Estimating the Bot Population on Twitter via Random Walk Based Sampling

MEI FUKUDA®, KAZUKI NAKAJIMA®, AND KAZUYUKI SHUDO®, (Senior Member, IEEE)
Department of Mathematical and Computing Science, Tokyo Institute of Technology, Tokyo 152-8552, Japan
Corresponding author: Mei Fukuda (fukuda.m.ai@m.titech.ac.jp)
The work of Kazuki Nakajima was supported by JSPS KAKENHI Grant Number JP21J10415.

ABSTRACT The rise of social bots, which contribute to marketing, political intervention, and the spread of fake news, has been noted. Analysis methods for the characteristics of Twitter bots have been developed for third-party researchers who have access limitations to Twitter data. Here, we propose a method for estimating the bot population on Twitter based on a random walk. The proposed method addresses two major problems in estimating the bot population on Twitter based on a random walk. First, the maximum number of retrievable friends or followers of a user per query is limited. Second, there is a certain percentage of private users who do not publish personal content, e.g., friends, followers, and tweets. We conduct a simulation analysis using directed social graph datasets to validate whether the proposed estimator is effective on the real Twitter follow graph. Then, we present three different estimates of the bot population on Twitter using the proposed estimator based on the three sample sequences of 25,000 users collected in 2.5 weeks each. The three estimates consistently suggest that 8%–18% of Twitter users during April–June 2021 are bots.

INDEX TERMS Bot population, estimation, online social networks, random walk, sampling, social bots, Twitter.

I. INTRODUCTION
The rise of social bots (i.e., accounts that automatically generate content) has been noted in recent years [1]. Social bots are created for a variety of purposes, including marketing [2], engaging in political intervention [3]–[5], and spreading fake news [6]–[8]. Many studies (e.g., [9]–[12]) have investigated the characteristics of bots on Twitter, which is a popular social networking service with hundreds of millions of daily active users. The bot population (i.e., the percentage of bot users) on Twitter is one of the basic properties for understanding the impact of bots on individuals’ opinions and behaviors [9], [11]. Twitter officially reported the percentage of bot users on Twitter in 2014 [13]. Nevertheless, it is crucial to allow third-party researchers who are not data holders to independently estimate the bot population on a given platform [6].

Estimating the bot population on Twitter is challenging for third-party researchers because they have limited access to the data. Uniform independent sampling generally yields unbiased estimates. However, it is practically difficult to sample user IDs uniformly at random because Twitter user IDs are sparsely distributed over the 64-bit integer space as of December 2021 [14]. One approach to sample Twitter users is to stream recent public tweets through application programming interfaces (APIs) and collect users who posted the tweets [9], [11]. However, this method introduces the sampling bias toward users who actively post tweets. It is typically necessary to sample a large number of users to correct the sampling bias (e.g., hundreds of thousands to tens of millions of users [9], [11]). Another method for sampling Twitter users is to perform a crawling method (e.g., breadth-first search and random walk) on the follow graph (i.e., a directed graph consisting of a set of nodes representing users and a set of directed edges representing follow relationships) [15]–[17].

Unbiased estimators using a random walk for the percentage of nodes with a specific label on a social network have been studied [18]–[20]. In general, online social networks provide APIs to retrieve neighboring users of a queried user [15], [18], [21], [22]. By repeatedly utilizing this function for a randomly selected neighbor, one obtains a sequence of sampled users via a random walk on the network. Then, one obtains the unbiased estimate by re-weighting each sampled user to correct the sampling bias derived from the Markov chain analysis. As of December 2021, Twitter provides APIs that allow retrieval of the friends (i.e., users...
who are followed by a user) and followers (i.e., users who follow a user) of a queried user [23], [24]. Therefore, one can attempt to estimate the bot population on Twitter using the existing estimators in which the label of the node is set to a binary label indicating whether the user is a bot or not. However, the existing estimators do not address two major problems in estimating the bot population on Twitter. First, the maximum number of retrievable friends or followers per query is restricted to 5,000 as of December 2021. Second, there is a certain percentage of private users who do not publish personal content, e.g., friends, followers, and tweets, if they are queried.

In the present study, we propose an estimator for the bot population on Twitter based on a random walk. In each step of our random walk, we perform only one query each to retrieve friends and followers of a sampled user and then move to a node randomly selected from the retrieved friends and followers. We theoretically show that the proposed estimator has approximately no bias caused by private users when all friends and followers of a user are retrievable using a single query each. Then, we conduct simulation analysis using datasets of directed social graphs to provide numerical and empirical evidence that the proposed estimator is effective for the real Twitter follow graph. Finally, we present three different estimates for the bot population on Twitter during April–June 2021. We obtained the three sample sequences of 25,000 users by performing our random walk on Twitter for 2.5 weeks each. The three estimates consistently suggest that 8%–18% of Twitter users are bots, which corresponds with existing studies. Our code for estimating the bot population on Twitter is available at https://github.com/meipipo/twitter-bot-population-estimator.

II. RELATED WORK

A. CHARACTERIZATION STUDIES OF TWITTER BOTS

The impact of Twitter bots on individuals’ opinions and behaviors has been actively studied. There have been many studies on the characteristics of activities of social bots during the U.S. presidential election [3]–[5]. For example, right-leaning bots noticeably shifted their tweets to topics encouraging people to vote for Republicans before and during the Democratic National Convention, even though there was no significant difference in the content of tweets from human accounts [4]. It has also been reported that Twitter bots systematically spread fake news and amplify conspiratorial information [6]–[8]. The role of Twitter bots during COVID-19 has also been studied recently. Yang et al. reported that bots tweet links to unreliable sources more than their proportion in overall tweets [25]. Because misinformation spreads faster than information from reliable sources, tweets from social bots are more likely to be retweeted, which can inhibit the spread of correct information and discussion [26].

Several studies have been conducted on classifying bot and non-bot accounts. Chu et al. analyzed 500,000 Twitter accounts to classify human, bot, and cyborg (between human and bot) accounts [10]. Stieglitz et al. found differences in characteristics between bots and humans, such as the number of followers, the number of retweets per day, and the number of links to web pages used per day [27]. In DARPA’s Twitter bot detection competition, several teams developed techniques for detecting malicious organizational activity using machine learning [28]. A software called Botometer [29], [30] enables accurate classification of specified Twitter users as automated bots or humans through APIs. Botometers have been frequently used in previous studies to investigate Twitter bots [7], [8], [12]. In this study, we use a Botometer to determine whether each sampled user is a bot for estimating the bot population on Twitter.

Several studies estimated the bot population on Twitter. Varol et al. suggested that up to 15% of active Twitter accounts were bots in 2015 [9]. Luceri et al. examined the period during the 2016 and 2018 elections and classified 12.6% of accounts as bots [11]. These previous studies used hundreds of thousands to tens of millions of users sampled over several months. In contrast, the proposed estimator realizes a reasonable estimate of the bot population using samples obtained over just 2.5 weeks. Our estimate is also consistent with the value of 8.5%, which is the percentage of bots that Twitter officially reported in 2014 [13].

B. RANDOM WALK BASED GRAPH SAMPLING

A random walk is an effective sampling technique for online social networks where neighboring users of a queried user are retrievable using APIs [18], [19], [31]. Ribeiro et al. proposed a sampling algorithm based on a random walk, called DURW, for directed graphs in which only the outgoing edges of a queried node are retrievable [31]. The DURW algorithm has been used for sampling the Twitter retweet graph [32]. For the Twitter follow graph as of December 2021, we are allowed to retrieve both the incoming and outgoing edges of a queried user by calling the endpoints GET friends/ids and GET followers/ids [23], [24]. Therefore, an estimation of the bot population on the Twitter follow graph is reduced to the problem of estimating the percentage of nodes with a specific label on an undirected social graph [18]–[20].

The existing estimators for a percentage of nodes with a specific label assume that all neighbors of a queried node are retrievable via a single query [18]–[20]. However, the maximum number of retrievable friends or followers per query is limited to 5,000 on Twitter as of December 2021 [23], [24]. In the present study, we propose a sampling algorithm based on a random walk in which we perform just one query each to retrieve friends and followers of a sampled user and then move to a node randomly selected from retrieved friends and followers. We numerically compare the proposed random walk with the existing random walk in which one queries each sampled node until one retrieves all of the friends and followers of the node. Our experimental results show that the proposed algorithm improves the estimation accuracy using the same number of queries as the existing algorithm.

The existing estimators [18]–[20] also do not address private users on Twitter, who do not publish personal content,
e.g., friends, followers, and tweets, if they are queried. There is typically a certain percentage of private users in online social networks (e.g., 27% on the Facebook network [33] and 34% on the Pokec network [34]). Although Twitter does not officially report the percentage of private users as of December 2021, we empirically found that there are a certain number of private users on Twitter. Private users prevent us from performing a simple random walk (i.e., one repeatedly traverses a randomly chosen neighbor), which causes the bias of estimators [35]. Nakajima and Shudo recently designed a framework for estimators based on a random walk considering private nodes for undirected social graphs [35]. In the present study, we extend the framework [35] to the case of the estimation of a percentage of private nodes, who do not publish personal content, e.g., friends, followers, and tweets if they are queried. Each node independently becomes private with probability \( p \), otherwise public, where \( 0 \leq p < 1 \). We assume that private nodes are uniformly distributed over the social graph. Nakajima and Shudo validated this assumption for social graphs [35].

III. PRELIMINARIES

A. NOTATIONS

We represent the Twitter follow graph as a directed graph \( G = (V, E) \), where \( V = \{v_1, v_2, \ldots, v_n\} \) is a set of nodes (users) and \( E \) is a set of directed edges (follow relationships). We denote by \( n \) the number of nodes. In the Twitter terminology, if a directed edge \((v_i, v_j)\) exists, node \( v_j \) is called a friend of \( v_i \), and node \( v_i \) is called a follower of \( v_j \) [23], [24]. The graph \( G \) has no multiple directed edges or self-loops. We denote by \( d^\text{in}_i \) and \( d^\text{out}_i \) the in-degree (i.e., the number of followers) and the out-degree (i.e., the number of friends), respectively, of node \( v_i \). We define \( d_{\text{max}} \) as the larger of the maximum out-degree of the node and the maximum in-degree of the node. Each node \( v_i \) has a bot label \( l\text{bot}(i) \in \{\text{bot, non-bot}\} \). We denote by \( 1_{\text{cond}} \) an indicator function that returns 1 if the condition \( \text{cond} \) holds and 0 otherwise. We define the proportion of bots as

\[
p_{\text{bot}} = \frac{1}{n} \sum_{v_i \in V} 1_{l\text{bot}(i)=\text{bot}}.
\]

As in the previous study [35], we introduce notations and definitions with respect to private nodes, who do not publish personal content, e.g., friends, followers, and tweets if they are queried. Each node \( v_i \) also has a privacy label \( l\text{priv}(i) \in \{\text{private, public}\} \). Suppose that we query a node \( v_i \). If \( v_i \) is a public node, we are allowed to retrieve friends, followers, and tweets of \( v_i \); otherwise, we are not. We refer to connected subgraphs consisting of public nodes in \( G \) as public clusters. We denote by \( C^* = (V^*, E^*) \) the largest public cluster. For example, for a directed graph shown in Fig. 1, there are three public clusters.

- \( C_1 = \{(1, 2, 3, 4, 5), (1, 2, 3, 4), (1, 5, 6, 7)\}\)
- \( C_2 = \{(8, 9), (9, 8)\}\)
- \( C_3 = \{(7)\}\)

The largest public cluster is \( C^* = C_1 \).

B. ASSUMPTIONS

We design a method for estimating the bot population \( p_{\text{bot}} \) under the following assumptions.

1. \( G \) is weakly connected. Previous studies on undirected social graphs similarly assumed that the original undirected graph is connected [18], [20]. There is also empirical evidence that the Twitter follow graph is mostly weakly connected, e.g., 99.94% of all nodes with at least one friend or follower belonged to the largest weakly connected component of the Twitter follow graph as of 2012 [36].

2. \( G \) remains static during the duration of our random walk. Previous studies on undirected social graphs similarly assumed that the original graph is static during the sampling of the graph data [18], [19]. Gjoka et al. verified that this assumption sufficiently holds for the Facebook graph [18].

3. The indices of all friends and followers of a queried public node are retrievable. For example, if node 8 is queried on a graph shown in Fig. 1, we are allowed to retrieve its friends, i.e., node 6 and node 9, and its follower, i.e., node 9. We confirmed that the Twitter APIs satisfy this assumption as of December 2021 [23], [24].

4. Each node independently becomes private with probability \( p \), otherwise public, where \( 0 \leq p < 1 \). We assume that private nodes are uniformly distributed over the social graph. Nakajima and Shudo validated this assumption for social graphs [35].

5. The initial node of our random walk is located in the largest public cluster of \( G \). We do not consider

FIGURE 1. A directed graph with privacy labels (left figure) and lists of retrievable friends and followers for each node (right table). Node 6 is a private node, and the others are public nodes.

We denote by \( k \) the number of retrievable friends or followers per query. We assume that we sequentially retrieve groups of \( k \) friends or followers of a node from the back of the list of friends or followers using a single query. For example, we consider the case of \( k = 100 \). If we want to obtain all 520 friends and 450 followers of a node, we need to perform the query \( \lceil \frac{520}{100} \rceil + \lceil \frac{450}{100} \rceil = 11 \) times. Note that the case of \( k = d_{\text{max}} \) (i.e., all friends or followers are available using one query) is identical to the assumption adopted in previous studies [18]–[20], [35].
First, we select an initial node \( G \) public cluster of \( k \) walk when examples of retrievable and traversable nodes in our random analysis is an extension of the theoretical analysis for directed social graphs presented in [35]. Unless otherwise stated, we assume that \( k = d_{\text{max}} \) in this section.

First, we introduce some notations. We denote by \( \hat{G} = (V, \hat{E}) \) the undirected graph generated by removing the directions of the edges in the original directed graph \( G \), where \( \hat{E} \) is the set of edges in the undirected graph. An edge \((v_i, v_j)\) exists in \( \hat{G} \) if and only if at least one directed edge between \( v_i \) and \( v_j \) exists in \( G \). We refer to nodes connected to node \( v_i \) on \( \hat{G} \) as the neighbors of \( v_i \). We denote the degree (i.e., the number of neighbors) of node \( v_i \) in \( \hat{G} \) by \( \hat{d}_i \). Let \( \hat{C}^* = (V^*, \hat{E}^*) \) denote the largest public cluster of \( \hat{G} \). We refer to public nodes connected to node \( v_i \) as the public neighbors of \( v_i \). Let \( \hat{d}_i^p \) denote the public degree (i.e., the number of public neighbors) of node \( v_i \) in \( V^* \). Let \( \hat{D}^p = \sum_{v_i \in V^*} \hat{d}_i^p \) denote the sum of public degrees.

For example, suppose that graph \( G \) is the directed graph shown in Fig. 1. One obtains \( \hat{E} = \{(1, 2), (1, 5), (2, 3), \ldots \} \) and the friends and followers of \( v_i \). We start our random walk from node 1, we query friends and followers of each node on the graph presented in Fig. 1 when \( k = 1 \). When we start our random walk from node 1, we query friends and followers of node 1 once each, and we obtain \( \Gamma_{k=1}^{\text{out}}(1) = \{5\} \) and \( \Gamma_{k=1}^{\text{in}}(1) = \{5\} \). Then, we move to node 5 and query friends and followers of node 5 once each and obtain \( \Gamma_{k=1}^{\text{out}}(5) = \{6\} \) and \( \Gamma_{k=1}^{\text{in}}(5) = \{6\} \). However, the algorithm cannot proceed anymore since node 6 is a private node. To address this problem, we include node 1, which was sampled in the step before node 5, in the set \( \Gamma_k \). This operation ensures that the set \( \Gamma_k \) contains at least one public node if \( i \geq 2 \). If the initial node \( v_{x_i} \) is a node such that \( \Gamma_k(v_{x_i}) \) consists of only private nodes such as node 5, we reselect the initial node from the largest public cluster of \( G \). However, unless \( k \) is too small, it is rare to choose such nodes (see Section VI-A and Fig. 5 for details).

In general, when \( k \) (i.e., the number of friends or followers retrievable via a single query) is less than \( d_{\text{max}} \), the number of public nodes reachable via our random walk depends on the initial node \( v_{x_i} \). For example, we consider the case of \( k = 1 \) in the directed graph shown in Fig. 1. Fig. 2(a) shows the lists of retrievable friends and followers of each node in this case. When we select node 1 as the initial node in Fig. 2(a), we have \( \Gamma_{k=1}^{\text{out}}(1) = \{5\} \) and \( \Gamma_{k=1}^{\text{in}}(1) = \{5\} \). Therefore, the set of reachable public nodes is \{1, 5\} (see Fig. 2(b)). When we select node 2 as the initial node in Fig. 2(a), the set of reachable public nodes is \{1, 2, 4, 5\} (see Fig. 2(c)). However, we empirically find that unless \( k \) is too small compared with \( d_{\text{max}} \), the variance in the number of reachable public nodes depending on the initial node is quite small, and most public nodes on the largest public cluster are reachable (see Section VI-A and Fig. 5 for details).

V. THEORETICAL ANALYSIS
We theoretically analyze the bias of the proposed estimator caused by private nodes when \( k = d_{\text{max}} \). Our theoretical analysis is an extension of the theoretical analysis for undirected social graphs presented in [35].
First, we calculate the expectation of the proposed estimator \( \hat{\pi}_v \) as in [35] on the largest public cluster, \( v \), of \( G \) because all the friends and followers of \( v \) are retrievable by querying \( v \) just once when \( k = d_{\text{max}} \). Therefore, in our random walk, one first chooses an initial node \( v_i \) that belongs to the largest public cluster of \( G \) and then repeatedly moves to a randomly chosen public neighbor from the set of neighbors of \( v_i \). This is equivalent to performing the random walk proposed in [35] on \( G \).

Lemma 1 allows us to apply the theoretical results shown in [35] to the undirected graph \( G \). Let the probability that event \( A \) will occur be denoted by \( \Pr[A] \). We define the distribution induced by the sequence of \( r \) indices of sampled nodes as \( \pi_r = (\Pr[x_r = i])_{i \in V} \). According to Lemma 3.1 in [35], each node in the largest public cluster, \( v_i \) in \( V^* \), is sampled from the following stationary distribution.

**Lemma 2:** The vector \( \pi_r \) converges to the stationary distribution \( \pi = (\pi_i)_{i \in V} \) after many steps of our random walk on \( G \), where \( \pi_i = \frac{1}{|V_i \cap V^*|} \frac{\bar{d}^*_i}{\bar{d}_i} \).

Then, we obtain the following lemma regarding the convergence value of the proposed estimator \( \hat{\pi}_v \).

**Lemma 3:** The proposed estimator \( \hat{\pi}_v \) converges to

\[
\hat{\pi}_v = \frac{\sum_{v_i \in V^*} \frac{1}{|\Gamma_{\text{max}}(v_i)|} \bar{d}^*_i}{\sum_{v_i \in V} \bar{d}^*_i / \bar{d}_i}
\]

after many steps of our random walk on \( G \).

**Proof:** We define the quantity \( \Phi_{\text{bot}} \) as

\[
\Phi_{\text{bot}} = \frac{1}{r} \sum_{i=1}^{r} \frac{1}{|\Gamma_{\text{max}}(x_i)|}.
\]

First, we calculate the expectation of \( \Phi_{\text{bot}} \) with respect to the stationary distribution \( \pi \):

\[
\mathbb{E}_\pi[\Phi_{\text{bot}}] = \mathbb{E}_\pi[1_{|\Gamma_{\text{max}}(x_i)|} / |\Gamma_{\text{max}}(x_i)|] = \frac{1}{D^*} \sum_{v_i \in V^*} \frac{1}{|\Gamma_{\text{max}}(v_i)|} \frac{\bar{d}^*_i}{\bar{d}_i}.
\]

The first equation holds because of the linearity of the expectation. The second equation holds because \( |\Gamma_{\text{max}}(i)| \) = \( \bar{d}_i \) holds for each node \( v_i \) and Lemma 2 holds.

Then, we define the quantity \( \Psi_{\text{bot}} \) as

\[
\Psi_{\text{bot}} = \frac{1}{r} \sum_{i=1}^{r} \frac{1}{|\Gamma_{\text{max}}(x_i)|}.
\]

We similarly obtain the expectation of \( \Psi_{\text{bot}} \) with respect to \( \pi \) as

\[
\mathbb{E}_\pi[\Psi_{\text{bot}}] = \mathbb{E}_\pi[1/|\Gamma_{\text{max}}(x_i)|] = \frac{1}{D^*} \sum_{v_i \in V^*} \frac{\bar{d}^*_i}{\bar{d}_i} = \mathbb{E}_\pi[\Phi_{\text{bot}}].
\]

Quantities \( \Phi_{\text{bot}} \) and \( \Psi_{\text{bot}} \) converge to their respective expectations, i.e., (2) and (3), with respect to \( \pi \) after many steps of our random walk because of the strong law of large numbers for a Markov chain [35]. Therefore, we conclude that \( \hat{\pi}_v \) converges to \( \bar{\pi}_v \).

The following proposition holds because \( C^* \) is equivalent to \( G \) when there are no private nodes in \( G \).

**Proposition 1:** When there are no private nodes in \( G \), the convergence value \( \hat{\pi}_v \) of the proposed estimator is equal to the true value \( \bar{\pi}_v \).

We show that private nodes cause approximately no bias between the convergence value \( \hat{\pi}_v \) of the proposed estimator and the true value \( \bar{\pi}_v \). To this end, we calculate the expectation of \( \hat{\pi}_v \) with respect to the set of privacy labels of nodes. We denoted by \( L_{\text{pri}} = \{l_{\text{pri}}(i) \}_{i \in V} \) the set of privacy labels of nodes. Let \( \mathbb{E}_\pi[X] \) denote the expectation of a random variable \( X \) with respect to the set \( L_{\text{pri}} \). We approximate \( \mathbb{E}_\pi[\hat{\pi}_v] \) under the condition that all public nodes belong to the largest public cluster of \( G \). Under this condition, it holds that \( \Pr[v_i \in V^*] = \Pr[l_{\text{pri}}(i) = \text{public}] = 1 - p \) because of Assumption 4. We also approximate \( \mathbb{E}_\pi[\hat{\pi}_v] \) as a fraction of the expectation of each quantity in the denominator and numerator of \( \hat{\pi}_v \).

**Theorem 1:** Under the condition that all public nodes belong to the largest public cluster, we have

\[
\mathbb{E}_\pi[\hat{\pi}_v] \approx \frac{\mathbb{E}_\pi[\sum_{v_i \in V^*} 1_{l_{\text{pri}}(i) = \text{public}} \bar{d}^*_i / \bar{d}_i]}{\mathbb{E}_\pi[\sum_{v_i \in V^*} \bar{d}^*_i / \bar{d}_i]} = \bar{\pi}_v.
\]

**Proof:** We define a random variable \( X_{\text{bot}} = \sum_{v_i \in V^*} 1_{l_{\text{bot}}(i) = \text{bot}} \bar{d}^*_i / \bar{d}_i \). The expectation of \( X_{\text{bot}} \) with respect to \( L_{\text{bot}} \) is given by

\[
\mathbb{E}_\pi[X_{\text{bot}}] = \sum_{v_i \in V} \Pr[v_i \in V^*] \mathbb{E}_\pi[1_{l_{\text{bot}}(i) = \text{bot}} \bar{d}^*_i / \bar{d}_i] = (1 - p)^2 \sum_{v_i \in V} 1_{l_{\text{bot}}(i) = \text{bot}}.
\]

Equation (4) holds because of the linearity of the expectation and the law of total expectations. Equation (5) holds because \( \mathbb{E}_\pi[\bar{d}^*_i] = (1 - p) \bar{d}_i \) holds (see Lemma 4.5 in [35] for the proof). We note that \( 1_{l_{\text{bot}}(i) = \text{bot}} \) and the degree \( \bar{d}_i \) are constants with respect to the set \( L_{\text{bot}} \) for each node \( v_i \). Then, we define \( Y_{\text{bot}} = \sum_{v_i \in V^*} \bar{d}^*_i / \bar{d}_i \). The expectation of \( Y_{\text{bot}} \) with respect to \( L_{\text{bot}} \) is given by \( \mathbb{E}_\pi[Y_{\text{bot}}] = (1 - p)^2 n \). See
TABLE 1. Datasets.

| Data      | |V| | |E| |dmax |
|-----------|------------|-----------|-----------------------------|
| ego-Twitter [37], [38] | 81,306 | 1,768,135 | 3,383 |
| YouTube [21], [39] | 1,134,885 | 4,938,950 | 28,564 |
| Higgs [37], [39] | 456,290 | 14,855,443 | 51,386 |
| Flickr [38], [39] | 2,173,370 | 32,948,343 | 26,367 |

Theorem 4.12 in [35] for the proof. Finally, we obtain the desired result, i.e., \( \mathbb{E}_{\text{pri}}[X_{\text{bot}}] / \mathbb{E}_{\text{pri}}[Y_{\text{bot}}] = p_{\text{bot}} \).

Theorem 1 implies that the expectation of \( p_{\text{bot}} \) with respect to the set \( L_{\text{pri}} \) is almost equal to the true value \( p_{\text{bot}} \) when \( k = d_{\text{max}} \). Through the following simulation analysis, we also find that the proposed estimator realizes an accurate estimate of \( p_{\text{bot}} \) independent of the proportion of private nodes unless \( k \) is too small compared with \( d_{\text{max}} \).

VI. SIMULATION ANALYSIS

In this section, we conduct numerical simulations using four directed social graph datasets to validate whether the proposed estimator is effective on the real Twitter follow graph. We use the largest weakly connected component of the original directed graph for each dataset. Table 1 shows the number of nodes, the number of edges, and the larger of the maximum in-degree of the node and the maximum out-degree of the node for each directed graph used in our simulations.

A. PERFORMANCE OF THE PROPOSED ESTIMATOR

We examine how the accuracy of the proposed estimator varies with the given number of retrievable friends or followers per query, i.e., \( k \). We could not find publicly available datasets of directed social graphs including real bot labels of users. Therefore, we consider three methods of synthetically labeling nodes as bots on the graph: Random, LowDegree, and HighDegree. In the Random labeling method, we label node \( v_i \) as a bot with uniform probability, i.e., \( 1/n \). In the LowDegree labeling method, we label node \( v_i \) as a bot with probability \((1/d_i)/\sum_{v_j\in V}(1/d_j)\), where we define \( d_i = \max(d_i^{\text{in}}, d_i^{\text{out}}) \). In the HighDegree labeling method, we label node \( v_i \) as a bot with probability \( \tilde{d}_i / \sum_{v_j\in V} \tilde{d}_j \). In each method, we repeatedly label a node chosen from the corresponding probability until the percentage of labeled nodes reaches a given value.

Fig. 3(a)–(d) shows the complementary cumulative distribution function (CCDF) of the larger of in-degree and out-degree \( \tilde{d} \) of all the nodes and 10% nodes labeled by each labeling method for each dataset. The degree \( \tilde{d} \) of nodes labeled by the Random labeling method follows a similar distribution of degree \( \tilde{d} \) of all nodes, as expected. The degree \( \tilde{d} \) of nodes labeled by the LowDegree and HighDegree labeling methods are biased towards low and high degrees, respectively, as expected.

We run a single simulation as follows. First, we label 10% of all nodes by using one of the three methods (i.e., Random, LowDegree, or HighDegree). Second, we set the privacy label of each node to private with the given probability \( p \) or public otherwise, according to Assumption 4. Third, we randomly shuffle the orders of the lists of friends and followers of each node independently. Fourth, we select the initial node for our random walk from the largest public cluster uniformly at random. Fifth, we perform our random walk with the sample size \( r = 10,000 \) for the given number of retrievable friends or followers per query, i.e., \( k \). We vary \( k \) from 1 to \( d_{\text{max}} \) and the probability \( p \) from 0.0 to 0.3. We use the normalized root mean square error (NRMSE), namely, \( (\mathbb{E}[(p_{\text{bot}} - p_{\text{bot}})^2])^{1/2} \), to measure the accuracy of the proposed estimator and its variance, as in [19], [35]. We calculate the NRMSE of the proposed estimator over 100 independent runs.

Fig. 4 shows the NRMSE of the proposed estimator as a function of \( k \). The following observations apply to all the datasets. First, for all three labeling methods, the proposed estimator achieves almost the same NRMSE when the proportion of private nodes varies from 0.0 to 0.3 when \( k = d_{\text{max}} \), which supports our theoretical result in Section V. We also found that the proposed estimator introduces almost no bias due to private nodes when \( k \geq 10 \) for all the labeling methods. Second, the NRMSE is large for small values of \( k \) (e.g., \( k = 1 \)) for all the labeling methods.

This is because the number of reachable public nodes from the initial node is considerably small when \( k \) is too small, causing bias in the proposed estimator. Third, for all the labeling methods, as \( k \) becomes larger, the NRMSE decreases, and when \( k \) is larger than a certain value, the NRMSE for \( k \) is comparable to that for \( d_{\text{max}} \). The critical value of \( k \) is different for each of the labeling methods. For example,
in the ego-Twitter dataset, $k \approx 5$ for the Random labeling method, $k \approx 100$ for the LowDegree labeling method, and $k \approx 300$ for the HighDegree labeling method. The intuitive reason why the critical value of $k$ is different for each labeling method is as follows. First, when $k < d_{\text{max}}$, the stationary distribution of nodes is typically biased compared to the vector $\pi$ shown in Lemma 2 because a set of edges that can be traversed from the initial node is restricted. However, the degree $\bar{d}$ of the labeled node is uniformly distributed, so the NRMSE rapidly decreases as the value of $k$ increases. In contrast, in the LowDegree labeling method and the HighDegree labeling method, the degree $\bar{d}$ of the labeled node is biased toward low and high degrees, respectively, and hence, the NRMSE relatively slowly decreases as the value of $k$ increases. In particular, the NRMSE is typically high for small values of $k$ in the HighDegree labeling method because a node with a higher degree $\bar{d}$ has a larger bias of the stationary distribution of the node.

Fig. 5 shows the maximum and minimum proportions of reachable public nodes on the largest public cluster depending on the initial node on the ego-Twitter dataset as a function of $k$. When $k = 1$, only 20% of the nodes in the largest public cluster are connected at most. The variance in the proportion is also relatively large when $k < 10$, e.g., the proportion varies from 0.0 to 0.6 when $k = 3$. This small size of the reachable node causes the bias of the estimator using our
random walk. When \( k \) reaches a certain value, for example, when \( k \geq 100 \), most of the nodes are reachable, and the variance becomes small.

**B. COMPARISON WITH THE EXISTING RANDOM WALK**

We compare the NRMSE of the estimator for the bot population defined as equation (1) using our random walk with that using the existing random walk [35] when we use the same number of queries. In the existing random walk, we repeatedly queries each sampled node until all of the friends and followers of the node are retrieved, and then, one moves to a public node randomly chosen from a union set of friends and followers. When we visit node \( v_i \) for the first time, while our random walk uses two queries for the node, the existing random walk uses \( \lceil d_i^\text{in}/k \rceil + \lceil d_i^\text{out}/k \rceil \) queries for the node for the given \( k \).

We run a single simulation on each dataset as follows. First, we synthetically label 10% of all nodes as bots by using one of the three methods (i.e., Random, LowDegree, or HighDegree). Second, we set the privacy label of each node to private with the given probability \( p \) or to public otherwise, according to Assumption 4. We set \( p = 0.2 \) because we confirmed that the proposed estimator introduces almost no bias due to private nodes (see Fig. 4). Third, we randomly shuffle the orders of the lists of friends and followers of each node independently. Fourth, we select an initial node for our random walk and the existing random walk from the largest public cluster uniformly at random. Fifth, we perform our random walk and the existing random walk on the ego-Twitter dataset as a function of \( k \).

Fig. 6 shows the NRMSE of each estimator as a function of the number of queries for each labeling method on the ego-Twitter dataset. The number of queries for the ego-Twitter dataset varies from 1,000 to 5,000 because of the small size of the graph. Fig. 7 shows the NRMSE of each estimator using 20,000 queries for each labeling method on the YouTube, Higgs, and Flicker datasets. The following observations apply to all the datasets. First, for the Random and LowDegree labeling methods, the estimator using our random walk typically achieves a smaller NRMSE than that using the existing random walk when we use the same number of queries in both the \( k \) values. For example, our random walk improves the NRMSE by 77% and 69% on the Flickr dataset in the case of Random and LowDegree methods when \( k = 50 \), respectively. Second, in the HighDegree labeling method, the NRMSE of the estimator using our random walk is larger than that using the existing random walk in both the \( k \) values. This is because our random walk induces the bias of the stationary distribution of the labeled node that has a high degree \( \hat{d} \) when \( k \) is small.

**C. DISCUSSION**

Up to our numerical efforts, we found that the proposed estimator is effective when the following two conditions are satisfied. First, the bot nodes should be either distributed uniformly throughout the graph or biased toward a lower number of friends and followers. Second, for the given number of retrievable friends or followers per query, i.e., \( k \), the proportion of nodes with friends or followers greater than \( k \) (i.e., nodes that satisfy \( \hat{d}_i > k \)) should be sufficiently small.

We consider that the real Twitter follow graph satisfies the two conditions. The empirical evidence is as follows. First, several studies empirically showed that Twitter bots tend to have a small number of friends and followers. Stieglitz et al. found that bots tend to have fewer followers than humans [27]. A study of a group of over 350,000 bots [40] found that both the number of friends and followers of bots were very small, with fewer than 31 friends and fewer than 10 followers. Second, several studies have reported that the percentage of users with more than 5,000 friends or followers is quite small. Note that Twitter APIs restrict the maximum number of retrievable friends or followers per query to 5,000 as of December 2021 [23], [24]. According to a previous study [15], fewer than 0.1% of all users have 5,000 or more friends or followers on Twitter as of 2009. Myers et al. reported that 5% of all users have 470 or more friends or followers on Twitter as of 2012 [36], which suggests that the percentage of users who have 5,000 or more friends or followers is sufficiently less than 5%. Therefore, based on our simulation analysis, we expect that the proposed estimator realizes a reasonable estimation of the bot population on Twitter.

**VII. ESTIMATION ON TWITTER**

**A. METHOD**

We performed our random walk three separate times on the real Twitter follow graph. We selected the initial users as the first author’s account, Jack Dorsey’s account, and Biz Stone’s account. We collected \( r = 25,000 \) sampled users each time. Then, we discarded the first 5,000 sampled users to sufficiently remove the dependency on the initial user, as in the previous case study of random-walk sampling on...
FIGURE 6. NRMSE for our random walk and existing random walk as a function of the number of queries for each labeling method (i.e., Random, LowDegree, or HighDegree) on the ego-Twitter dataset. In (a)–(c), we set the value of $k$ to the smallest value such that $\Pr[\tilde{d} > k] \leq 0.05$ holds. In (d)–(f), we set the value of $k$ to the smallest value such that $\Pr[\tilde{d} > k] \leq 0.01$ holds.

FIGURE 7. NRMSE for our random walk and existing random walk for each labeling method (i.e., Random, LowDegree, or HighDegree) on the YouTube, Higgs and Flickr datasets. We set the number of queries to 20,000. In (a)–(c), we set the value of $k$ to the smallest value such that $\Pr[\tilde{d} > k] \leq 0.05$ holds. In (d)–(f), we set the value of $k$ to the smallest value such that $\Pr[\tilde{d} > k] \leq 0.01$ holds.

the Facebook graph [18]. The periods of data collection were April 13 – 30, May 6 – 23, and May 23 – June 9 in 2021.

We used three functions of the Twitter APIs. We called the endpoints GET friends/ids and GET followers/ids to retrieve friends and followers of a public user [23],[24]. As of December 2021, we are allowed to send such queries up to 15 times in 15 minutes. We identified whether a user was private or public by calling the endpoint GET statuses/show/:id. This endpoint allows 900 calls in 15 minutes as of December 2021 [41].

Twitter does not officially provide APIs to determine whether a given Twitter user is a bot. Botometer v4 [30] to determine whether each sampled public user was a bot. Botometer calculates a score that gives the conditional probability that accounts with an equal or greater score are automated. We labeled users with scores equal to or greater than a given threshold $\theta$ as bots. We labeled users whose scores could not be calculated due to problems such as an insufficient number of tweets as non-bots.

B. RESULTS

Fig. 8 shows estimates calculated from the 1st to $r'$-th sampled users ($1,000 \leq r' \leq 20,000$) for the three sample sequences with the first 5,000 sampled users discarded each.
We found that all three estimates converge to approximately 0.08 when we set $\theta = 0.95$ and to approximately 0.18 when we set $\theta = 0.90$, illustrating the robustness of the proposed estimator with respect to the initial user. Our estimate of between 8% and 18% is sufficiently consistent with Varol et al.’s estimate of 9% to 15% as of 2015 [9] and Luceri et al.’s estimate of 12.6% as of 2019 [11].

We performed the existing random walk [35] independently multiple times for 10 days each in October and November 2021. However, the existing random walk collected only a few dozen sampled users, which suggests that the existing random walk is much less efficient in terms of the number of queries than our random walk. This is because the existing random walk requires a large number of queries to retrieve all the friends or followers of a user with an enormous number of friends or followers. In fact, the existing random walk visited Katy Perry, who has over one hundred million followers, and it took more than 5 days to retrieve all the followers using the Twitter APIs [24]. The existing random walk requires an exorbitant number of queries in total because it tends to frequently traverse users with an enormous number of friends or followers.

\section{VIII. CONCLUSION}

In this study, we proposed an estimator based on a random walk for the bot population on Twitter. We addressed two major problems in estimating the bot population on Twitter: (i) the maximum number of retrievable friends or followers per query is restricted to 5,000, and (ii) a certain percentage of users are private users. We obtained the estimates of the bot population on Twitter between 8% and 18%, which is sufficiently consistent with the estimates reported in previous studies. Our future work includes the improvement of the accuracy of bot detectors (e.g., Botometer), which leads to improving the estimation accuracy of the proposed method.

\section*{REFERENCES}

[1] E. Ferrara, O. Varol, C. Davis, F. Menczer, and A. Flammini, “The rise of social bots,” Commun. ACM, vol. 59, no. 7, pp. 96–104, Jun. 2016.
[2] S. Kim and J. Han, “Detecting engagement bots on social influencer marketing,” in Proc. SocInfo, 2020, pp. 124–136.
[3] N. Grinberg, K. Joseph, L. Friedland, B. Swire-Thompson, and D. Lazer, “Fake news on Twitter during the 2016 U.S. presidential election,” Science, vol. 363, no. 6425, pp. 374–378, Jan. 2019.
[4] E. Ferrara, H. Chang, E. Chen, G. Muric, and J. Patel, “Characterizing social media manipulation in the 2020 U.S. presidential election,” First Monday, vol. 25, no. 11, 2020.
[5] A. Bovet and H. A. Makse, “Influence of fake news in Twitter during the 2016 US presidential election,” Nat. Commun., vol. 10, no. 7, Dec. 2019, Art. no. 7.
[6] D. M. J. Lazer, M. A. Baum, Y. Benkler, A. J. Berinsky, K. M. Greenhill, F. Menczer, M. J. Metzger, B. Nyhan, G. Pennycook, D. Rothschild, M. Schudson, S. A. Sloman, C. R. Sunstein, E. A. Thorson, D. J. Watts, and J. L. Zittrain, “The science of fake news,” Science, vol. 359, no. 6380, pp. 1094–1096, 2018.
[7] C. Shao, G. L. Ciampaglia, O. Varol, K.-C. Yang, A. Flammini, and F. Menczer, “The spread of low-credibility content by social bots,” Nat. Commun., vol. 9, no. 4787, Dec. 2018, Art. no. 4787.
[8] S. Vosoughi, D. Roy, and S. Aral, “The spread of true and false news online,” Science, vol. 359, no. 6380, pp. 1146–1151, Mar. 2018.
[9] O. Varol, E. Ferrara, C. B. Davis, F. Menczer, and A. Flammini, “Online human-bot interactions: Detection, estimation, and characterization,” in Proc. ICWSM, 2017, pp. 280–289.
[10] Z. Chu, S. Gianvecchio, H. Wang, and S. Jajodia, “Who is tweeting on Twitter: Human, bot, or cyborg?” in Proc. ACSAC, 2010, pp. 21–30.
[11] L. Luceri, A. Deb, S. Giordano, and E. Ferrara, “Evolution of bot and human behavior during elections,” First Monday, vol. 24, no. 9, 2019.
[12] Z. Gilani, R. Furabakhsh, G. Tyson, L. Wang, and J. Crowcroft, “Of bots and humans (on Twitter),” in Proc. ASONAM, Jul. 2017, pp. 349–354.
[13] Securities and E. Commission. Twitter, Form 10-Q for the Period Ending June 30, 2014. Accessed: Dec. 2021. [Online]. Available: https://sec.report/Document/0001564590-14-003474/.
[14] Twitter. Twitter IDs, Accessed: Dec. 2021. [Online]. Available: https://developer.twitter.com/en/docs/twitter-ids.
[15] H. Kwak, C. Lee, H. Park, and S. Moon, “What is Twitter, a social network or a news media?” in Proc. WWW, 2010, pp. 591–600.
[16] M. Cha, H. Haddadi, F. Benevenuto, and K. Gummadi, “Measuring user influence in Twitter: The million follower fallacy,” in Proc. ICWSM, 2010, pp. 10–17.
[17] B. Krishnamurthy, P. Gill, and M. Arlitt, “A few chirps about Twitter,” in Proc. WOSN, 2008, pp. 19–24.
[18] M. Gjoka, M. Kurant, C. T. Butts, and A. Markopoulou, “Practical recommendations on crawling online social networks,” IEEE J. Sel. Areas Commun., vol. 29, no. 9, pp. 1872–1892, Oct. 2011.
[19] B. Ribeiro and D. Towsley, “Estimating and sampling graphs with multidimensional random walks,” in Proc. IMC, 2010, pp. 390–403.
[20] M. J. Salganik and D. D. Heckathorn, “Sampling and estimation in hidden populations using respondent-driven sampling,” Social. Methodol., vol. 34, no. 1, pp. 193–240, 2004.
[21] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee, “Measurement and analysis of online social networks,” in Proc. IMC, 2007, pp. 29–42.
[22] Y.-Y. Ahn, S. Han, H. Kwak, S. Moon, and H. Jeong, “Analysis of topological characteristics of huge online social networking services,” in Proc. WWW, 2007, pp. 835–844.
[23] Twitter. GET friends/ids. Accessed: Dec. 2021. [Online]. Available: https://developer.twitter.com/en/docs/twitter-api/v1/accounts-and-users/follow-search-get-users/api-reference/get-friends-ids

[24] Twitter. GET followers/ids. Accessed: Dec. 2021. https://developer.twitter.com/en/docs/twitter-api/v1/accounts-and-users/follow-search-get-users/api-reference/get-followers-ids

[25] K.-C. Yang, C. Torres-Lugo, and F. Menczer, “Prevalence of low-credibility information on Twitter during the COVID-19 outbreak,” in Proc. ICWSM CySoc, 2020.

[26] K. Shu, S. Wang, D. Lee, and H. Liu, Disinformation, misinformation, and fake news in social media. Cham, Switzerland: Springer, 2020.

[27] S. Stieglitz, F. Brachten, D. Berthelé, M. Schlaus, C. Venetopoulou, and D. Veutgen, “Do social bots (still) act different to humans?–Comparing metrics of social bots with those of humans,” in Proc. SCSM, 2017, pp. 379–395.

[28] V. Subrahmanian, A. Azaria, S. Durst, V. Kagan, A. Galstyan, K. Lerman, L. Zhu, E. Ferrara, A. Flammini, and F. Menczer, “The DARPA Twitter bot challenge,” Computer, vol. 49, no. 6, pp. 38–46, Jun. 2016.

[29] C. A. Davis, O. Varol, E. Ferrara, A. Flammini, and F. Menczer, “BotOrNot: A system to evaluate social bots,” in Proc. WWW Companion, 2016, pp. 273–274.

[30] M. Sayyadifarikandeh, O. Varol, K.-C. Yang, A. Flammini, and F. Menczer, “Detection of novel social bots by ensembles of specialized classifiers,” in Proc. CIKM, Oct. 2020, pp. 2725–2732.

[31] B. Ribeiro, P. Wang, F. Murai, and D. Towsley, “Sampling directed graphs with random walks,” in Proc. IEEE INFOCOM, Mar. 2012, pp. 1692–1700.

[32] M. Ribeiro, P. Calais, Y. Santos, V. Almeida, and W. Meira, “Characterizing and detecting hateful users on Twitter,” Proc. ICWSM, 2018, pp. 676–679.

[33] S. A. Catanese, P. De Meo, E. Ferrara, G. Fiumara, and A. Provetti, “Crawling Facebook for social network analysis purposes,” in Proc. WINS, 2011, pp. 1–8.

[34] L. Takac and M. Zabovsky, “Data analysis in public social networks,” in Proc. Int. Conf. Int. Workshop Present Day Trends Innov., 2012, vol. 1, no. 6.

[35] K. Nakajima and K. Shudo, “Estimating properties of social networks via random walk considering private nodes,” in Proc. KDD, Aug. 2020, pp. 720–730.

[36] S. A. Myers, A. Sharma, P. Gupta, and J. Lin, “Information network or social network? The structure of the Twitter follow graph,” in Proc. WWW, 2011, pp. 1–8.

[37] J. Leskovec and A. Krevl. (2014). SNAP Datasets: Stanford Large Network Dataset Collection. [Online]. Available: http://snap.stanford.edu/data

[38] J. Leskovec and J. Mcauley, “Learning to discover social circles in ego networks,” in Proc. NIPS, 2012, pp. 539–547.

[39] J. Kunegis, “KONECT: The Koblenz network collection,” in Proc. WWW Companion, 2013, pp. 1343–1350.

[40] J. Echeverria and S. Zhou, “Discovery, retrieval, and analysis of the ‘Star Wars’ botnet in Twitter,” in Proc. ASONAM, Jul. 2017, pp. 1–8.

[41] Twitter. GET statuses/show/id. Accessed: Dec. 2021. [Online]. Available: https://developer.twitter.com/en/docs/twitter-api/v1/tweets/post-and-engage/api-reference/get-statuses-show-id

**MEI FUKUDA** received the B.S. degree in mathematical and computing science from the Tokyo Institute of Technology, Tokyo, Japan, in 2019, where she is currently pursuing the master’s degree in mathematical and computing science. Her research interest includes social network analysis.

**KAZUKI NAKAJIMA** received the B.S. degree in information science and the M.S. degree in mathematical and computing science from the Tokyo Institute of Technology, in 2018 and 2020, respectively. Currently, he is pursuing the Ph.D. degree with the Department of Mathematical and Computing Science, Tokyo Institute of Technology. His research interests include social network analysis and network science.

**KAZUYUKI SHUDO** (Senior Member, IEEE) received the B.E., M.E., and Ph.D. degrees in computer science from Waseda University, in 1996, 1998, and 2001, respectively. He worked as a Research Associate with Waseda University, from 1998 to 2001. Later, he served as a Research Scientist at the National Institute of Advanced Industrial Science and Technology. In 2006, he joined as the Director and a Chief Technology Officer at Utagoe Inc. Since December 2008, he has been an Associate Professor with the Tokyo Institute of Technology. His research interests include distributed computing, programming language systems, and information security. He is a member of IEEE Computer Society, IEEE Communications Society, and ACM. He has received the Best Paper Award at SACSIS 2006; the Information Processing Society Japan (IPSJ) Best Paper Award, in 2006; the Super Creator Certification by Japanese Ministry of Economy Trade and Industry (METI) and Information Technology Promotion Agency (IPA), in 2007; the IPSJ Yamashita SIG Research Award, in 2008; the Funai Prize for Science, in 2010; the Young Scientists’ Prize; the Commendation for Science and Technology by the Minister of Education, Culture, Sports, and Technology, in 2012; and the IPSJ Nagao Special Researcher Award, in 2013.