate the potential of new sets of relations that are challenging to circumvent; any attempts to do so could drastically limit the functionality of organized automated actions by restricting their common operational patterns. The goal of our research is to determine the feasibility of utilizing coordination patterns for the purpose of bot detection, with due consideration to both the inherent complexities and data restrictions. By recognizing these challenges, we contrast first-order behavior-based relations, such as retweets (a user sharing a tweet), with higher-order relations like co-retweet (two users retweet the same tweet) and co-hashtag (two users tweet the same hashtag more than a certain number of times). The former highlights direct user behavior, while the latter reveals shared interests or subjects, uncovering subtler collective actions. This approach is set against the current conventional method of utilizing follow relations, which are more static. Utilizing the same dataset and graph neural network architecture across our experiments, we conduct a comparative study between the conventional follow relations and those centered around behavioral patterns to assess their impact on bot detection, avoiding the introduction of new uncertainties through algorithmic changes or dataset variations. Though our results did not surpass the conventional approach, they remain competitive in terms of accuracy and F1-score, demonstrating the viability of this approach. To the best of our knowledge, this is the first work that integrates higher-order relations in a behavior-based approach for bot detection.

2 Methodology

2.1 Dataset

We utilize the `TwiBot-22` dataset for our experiments. Compared to previous datasets, `TwiBot-22` includes a broader and more diverse range of relations. For an in-depth exploration of the dataset's conceptual framework, we refer the readers to the work of Feng at al. [6] that introduced `TwiBot-22`. Previous bot detection methods were constrained to rely only on FOLLOWER/FOLLOWING relationships between user entities and an implicit relation between users and their tweets. The `TwiBot-22` dataset encompasses extensive 14 different kinds of relations. In this work we leverage the FOLLOWER (user A is followed by user B), FOLLOWING (user A follows user B), RETWEET (tweet A retweets tweet B), POST (user A posts tweet B), and DISCUSS (tweet A discusses hashtag B) relations. We believe that this range of relations offers a lot of potential for future development of more sophisticated and accurate bot detection methods. The accessibility of these diverse relations not only enhances our analytical capabilities

but also allows us to reveal hidden connections between users, cross-referencing entities in ways previously unattainable. We refer the reader to Table 1 for an overview of `TwiBot-22`, comprising both statistics as well as an exploration of some of the characteristics that differentiate humans from bots. The left side of the table provides a quantitative overview of the dataset. On the right side, a more nuanced analysis of the variances in human and bot behavior. Key contrasts include variations in tweet and following/follower count[3] as well as ratios like hashtag-to-tweet, revealing discrepancies between the two types of accounts. This comparative analysis offers valuable insight that guides the process of deriving new relations. We explore these aspects further and delve into more detail in subsequent sections, specifically in Subsect. 2.3.

Table 1. Statistics (left) and in-depth analysis (right) of human and bot characteristics in `TwiBot-22`. ∗users with at least 1 tweet. † with at least 1 follower / following.

Measurement	Human	Bot	Total
Users (all)	860,057	139,943	1,000,000
Users (min. 1 tweet)	818,613	115,259	933,872
Tweet	81,250,102	6,967,355	88,217,457
Following	1,038,302	78,353	1,116,655
Followers	2,383,574	243,405	2,626,979
Retweet	1,482,911	97,732	1,580,643
Hashtags	56,353,776	9,646,857	66,000,633

Measurement	Human	Bot	Diff.
Mean tweet count*	99.25	60.45	−48.59%
Median tweet count*	56.00	40.00	−33.33%
Mean following count†	200.22	170.21	−16.20%
Mean follower count†	124.59	59.40	−70.86%
Mean retweet count*	200.39	104.98	−62.49%
Ratio: tweet / retweet	54.79	71.29	+26.17%
Ratio: hashtag / tweet	0.69	1.38	+66.66%

While `TwiBot-22` is believed to contain high-quality labels, it is important to recognize that we cannot entirely dismiss the possibility of underlying biases towards older notions of bot characteristics. A potential bias could be introduced by the use of non-transparent hand-crafted labeling functions and dependence on existing bot detection methods. These methods are often trained on follow relationships, an assumption we challenged in the introduction. This reliance on possibly flawed assumptions may further deviate bot detection in the wrong direction. In addition, recent evidence indicates that classifiers performing exceptionally within one dataset may significantly underperform when applied to others, even when employing more sophisticated models [8]. This may be attributed to the reliance on inherently unstable features present in the initial training data. Therefore, although `TwiBot-22`'s expert-guide process signals a marked improvement, the broader methodology might compromise the dataset's overall effectiveness. Nevertheless, we assume the labels in the data to be the ground truth. This assumption is made due to the lack of better annotation methods and inherent difficulty of this problem.

[3] Somewhat counter-intuitively, the total following and follower counts do not match. This is due to specifics of data collection, see [6] for insights into the process.

2.2 BotRGCN

`BotRGCN` (Bot detection with Relational Graph Convolutional Networks) [7] is a graph-based method for Twitter bot detection. The model first creates a multimodal encoding by jointly encoding multiple numerical and categorical user properties, as well as encoding user tweets and descriptions using a pre-trained RoBERTa model. These encodings serve to represent individual users, capturing diverse aspects of their behavior and characteristics. A heterogeneous graph is constructed by defining multiple relational neighborhoods for each Twitter user. `BotRGCN` applies relational graph convolutional networks (RGCN), which support a variable number of relations, allowing the model to capture complex patterns of interactions between users. We chose to work with `BotRGCN` due to its modular and well-designed architecture that allows for easy modification and experimentation. The model was used with the initialization of hyperparameters as found in the original implementation, available at the corresponding Github repository.[4] Adjustments were made to accommodate the specific number of categorical and numerical properties in `TwiBot-22`. The architecture and specific components of `BotRGCN` are further detailed in Table 2.

Table 2. Architecture of the `BotRGCN` model. Variables: D: embedding size, D_s: description size, T_s: tweet size, N_s: numerical properties size, C_s: categorical properties size. The input layers' outputs are concatenated before processing through the hidden layers. A dropout regularization technique is applied between the RGCN layers. The model is used with the CrossEntropyLoss, which implicitly includes a Softmax activation on the output.

BotRGCN Architecture		
Input Layers	Description Embedding	Linear ($\mathbb{R}^{D_s \times \frac{D}{4}}$) + LeakyReLU
	Tweet Embedding	Linear ($\mathbb{R}^{T_s \times \frac{D}{4}}$) + LeakyReLU
	Numerical Properties Embedding	Linear ($\mathbb{R}^{N_s \times \frac{D}{4}}$) + LeakyReLU
	Categorical Properties Embedding	Linear ($\mathbb{R}^{C_s \times \frac{D}{4}}$) + LeakyReLU
Hidden Layers	Input Transformation	Linear ($\mathbb{R}^{D \times D}$) + LeakyReLU
	RGCN 1st Layer	RGCN Convolution ($\mathbb{R}^{D \times D}$)
	RGCN 2nd Layer	RGCN Convolution ($\mathbb{R}^{D \times D}$)
	Hidden Transformation	Linear ($\mathbb{R}^{D \times D}$)+ LeakyReLU
Output Layer	Final Output	Linear ($\mathbb{R}^{D \times 2}$)

2.3 Derived Relations

Elmas et al. [5] argue that a significant challenge in bot detection is the non-intuitive nature of bot characteristics. For instance, their analysis revealed that

[4] https://github.com/BunsenFeng/BotRGCN.

the majority of bot accounts in their dataset had more followers than accounts they were following, and no two bots followed each other.

Moreover, the authors also observed different retweet behaviour for bots, both temporal as well as quantitative. This insight, coupled with the observation of bot evolution, led us to investigate the potential offered by new sets of relations.

Inspired by work from Vargas et al. [12], which builds upon coordination patterns from [9] we introduce the following relations:

- RETWEET: a user retweeted the tweet of another user.
- CO-RETWEET: two users retweeted the same tweet.
- CO-HASHTAG: two users tweet the same hashtag above a certain threshold.

These relations are behavior-based, which makes them harder to manipulate than, e.g., FOLLOWER and FOLLOWING relations. We believe that this approach has the potential to reveal additional patterns of coordinated behavior among users. However, none of these are readily usable for us out-of-the-box and require some data transformation steps.

RETWEET: Our analysis showed that bots tend to retweet disproportionately. In order to take advantage of this, we first need to transform the existing RETWEET relation from tweet→tweet to user→user. By cross-referencing the given RETWEET relation with the POST relation (user→tweet), we are able to associate a user for each tweet and subsequently derive the RETWEET relation in the form of user→user. This process is illustrated in Fig. 1.

CO-RETWEET: We introduce this relation to emphasize instances where two users retweeted the same tweet. To achieve this, we map a user to each tweet that retweets another tweet, similar to the process laid out in RETWEET above. Then, we group these users by their retweeted target tweet. From these groups, we create all possible combinations of users (excluding pairs with the same user twice) and export them as our new CO-RETWEET relation.

CO-HASHTAG: Using a similar grouping and pairing approach as with the CO-RETWEET relation, we focus on the DISCUSS relation (tweet→hashtag). Prior to the pairing step, we filter out hashtags with an unusually large number of users to decrease computational demands and filter out those hashtags that do not offer any reasonable insight. After this step, we create pairs of users who tweeted the same hashtag a minimum of n times. The choice of n can be regarded as a hyperparameter itself and is detailed further, in the subsequent experiments section and Table 3.

3 Experiments

To determine the feasibility of utilizing coordination patterns for bot detection we conducted sensitivity and ablation studies. We kept hyperparameters constant across all experiments. The model is initialized with the same parameters as mentioned in Subsect. 2.2. We further fixed the dropout rate at 0.3, the learning rate at 0.001, and weight decay at 0.005. Furthermore, we standardized the

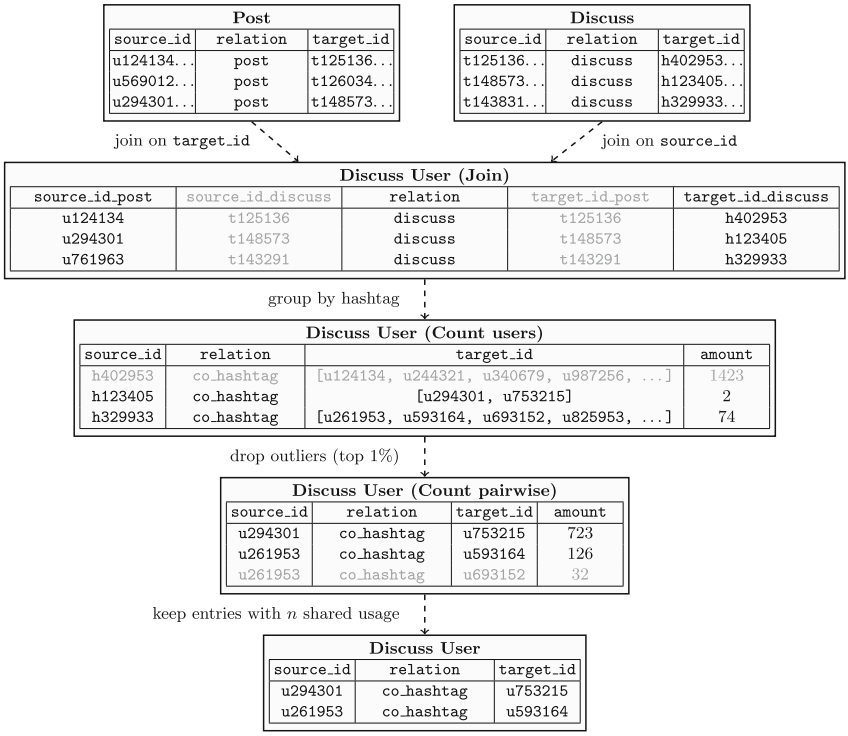

Fig. 1. Visualization of the process of deriving the new Co-Hashtag (`co_hashtag`) relation. Initially, the edge file is split into individual relations (not depicted). We then join the `post` and `discuss` relation to associate user-ids with each hashtag in the `discuss` table. In this example we assume a threshold `amount` value of 100, below which `co_hashtag` occurrences are discarded. We then create pairs of users with the respective count of how often they share a hashtag. Lastly, we keep only those with at least n shared hashtags and discard the `amount` column to get the expected format.

number of training epochs to 200 across all experimental runs. We reused the train/test split that comes with `TwiBot-22`, for comparability with prior work.

First, we defined a threshold for the Co-Hashtag relation. The threshold was set to three standard deviations above the mean, with values provided in Table 3. Since the differences between the thresholds were minor, we chose the one that achieved the highest F1-score, indicating the most reliable predictions. Additional experimentation with the sets and quantities of relations can be referenced in Table 4. Notably, the follower relation yielded the best results, as opposed to the common follower+following combination. It matches the intuition that this relation can be a strong indicator. Our main interest, however, was on the newly derived behavioral relations, with *follow* relationships serving as a baseline for comparison.

Table 3. Sensitivity study of the `co-hashtag` edge creation threshold. The Amount column corresponds to the parameter n, representing the minimum number of times pairs of users tweeted the same hashtag. We run each experiment five times and report the average value as well as the standard deviation in parentheses.

Threshold	Amount	Accuracy	F1-score
mean + 1 SD	467	76.02 (0.65)	42.64 (3.53)
mean + 2 SD	907	75.59 (0.23)	41.03 (1.85)
mean + 3 SD	1347	75.91 (0.39)	43.06 (2.15)
mean + 4 SD	1787	75.36 (0.60)	39.76 (3.52)
mean + 5 SD	2226	75.61 (0.45)	41.07 (2.19)

Our findings necessitate contextual interpretation, contrasting our approach with the conventional use of FOLLOWER+FOLLOWING relations. Instead, we leverage the higher-order CO-RETWEETED and CO-HASHTAG relations to capture more complex user behaviors like mutual affinity for retweeting particular content or using the same hashtags above a certain level. However, we do not dismiss the RETWEET relation and still consider it valuable for future exploration. Though we did not outperform the conventional approach, our results are closely competitive, with differences of less than 1.22 percent points lower in accuracy and 3.78 percent points in F1-score.

Table 4. Sensitivity analysis of `BotRGCN` to different edge types in the graph. We run each experiment five times and report the average value as well as the standard deviation in parentheses.

Category	Sensitivity Settings	Accuracy	F1-score
=single relation type	`follower`	77.63 (0.47)	50.70 (2.03)
	`following`	75.38 (0.59)	37.19 (3.76)
	`retweeted`	75.78 (0.69)	41.75 (4.20)
	`co-retweeted`	75.56 (0.80)	40.43 (4.63)
	`co-hashtag`	75.91 (0.39)	43.06 (2.15)
=two relation types	`follower+following`	76.99 (0.43)	46.06 (2.31)
	`retweeted+co-retweeted`	75.43 (0.34)	39.70 (2.23)
	`co-retweeted+co-hashtag`	75.77 (0.13)	42.28 (1.08)
=three relation types	`following+follower+retweeted`	77.55 (0.57)	48.92 (3.24)
	`retweeted+co-retweeted+co-hashtag`	75.81 (0.52)	41.51 (2.42)
five relation types	all of the above	77.11 (0.32)	46.72 (1.63)

This gap, although initially discouraging, reveals upon closer examination the capability to make predictions, avoiding biases that might have characterized previous approaches. Despite the notable perfomance of the single `follower`

relation, there's evident improvement when using three or five relations instead of two. Our concerns regarding these biases are outlined in Subsect. 2.1 dedicated to the dataset. This highlights the potential of a multi-rational approach, but it is essential to note that inherent characteristics of the used dataset might influence these observations. Such results are particularly significant, as bot developers may find it challenging to avoid behavior-based detection without substantially constraining their capabilities. Building on the findings from Feng et al. [7], where it was confirmed that the optimal performance is achieved with 2 layers of RGCN, we have carried out an ablation study of BotRGCN, utilizing the same layer configuration. Our experiments, as detailed in Table 5, prove that the integration of all available modalities remains essential for robust bot detectors. The challenge requires a multi-faceted approach, integrating various modalities. This approach must then model the aggregation of these signals, aiming to ensure a clear distinction between accounts involved in automated coordinated efforts and those demonstrating authentic behavior, which may stem from social initiatives.

Table 5. Ablation Study of `BotRGCN` under different relation types using 2 layers of `RGCN`. Abbreviations used: T = User Tweets; N = User Numerical Properties; C = User Categorical Properties; D = User Descriptions. We run each experiment five times and report the average value as well as the standard deviation in parentheses.

	`follower + following`		`co-retweeted + co-hashtag`	
Ablation Setting	Accuracy	F1-score	Accuracy	F1-score
`RGCN` + T	70.51 (0.01)	1.34 (0.23)	70.54 (0.02)	0.89 (0.34)
`RGCN` + T, N	70.83 (0.18)	7.05 (2.69)	70.81 (0.28)	6.72 (3.90)
`RGCN` + T, N, C	73.07 (0.34)	25.67 (3.18)	72.70 (0.34)	20.79 (3.02)
`RGCN` + T, N, C, D (`BotRGCN`)	76.99 (0.43)	46.06 (2.31)	75.77 (0.13)	42.28 (1.08)

4 Conclusion

The complexity of bots continues to evolve, making the task of bot detection a critical challenge. Our investigation into alternative higher-order, behavioral-based relations emphasizes a different approach in detecting automated coordinated group activities. Although not surpassing the conventional approach, the competitiveness of our results suggest a reliable method without falling into suspected biases of traditional techniques. Bot developers seeking to avoid detection may find it increasingly difficult without limiting their capacities. `TwiBot-22`, the dataset used in this study, has been instrumental in establishing these new relations. Yet, as we look into further research, the incorporation of temporal patterns into these newly established relations seems promising. This direction, however, necessitates datasets that support this, a limitation we currently face. We are optimistic that pursuits into this direction can foster the development of more robust and reliable detection methods.

Acknowledgements. The authors thank Ali Alhosseini for his guidance during the early conceptual phase and Lukas Drews for his collaboration in the initial experiments.

References

1. Cinelli, M., Cresci, S., Quattrociocchi, W., Tesconi, M., Zola, P.: Coordinated inauthentic behavior and information spreading on twitter. Decision Support Syst. **160**, 113,819 (2022)
2. Cresci, S.: A decade of social bot detection. Commun. ACM **63**(10), 72–83 (2020)
3. Cresci, S., Di Pietro, R., Petrocchi, M., Spognardi, A., Tesconi, M.: 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, pp. 963–972 (2017)
4. Cresci, S., Lillo, F., Regoli, D., Tardelli, S., Tesconi, M.: Cashtag piggybacking: uncovering spam and bot activity in stock microblogs on twitter. ACM Trans. Web (TWEB) **13**(2), 1–27 (2019)
5. Elmas, T., Overdorf, R., Aberer, K.: Characterizing retweet bots: The case of black market accounts. In: Proceedings of the International AAAI Conference on Web and Social Media, vol. 16, pp. 171–182 (2022)
6. Feng, S., Tan, Z., Wan, H., Wang, N., Chen, Z., Zhang, B., Zheng, Q., Zhang, W., Lei, Z., Yang, S., et al.: Twibot-22: towards graph-based twitter bot detection. Adv. Neural. Inf. Process. Syst. **35**, 35254–35269 (2022)
7. Feng, S., Wan, H., Wang, N., Luo, M.: Botrgcn: Twitter bot detection with relational graph convolutional networks. In: Proceedings of the 2021 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, pp. 236–239 (2021)
8. Hays, C., Schutzman, Z., Raghavan, M., Walk, E., Zimmer, P.: Simplistic collection and labeling practices limit the utility of benchmark datasets for twitter bot detection. In: Proceedings of the ACM Web Conference 2023, pp. 3660–3669 (2023)
9. Keller, F.B., Schoch, D., Stier, S., Yang, J.: Political astroturfing on twitter: how to coordinate a disinformation campaign. Polit. Commun. **37**(2), 256–280 (2020)
10. Martini, F., Samula, P., Keller, T.R., Klinger, U.: Bot, or not? comparing three methods for detecting social bots in five political discourses. Big data & society **8**(2), 20539517211033,566 (2021)
11. Rauchfleisch, A., Kaiser, J.: The false positive problem of automatic bot detection in social science research. PloS one **15**(10), e0241,045 (2020)
12. Vargas, L., Emami, P., Traynor, P.: On the detection of disinformation campaign activity with network analysis. In: Proceedings of the 2020 ACM SIGSAC Conference on Cloud Computing Security Workshop, pp. 133–146 (2020)
13. Varol, O., Ferrara, E., Davis, C., Menczer, F., Flammini, A.: Online human-bot interactions: Detection, estimation, and characterization. In: Proceedings of the International AAAI Conference on Web and social media, vol. 11, pp. 280–289 (2017)