

# ランダムウォークに基づく Twitter のボット割合の推定

福田 萌斐<sup>†</sup> 中嶋 一貴<sup>†</sup> 首藤 一幸<sup>†</sup>

<sup>†</sup> 東京工業大学 情報理工学院 数理・計算科学系

あらまし 近年、マーケティングや政治的介入、フェイクニュースの拡散などに寄与するソーシャルボットの台頭が指摘されている。本研究では、ランダムウォークに基づく Twitter のボット割合の推定手法を提案する。提案手法は、1 クエリで取得可能なフレンドやフォロワーの最大数が限られていること、非公開ユーザが一定の割合で存在することといった問題に取り組む。有向ソーシャルグラフのデータセットを用いてシミュレーション分析を行い、非公開ノードによるバイアスがほとんどないことや、ラベル付けされたノードの分布によっては提案手法が既存手法と比較してより小さい誤差を達成できることを示す。また、実際の Twitter のフォロワーグラフにおいて提案手法を適用し、Twitter のボット割合の推定値を提示する。実験の結果、2021 年 4 月～6 月の Twitter ユーザのうち 8～18 % のユーザがボットであることを示唆しており、これは先行研究の調査結果と整合的である。

キーワード ランダムウォーク, グラフサンプリング, 推定, ソーシャルボット, Twitter

## 1 はじめに

ソーシャルボット（ソーシャルネットワークにおいてコンテンツを自動生成するアカウント）の台頭が近年指摘されている。ソーシャルボットは、マーケティング [1] や政治介入 [2-4]、フェイクニュースの拡散 [5-7] など、さまざまな目的で作成されている。数億人のユーザを抱える Twitter のボットの特徴を調査した研究が数多く行われており、その中でもボットユーザの割合は、ボットが個人の意見や行動に与える影響を理解するための特徴量の 1 つである [8, 9]。Twitter 社は、2014 年に Twitter 上の厳密なボットユーザの割合を公式に報告している [10]。一方、データ保有者ではない第三者の研究者が、与えられたプラットフォームにおいてボットユーザの割合を独自に推定できるようにすることは重要である [5]。

Twitter 上のボット割合の推定は、データへのアクセスが制限されているため、第三者の研究者にとって困難である。一般に、一様サンプリングによって不偏推定量を得ることができる。しかし、Twitter のユーザ ID は 2022 年 1 月現在、64 ビット整数空間上にまばらに分布しているため、ユーザ ID を一様ランダムにサンプリングすることは現実的に困難である。Twitter ユーザをサンプリングする手法として、アプリケーションプログラミングインターフェース (API) を通じて最近の公開ツイートを取得し、そのツイートを投稿したユーザを収集する方法がある。しかし、この方法では、アクティブにツイートを投稿するユーザへのサンプリングのバイアスが発生してしまうため、このバイアスを補正するために、通常、数十万～数千万人程度といった多数のユーザをサンプリングする必要がある [8, 9]。

ユーザをサンプリングする他のアプローチとして、フォロワーグラフ（ユーザを表すノード集合とフォロー関係を表すエッジ集合からなる有向グラフ）に対してクロウリング手法を適用する方法がある [11-13]。特に、ソーシャルネットワークにお

る特定のラベルを持つノードの割合について、ランダムウォークを用いた推定手法について研究がなされてきた [14-16]。一般に、ソーシャルネットワークでは、クエリを実行したユーザの隣接ユーザを取得するための API が提供されており、2022 年 1 月現在、Twitter も隣接ユーザを取得するための API を提供している [17, 18]。そのため、ノードのラベルをボットかどうかを示す 2 値ラベルとすることで、既存の推定手法を用いた Twitter 上のボットの割合の推定が可能である。しかし、既存の推定手法は、Twitter 上のボット割合を推定する際の主に 2 つの問題に対処していない。第一に、1 クエリで取得可能なフレンドやフォロワーの最大数が 5,000 に制限されていることである。第二に、フレンド、フォロワー、ツイートなどの個人的なコンテンツを公開しない非公開ユーザが一定の割合で存在することである。

本研究では、ランダムウォークに基づく Twitter 上のボット割合の推定手法を提案する。ランダムウォークの各ステップでは、サンプリングしたユーザのフレンドやフォロワーを取得するクエリをそれぞれ 1 回だけ実行し、取得したフレンドやフォロワーからランダムに選択したノードに移動する。有向グラフのデータセットを用いたシミュレーション分析を行い、提案する推定手法が実際の Twitter のフォロワーグラフに対して有効であることを数値的に示す。最後に、2021 年 4 月から 6 月の Twitter 上のボットの割合について、3 種類の推定値を示す。この 3 つの推定値は一貫して Twitter ユーザの 8%～18% がボットであることを示唆しており、これは先行研究の調査結果と整合的である。

## 2 関連研究

### 2.1 Twitter ボットの特徴

Twitter ボットが個人の意見や行動に与える影響については、これまで盛んに研究されている。例えば、米国大統領選挙におけるソーシャルボットのツイート内容のシフトといった顕著な

活動について多くの研究がある [2–4]. また, Twitter のボットが組織的にフェイクニュースを拡散し, 陰謀論的な情報を増幅していることも報告されている [5–7].

ボットアカウントと非ボットアカウントの分類についても盛んに研究が行われている. Botometer ソフトウェア [19,20] は, API を通じて指定した Twitter ユーザをボットか人間かに正確に分類することを可能にしている. 本研究では, Twitter 上のボット割合を推定するために, Botometer を用いてサンプリングされた各ユーザがボットであるかどうかを判定する.

いくつかの先行研究では, Twitter のボット割合を推定している. Varol らは, Twitter のアクティブなアカウントの最大 15% がボットであることを示した [8]. Luceri らは 2016 年と 2018 年の大統領選挙中の期間を調査し, 12.6% のアカウントをボットと分類した [9]. これらの先行研究では, 数ヶ月に渡ってサンプリングされた数十万から数千万のユーザを使用している. これに対し, 提案手法では, 2.5 週間で得られたサンプルで先行研究と相違ないボット割合の値を推定する. また, Twitter が 2014 年に公式に報告したボットの割合である 8.5% という値とも整合的な値が得られている [10].

## 2.2 ランダムウォークに基づくグラフサンプリング

ランダムウォークは, クエリを実行したユーザの隣接ユーザを API で取得することが可能なソーシャルネットワークにおいて有効なサンプリング手法である [14,15]. 2022 年 1 月時点の Twitter フォロワーグラフでは, エンドポイント GET friends/ids と GET followers/ids の呼び出しにより, クエリを実行したユーザに入ってくるエッジと出ていくエッジの両方を取得することが許されている [17,18]. したがって, Twitter のフォロワーグラフにおけるボット割合の推定は, 無向ソーシャルグラフにおける特定のラベルを持つノードの割合を推定する問題に帰着する [14–16].

既存のラベル密度推定手法は, クエリを実行したノードの全隣接ノードが 1 回のクエリで取得可能であることを仮定している [14–16]. しかし, Twitter では 2022 年 1 月現在, 1 回のクエリで取得可能なフレンドやフォロワーの数が最大 5000 人に制限されている [17,18]. 本研究では, サンプリングされたユーザのフレンド, フォロワーを取得するために各クエリを 1 回だけ実行し, 取得したフレンド, フォロワーからランダムに選ばれたノードに移動する, ランダムウォークに基づくサンプリングアルゴリズムを提案する.

また, 既存の推定手法は, Twitter 上の個人的なコンテンツ (フレンド, フォロワー, ツイートなど) を公開しない非公開ユーザを対象としていない. しかし一般に, ソーシャルネットワークには一定割合の非公開ユーザが存在する (例えば, Facebook ネットワークでは 27% [21], Pokec ネットワークでは 34% [22]). Twitter は 2022 年 1 月現在, 非公開ユーザの割合を公式に発表していないが, 経験的に Twitter には非公開ユーザが一定数いると言える. 非公開ユーザがいると, 単純なランダムウォーク (ランダムに選ばれた隣接ノードを訪問する) ができなくなり, 推定量に偏りが生じてしまう [23]. 中嶋と首

藤は, 無向ソーシャルグラフに対して非公開ノードを考慮したランダムウォークに基づく推定手法の枠組みを設計した [23]. 本研究では, このフレームワークを, 非公開ノードを含む有向ソーシャルグラフにおいて, 特定のラベルを持つノードの割合を推定する場合に拡張した.

## 3 準備

### 3.1 記法

Twitter フォロワーグラフを有向グラフ  $G = (V, E)$  として表現する. ここで, ノード (ユーザ) の集合を  $V = \{v_1, v_2, \dots, v_n\}$  ( $n$  はノード数), 有向エッジ (フォロー関係) の集合を  $E$  とする. Twitter の用語に則し, 有向エッジ  $(v_i, v_j)$  が存在するとき, ノード  $v_j$  をノード  $v_i$  のフレンド, ノード  $v_i$  をノード  $v_j$  のフォロワーと呼ぶ [17,18]. グラフ  $G$  は多重エッジや自己ループを持たない. ノード  $v_i$  のフォロワーの数 (入次数) とフレンドの数 (出次数) をそれぞれ  $d_i^{\text{in}}, d_i^{\text{out}}$  と定義する. また, 入次数と出次数それぞれの最大値の大きい方の値を  $d_{\max}$  と定義する. それぞれのノード  $v_i$  はボットラベル  $l_{\text{bot}}(i) \in \{\text{bot}, \text{non-bot}\}$  を持つ.  $1_{\{cond\}}$  を条件  $cond$  を満たす場合に 1, そうでなければ 0 を返す指示関数としたとき, グラフ  $G$  上のボットの割合  $p_{\text{bot}}$  は以下のように表される.

$$p_{\text{bot}} = \frac{1}{n} \sum_{v_i \in V} 1_{\{l_{\text{bot}}(i)=\text{bot}\}}.$$

先行研究 [23] に従い, フレンドやフォロワー, ツイートといった個人的なコンテンツを公開していない非公開ノードを扱うための記法を取り入れる. それぞれのノード  $v_i$  はプライバシーラベル  $l_{\text{pri}}(i) \in \{\text{private}, \text{public}\}$  を持つ. ノード  $v_i$  をクエリしたと仮定する. もしノード  $v_i$  が公開ノード (すなわち,  $l_{\text{pri}}(i) = \text{public}$ ) であればノード  $v_i$  のフレンド, フォロワー, ツイートを取得することができ, 非公開ノード (すなわち,  $l_{\text{pri}}(i) = \text{private}$ ) であれば取得できない.  $G$  上の公開ノードによって構成される弱連結な部分グラフをパブリッククラスターと呼ぶ. 最大パブリッククラスターを  $C^* = (V^*, E^*)$  と定義する. 例えば, 図 1 に示される有向グラフには 3 つのパブリッククラスターが存在する.

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

この例において, 最大パブリッククラスターは  $C^* = C_1$  である.

また, 1 クエリで取得可能なフレンドやフォロワーの数を  $k$  で表す. 1 回のクエリで, フレンドやフォロワーのリストの後ろから  $k$  個ずつノードを取得できると仮定する. 例えば,  $k = 100$  の場合を考えると, もしあるノードの 520 人のフレンドと 450 人のフォロワーを全て取得したい場合, 我々はクエリを  $\lceil \frac{520}{100} \rceil + \lceil \frac{450}{100} \rceil = 11$  回使う必要がある. ここで,  $k = d_{\max}$  の場合, すなわち全てのフレンドやフォロワーが 1 クエリで取得可能な場合, 先行研究 [14–16,23] で採用されている仮定と同等である.

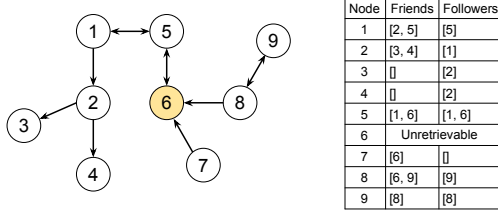


図 1: プライバシーラベル付きの有向グラフ (左図) とそれぞれのノードに対する取得可能なフレンドとフォロワーのリスト (右表)。ノード 6 は非公開ノードであり、他は公開ノードである。

### 3.2 仮 定

我々は、以下の仮定のもとで、ボット割合  $p_{\text{bot}}$  を推定する手法を提案する。

1. グラフ  $G$  は弱連結である。
2. グラフ  $G$  はランダムウォークを実行している間、静的である。
3. クエリされた公開ノードのすべてのフレンドとフォロワーのインデックスを取得できる。
4. 各ノードは確率  $p$  ( $0 \leq p < 1$ ) で独立に非公開ノードになり、そうでない場合は公開ノードとなる。
5. ランダムウォークの初期ノードは、 $G$  の最大パブリッククラスター上に存在する。

## 4 提案手法

本章では、Twitter のフォローグラフ上のボット割合に対して、ランダムウォークに基づく推定手法を提案する。まず、初期ノード  $v_{x_1} \in V^*$  を  $G$  の最大パブリッククラスターから選択する。ここで、 $x_i$  は  $i$  番目のサンプルノードのインデックスを表す ( $i = 1, 2, \dots$ )。次に、ノード  $v_{x_i}$  のフレンドとフォロワーにそれぞれ 1 回ずつクエリを実行する。得られた  $v_{x_i}$  のフレンドとフォロワーの部分集合をそれぞれ  $\Gamma_k^{\text{out}}(x_i)$ ,  $\Gamma_k^{\text{in}}(x_i)$  とする。このとき、それぞれの集合のサイズは  $|\Gamma_k^{\text{out}}(x_i)| = \min(k, d_i^{\text{out}})$ ,  $|\Gamma_k^{\text{in}}(x_i)| = \min(k, d_i^{\text{in}})$  で与えられる。ノード  $v_{x_i}$  から遷移する候補ノードの集合を  $\Gamma_k(x_i)$  と表し、以下のように定義する。

$$\Gamma_k(x_i) = \begin{cases} \Gamma_k^{\text{out}}(x_i) \cup \Gamma_k^{\text{in}}(x_i) & \text{if } i = 1, \\ \Gamma_k^{\text{out}}(x_i) \cup \Gamma_k^{\text{in}}(x_i) \cup \{v_{x_{i-1}}\} & \text{if } i \geq 2. \end{cases}$$

ノード  $v_{x_i}$  に初めて訪問した際、今後のステップで  $v_{x_i}$  へ再度クエリを実行することを避けるために、 $\Gamma_k(x_i)$  を記録する。 $v_{x_i}$  を再訪したときは、記録された集合を  $\Gamma_k(x_i) \cup \{v_{x_{i-1}}\}$  で置き換える。そして、集合  $\Gamma_k(x_i)$  からランダムにノード  $u$  を選択する。もしノード  $u$  が公開ノードであれば、次のサンプルノード  $v_{x_{i+1}}$  として  $u$  に移動し、そうでなければ集合  $\Gamma_k(x_i)$  から  $u$  をランダムに再選択する。 $r$  をサンプルサイズとしたとき、この遷移手順を  $r - 1$  回繰り返す。最後に、以下のボット割合の推定値  $\hat{p}_{\text{bot}}$  を得る。

$$\hat{p}_{\text{bot}} = \frac{\sum_{i=1}^r \mathbf{1}_{\{l_{\text{bot}}(x_i) = \text{bot}\}} / |\Gamma_k(x_i)|}{\sum_{i=1}^r 1 / |\Gamma_k(x_i)|}. \quad (1)$$

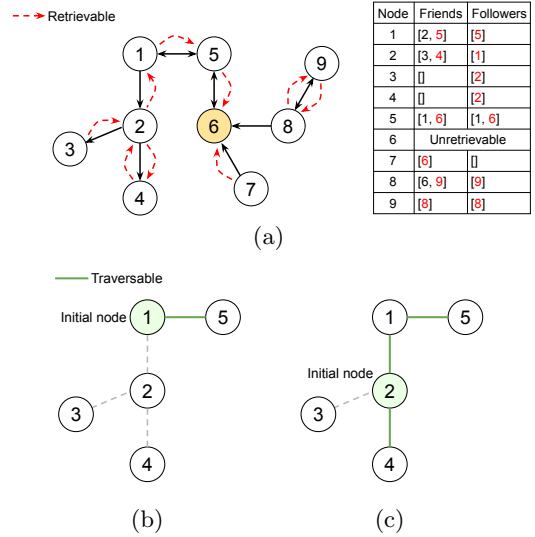


図 2:  $k = 1$  のときのランダムウォークにおける探索可能なノードと取得可能なノードの例。あるノードのフレンドとフォロワーはそれぞれのリストの後ろから取得される。

$i \geq 2$  のとき前のノード  $v_{x_{i-1}}$  を集合  $\Gamma_k(x_i)$  に含めるのは、取得されたフレンドとフォロワー (すなわち、 $\Gamma_k^{\text{out}}(x_i) \cup \Gamma_k^{\text{in}}(x_i)$ ) が全て非公開ノードとなることを避けるためである。図 2(a) は、図 1 で示したグラフ上の各ノードに対して、 $k = 1$  の時に取得できるフレンドとフォロワーを示している。ノード 1 からランダムウォークを開始すると、ノード 1 のフレンドとフォロワーにそれぞれ 1 回ずつクエリを実行し、 $\Gamma_k^{\text{out}}(1) = \{5\}$  と  $\Gamma_k^{\text{in}}(1) = \{5\}$  が得られる。次に、ノード 5 に移動し、ノード 5 のフレンドとフォロワーにそれぞれ 1 回ずつクエリを実行し、 $\Gamma_k^{\text{out}}(5) = \{6\}$  と  $\Gamma_k^{\text{in}}(5) = \{6\}$  を得る。しかし、ノード 6 が非公開ノードであるため、これ以上アルゴリズムを進めることができない。この問題を解決するために、ノード 5 の前のステップでサンプリングされたノード 1 を集合  $\Gamma_k(5)$  に含める。この操作により、 $i \geq 2$  の場合、集合  $\Gamma_k(x_i)$  に少なくとも 1 つの公開ノードが含まれることが保証される。初期ノード  $v_{x_1}$  が、ノード 5 のように  $\Gamma_k(x_1)$  が非公開ノードのみからなるようなノードである場合、 $G$  の最大パブリッククラスターから初期ノードの再選択を行う。

一般に、 $k$  が  $d_{\text{max}}$  より小さい場合、我々のランダムウォークで到達可能な公開ノードの数は、初期ノード  $v_{x_1}$  に依存する。例えば、図 1 の有向グラフにおいて  $k = 1$  の場合を考える。この場合の各ノードの取得可能なフレンドとフォロワーのリストを図 2(a) に示す。図 2(a) のノード 1 を初期ノードとした場合、 $\Gamma_k^{\text{out}}(1) = \{5\}$  and  $\Gamma_k^{\text{in}}(1) = \{5\}$  となる。したがって、到達可能な公開ノードの集合は  $\{1, 5\}$  となる (図 2(b) を参照)。図 2(a) において、ノード 2 を初期ノードとした場合、到達可能な公開ノードの集合は  $\{1, 2, 4, 5\}$  (図 2(c) を参照) となる。しかし、経験的に  $k$  が  $d_{\text{max}}$  に比べて小さすぎない限り、初期ノードによる到達可能な公開ノード数の分散はかなり小さく、最大パブリッククラスター上のほとんどの公開ノードが到達可能であることが分かっている。

表 1: データセット.

データセット	ノード数	エッジ数	$d_{\max}$
ego-Twitter [25]	81,306	1,768,135	3,383
YouTube [26]	1,134,885	4,938,950	28,564
Higgs [25, 26]	456,290	14,855,443	51,386
Flickr [26]	2,173,370	32,948,343	26,367

また、 $k = d_{\max}$  の場合、我々の提案手法による推定値に、非公開ノードによるバイアスがほとんど生じないことを理論的に示した [24].

## 5 シミュレーション分析

本章では、4つの有向グラフデータセットを用いて数値的なシミュレーションを行い、提案する推定手法が実際の Twitter のフォローグラフで有効であるかどうかを検証する. 各データセットについて、元の有向グラフの最大弱連結成分を取り出して使用する. 表 1 に、シミュレーションに用いた各有向グラフのノード数、エッジ数、ノードの最大入次数と最大出次数の大きい方の値 ( $d_{\max}$ ) を示している.

### 5.1 提案手法の性能

提案手法の精度がクエリごとに取得可能なフレンドやフォロワーの数、すなわち  $k$  に対してどのように変化するかを検証する. 我々は、ユーザの実際のボットラベルを含む有向グラフの公開データセットを見つけることができなかった. そこで、我々はグラフ上のノード  $v_i$  のボットラベル  $l_{\text{bot}}(i)$  を合成的に付与する 3つの方法: Random, LowDegree, HighDegree を検討する. Random 法では、ノード  $v_i$  を一様な確率、すなわち確率  $1/n$  で  $l_{\text{bot}}(i) = \text{bot}$  としてラベル付けする. LowDegree 法では、 $\check{d}_i = \max(d_i^{\text{in}}, d_i^{\text{out}})$  としたとき、ノード  $v_i$  を確率  $(1/\check{d}_i) / \sum_{v_i \in V} (1/\check{d}_i)$  で  $l_{\text{bot}}(i) = \text{bot}$  としてラベル付けする. HighDegree 法では、ノード  $v_i$  を確率  $\check{d}_i / \sum_{v_i \in V} \check{d}_i$  で  $l_{\text{bot}}(i) = \text{bot}$  としてラベル付けする. 各手法とも、ラベル付けされたノードの割合が所定の値に達するまで、対応する確率から選ばれたノードを繰り返しラベル付けする.

図 3(a)–(d) はそれぞれのデータセットに対し、全てのノードと、それぞれのラベル付け手法でラベル付けされた 10% のノードの入次数と出次数の大きい方の値  $\check{d}$  の相補累積分布 (CCDF) を示している. Random 法でラベル付けされたノードの度数  $\check{d}$  は、予想通り、全てのノードの度数  $\check{d}$  と同様の分布に従う. LowDegree 法と HighDegree 法でラベル付けされたノードの度数は、それぞれ低次と高次に偏っていることがわかる.

1回のシミュレーションを以下のように実行する. まず、全ノードの 10% に対して、3つのラベル付け手法 (Random, LowDegree, HighDegree) のいずれかを用いてラベル付けを行う. 次に、仮定 4 に従い、各ノードのプライバシーラベルを与えられた確率  $p$  で private に、それ以外は public に設定する. 第三に、各ノードのフレンドとフォロワーのリストの順序を独立にランダムにシャッフルする. 第四に、ランダムウォークの初期ノード

を最大パブリッククラスターからランダムに選択する. 第五に、ランダムウォークをサンプルサイズ  $r = 10,000$  で、1 クエリで取得可能なフレンドまたはフォロワーの数  $k$  に対して実行する.  $k$  は 1~10,000、確率  $p$  は 0.0~0.3 まで変化させる. 提案手法の精度と分散を測定するために、先行研究 [15, 23] に従い正規化平方二乗誤差 (NRMSE)、すなわち、 $(E[(\hat{p}_{\text{bot}}/p_{\text{bot}} - 1)^2])^{1/2}$  を用いる. 提案手法の NRMSE を独立した 100 回の実行に対し計算する.

図 4 に、提案手法の NRMSE を  $k$  の関数として示す. 全てのデータセットにおいて以下のことが言える. まず、すべてのラベル付け手法において、 $k \geq 10$  のときに非公開ノードによるバイアスがほとんど発生しない. 次に、全てのラベル付け手法において、 $k$  が小さい場合 (例えば  $k = 1$  の場合)、NRMSE が大きくなることがわかった. これは、 $k$  が小さいと初期ノードから到達可能な公開ノードの数かなり少なくなり、提案手法の推定量に偏りが生じるためである. さらに、全てのラベル付け手法において、 $k$  が大きくなると NRMSE が減少し、 $k$  がある値より大きくなると、 $k$  の NRMSE は  $d_{\max}$  のそれと同程度になる.  $k$  の臨界値はラベル付け手法によって異なり、例えば、ego-Twitter データセットでは、Random 法では  $k \approx 5$ 、LowDegree 法では  $k \approx 100$ 、HighDegree 法では  $k \approx 300$  が臨界値となる. ラベル付け手法によって  $k$  の臨界値が異なる直感的な理由は次の通りである. まず、 $k < d_{\max}$  のとき、初期ノードから探索できるエッジの集合が制限されるため、ノードの定常分布は典型的に偏る. しかし、Random 法によってラベル付けされたノードの度数  $\check{d}$  は一様に分布しているため、 $k$  の値が大きくなると NRMSE は急激に減少する. 一方、LowDegree 法と HighDegree 法では、ラベル付けされたノードの度数  $\check{d}$  がそれぞれ低次と高次に偏っているため、 $k$  の値が大きくなると NRMSE は比較的ゆっくりと減少していく. 特に、HighDegree 法では、高次のノードほど定常分布の偏りが大きくなるため、 $k$  の値が小さいときに NRMSE が典型的に大きくなる.

### 5.2 既存のランダムウォークとの比較

式 (1) で定義されるボット割合の推定値について、同じ数のクエリを用いた場合の我々のランダムウォークと既存のランダムウォーク [23] の NRMSE を比較する. 既存のランダムウォークでは、サンプリングした各ノードに対して、そのノードのフレンドやフォロワーをすべて取得するまでクエリの実行を繰り返し、その後、フレンドとフォロワーの和集合からランダムに選んだ公開ノードに移動する. 初めてノード  $v_i$  を訪れたとき、我々のランダムウォークではちょうど 2 回のクエリを実行するが、既存のランダムウォークでは与えられた  $k$  に対して  $\lceil d_i^{\text{in}}/k \rceil + \lceil d_i^{\text{out}}/k \rceil$  回のクエリが必要である.

各データセットに対して、1回のシミュレーションを以下のように実行する. まず、3つのラベル付け手法 (Random, LowDegree, HighDegree) のいずれかを用いて、全ノードの 10% をボットとして合成的にラベル付けする. 次に、仮定 4 に従い、各ノードのプライバシーラベルを与えられた確率  $p$  で private に、それ以外は public に設定する. ここで、 $p = 0.2$

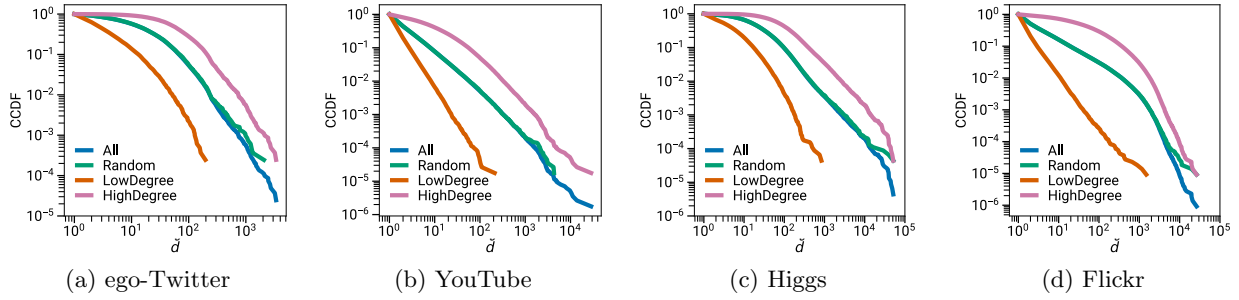


図 3: 全てのノードと、それぞれのラベル付け手法でラベル付けされた 10% のノードの入次数と出次数の大きい方の値  $\tilde{d}$  の相補累積分布 (CCDF)。

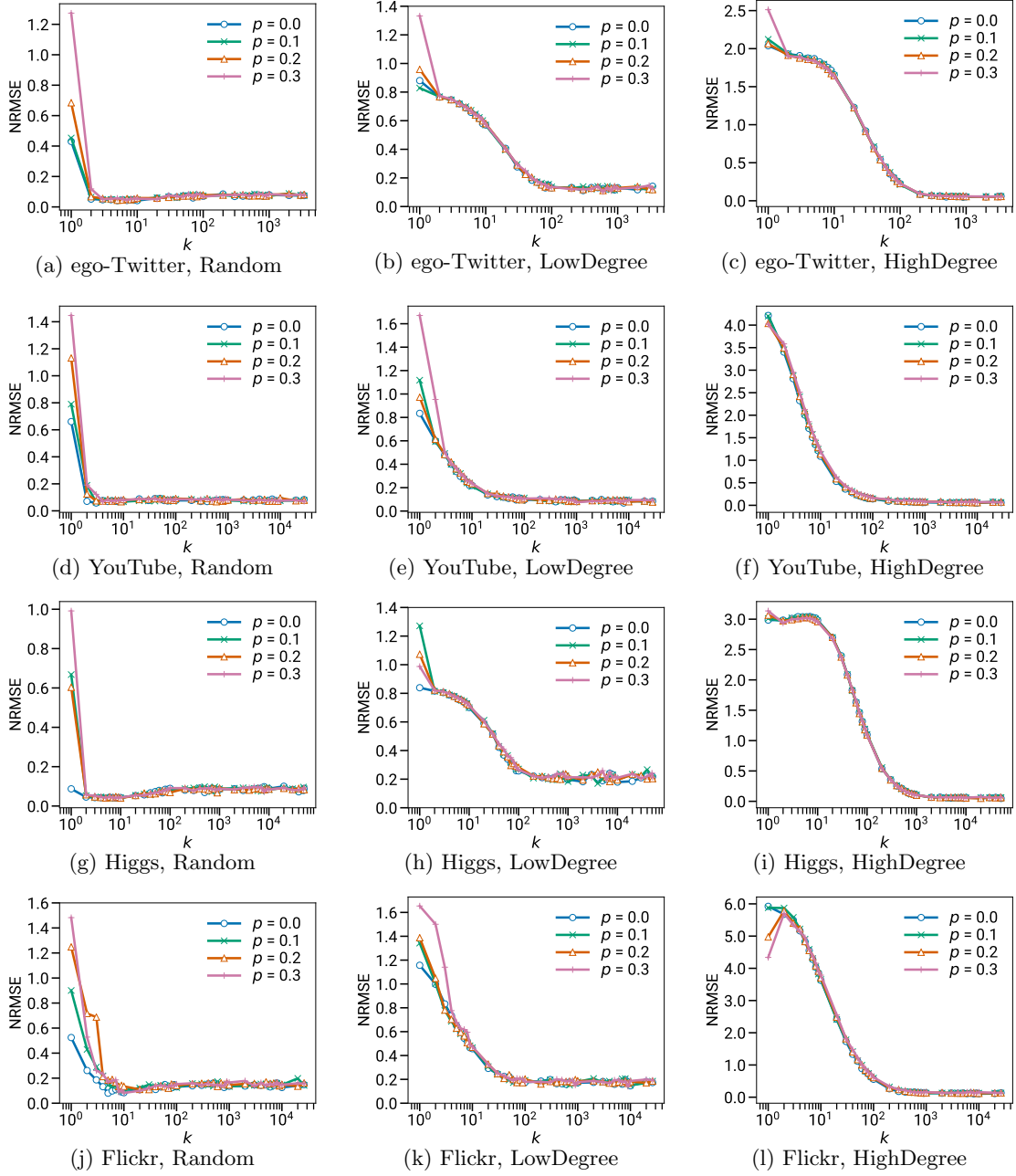


図 4: 各ラベル付け手法と各データセットに対する提案手法の NRMSE を  $k$  の関数として表したもの。

と設定するが、これは提案手法で非公開ノードによるバイアスがほとんど発生しないことを確認したためである (図 4 参照)。第三に、各ノードのフレンドとフォロワーのリストの順序を独立にランダムにシャッフルする。第四に、我々のランダム

ワークと既存のランダムワークの初期ノードを最大パブリッククラスターからランダムに選択する。第五に、与えられたクエリ数を用いて、我々のランダムワークと既存のランダムワークを実行する。各データセットにおいて、 $k$  の値は

$\Pr[\check{d} > k] \leq 0.05$  または  $\Pr[\check{d} > k] \leq 0.01$  が成り立つ最小の値、すなわちノードの入次数と出次数の大きい方の 95 パーセントイルと 99 パーセントイルにあたる値に設定する。それぞれのランダムウォークの推定値の NRMSE を独立した 100 回の実行で計算する。

図 5 は、ego-Twitter データセットにおける各ラベル付け手法の NRMSE をクエリ数の関数として示したものである。ego-Twitter データセットはグラフサイズが小さいため、クエリ数は 1,000~5,000 に設定した。図 6 は、YouTube, Higgs, Flickr の各データセットにおいて、各ラベル付け手法に対して 20,000 クエリを用いた場合のそれぞれのランダムウォークの NRMSE を示したものである。以下の考察は全てのデータセットに当てはまる。まず、Random 法と LowDegree 法において、どちらの  $k$  の値においても、同じクエリ数を用いた場合、我々のランダムウォークを用いた推定値の方が、既存のランダムウォークを用いた推定値よりも低い NRMSE を達成している。例えば、我々のランダムウォークは既存のランダムウォークに比べて、Flickr データセットにおいて  $k = 50$  のとき、Random 法では 77%, LowDegree 法では 69% NRMSE を改善する。次に、HighDegree 法では、いずれの  $k$  の値においても、我々のランダムウォークを用いた推定値の NRMSE は、既存のランダムウォークを用いた推定値より大きいことがわかる。これは、我々のランダムウォークが、 $k$  が小さいときに高い次数  $\check{d}$  を持つラベル付きノードの定常分布に偏りを生じさせるためである。

### 5.3 考察

我々のシミュレーションの数値的解析の結果、以下の 2 つの条件を満たす場合に、提案する推定手法が有効であることが分かった。第一に、ボットノードはグラフ全体に一樣に分布しているか、フレンド、フォロワー数が少ない方に偏っている必要がある。第二に、1 クエリで取得可能なフレンドやフォロワーの数  $k$  に対して、 $k$  より大きい数のフレンド、フォロワーを持つノード、すなわち、 $\check{d}_i > k$  を満たすノードの割合が十分小さくなければならない。

我々は、実際の Twitter のフォローグラフがこの 2 つの条件を満たしていると考えている。実証的な根拠は以下の通りである。まず、Twitter のボットはフレンドやフォロワーが少ない傾向にあることがいくつかの研究で示されている。Stieglitz らは、ボットは通常の人間のアカウントよりもフォロワー数が少ない傾向にあることを明らかにした [27]。35 万アカウント以上からなるボットグループを対象とした研究 [28] では、そのグループに所属するボットのフレンド数、フォロワー数がともに非常に少なく、フレンド数は 31 人以下、フォロワー数は 10 人以下であった。次に、いくつかの研究により、5,000 人以上のフレンドやフォロワーを持つユーザの割合が非常に小さいことが報告されている。なお、Twitter の API では、2022 年 1 月現在、1 回のクエリで取得可能なフレンドまたはフォロワーの最大数が 5,000 人に制限されている [17, 18]。先行研究によると、2009 年時点で Twitter において 5,000 人以上のフレンドまたはフォロワーを持つユーザは全体の 0.1% 未満である [11]。ま

た、Myers らの研究によると、2012 年時点で 470 人以上のフレンドまたはフォロワーを持つユーザは全体の 5% であり [29]、5,000 人以上のフレンドまたはフォロワーを持つユーザの割合は 5% より十分少ないと考えられる。したがって、我々のシミュレーション分析の結果、提案手法は Twitter 上のボット割合を適切に推定できることが期待される。

## 6 Twitter 上での推定

### 6.1 手順

実際の Twitter のフォローグラフに対して、我々の提案するランダムウォークを 3 回に分けて独立に実施した。初期ノードは筆頭著者のアカウント、Jack Dorsey のアカウント、Biz Stone のアカウントとした。各回とも  $r = 25,000$  のサンプルユーザを収集した。また、Facebook グラフにおけるランダムウォークサンプリングの前例 [14] と同様に、初期ノードへの依存性を十分に排除するために、最初の 5,000 のサンプルユーザを廃棄した。データの収集期間は、それぞれ 2021 年 4 月 13 日~30 日、5 月 6 日~23 日、5 月 23 日~6 月 9 日である。

我々は Twitter API の 3 つの関数を利用した。エンドポイント GET friends/ids と GET followers/ids を呼び出し、公開ユーザのフレンドとフォロワーを取得した [17, 18]。2022 年 1 月現在、これらのクエリを 15 分間に 15 回まで実行することが許されている。ユーザが非公開か公開かの判別は、エンドポイント GET statuses/show/:id を呼び出すことで行った。このエンドポイントは、2022 年 1 月現在、15 分間に 900 回の呼び出しを許可している [30]。

Twitter は、ある Twitter ユーザがボットであるかどうかを判断するための API を公式に提供していない。そこで、Botometer v4 [20] を用いて、サンプリングされた各公開ユーザがボットであるか否かを判定した。Botometer は、各ユーザに対し、それ以上のスコアを持つアカウントが自動化されているかどうかを示す条件付き確率を、ボットスコアとして提供する。我々は、スコアが所定の閾値  $\theta$  以上のユーザをボットとしてラベル付けした。ツイート数が少ないなどの問題でスコアが算出できなかったユーザは非ボットとした。

### 6.2 結果

図 7 は、3 つのサンプル列について、1 番目から  $r'$  番目のサンプルユーザ ( $1,000 \leq r' \leq 20,000$ ) から、それぞれ最初の 5,000 のサンプルユーザを破棄して計算したボット割合の推定値を示している。その結果、3 つの推定値とも、 $\theta = 0.95$  としたときに  $\approx 0.08$  に、 $\theta = 0.90$  としたときに  $\approx 0.18$  に収束し、提案手法の初期ノードに対する頑健性が確認された。我々の推定値 8%~18% は、Varol らの 2015 年時点の推定値 9%~15% [8]、Luceri らの 2019 年時点の推定値 12.6% [9] と十分に整合していることが分かる。

また、我々は 2021 年 10 月と 11 月にそれぞれ 10 日間、実際の Twitter 上で既存のランダムウォーク [23] を独立に複数回実行した。しかし、既存のランダムウォークでは数十人のユー

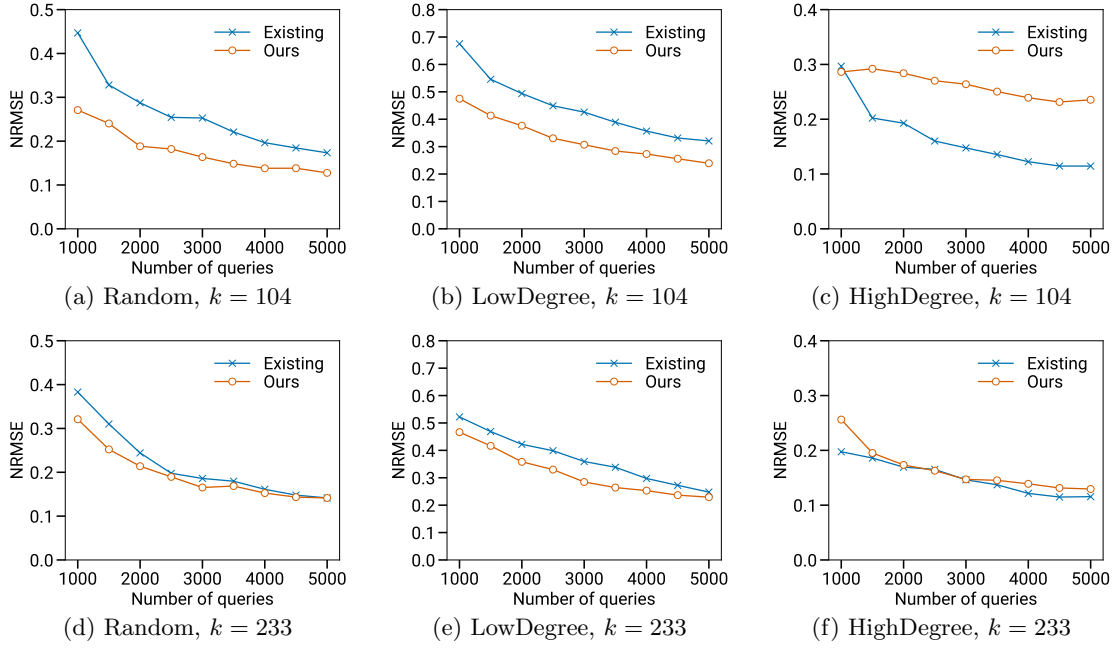


図 5: ego-Twitter データセットにおける、各ラベル付け手法に対する我々のランダムウォークと既存のランダムウォークの NRMSE をクエリ数の関数として表したもの。 (a)–(c) において、 $k$  の値は  $\Pr[\tilde{d} > k] \leq 0.05$  が成立する最小の値に、(d)–(f) では  $\Pr[\tilde{d} > k] \leq 0.01$  が成立する最小の値に設定している。

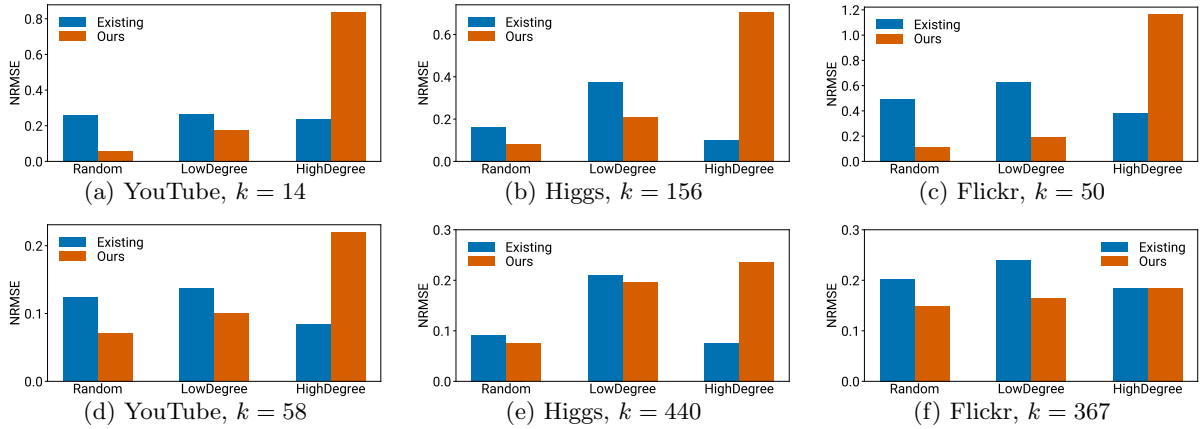


図 6: YouTube, Higgs, Flickr データセットにおける、各ラベル付け手法に対する我々のランダムウォークと既存のランダムウォークの NRMSE. クエリ数は 20,000 とした。  $k$  の値は (a)–(c) において  $\Pr[\tilde{d} > k] \leq 0.05$  が成立する最小の値に、(d)–(f) では  $\Pr[\tilde{d} > k] \leq 0.01$  が成立する最小の値に設定している。

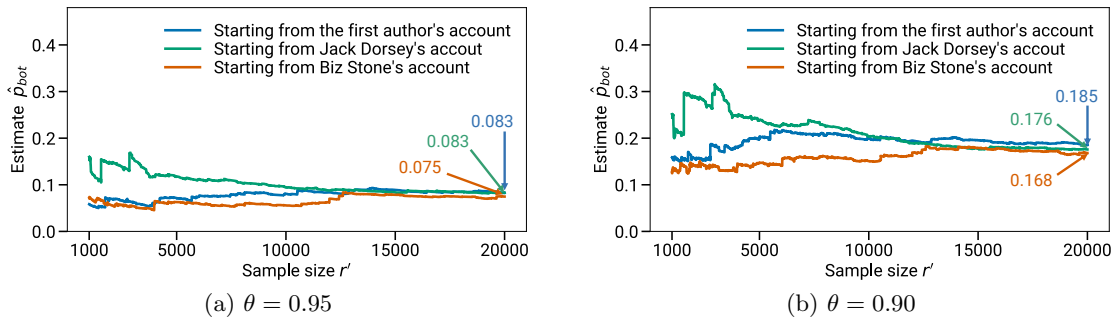


図 7: ランダムウォークにより得られた Twitter のボット割合の推定値。

ずしか収集できず、既存のランダムウォークは我々のランダムウォークに比べてクエリ数の面で効率が悪いことが示唆された。これは、既存のランダムウォークでは、膨大な数のフレンドやフォロワーを持つユーザのすべてのフレンドやフォロワーを取

得するために、大量のクエリを必要とするためである。実際、既存のランダムウォークでは、1 億人以上のフォロワーを持つ Katy Perry のアカウントを訪問し、Twitter API [18] を用いて全てのフォロワーを取得するのに 5 日以上かかっている。既



存のランダムウォークでは、膨大な数のフレンドやフォロワーを持つユーザを頻繁に訪問する傾向があるため、全体として膨大な数のクエリを必要とする。

## 7 おわりに

本研究では、Twitter 上のボット割合に対して、ランダムウォークに基づく推定手法を提案した。本研究では、Twitter 上のボット割合を推定する際の 2 つの大きな問題点に取り組んだ。(i) 1 回のクエリで取得できるフレンドやフォロワーの数が最大 5,000 に制限されていること、(ii) 一定割合のユーザが非公開ユーザであること、である。その結果、Twitter 上のボット割合を 8%~18%と推定し、先行研究で報告されている推定値と整合する値を得た。提案手法の推定精度はボット検出器 (Botometer など) の精度に依存するため、今後の課題はボット検出器の精度向上などであると考えられる。

謝辞 中嶋は JSPS 科研費 JP21J10415 の助成を受けました。

## 文 献

- [1] Seungbae Kim and Jinyoung Han. Detecting engagement bots on social influencer marketing. In *Proc. SocInfo*, pages 124–136, 2020.
- [2] Nir Grinberg, Kenneth Joseph, Lisa Friedland, Briony Swire-Thompson, and David Lazer. Fake news on Twitter during the 2016 U.S. presidential election. *Science*, 363(6425), 2019.
- [3] Emilio Ferrara, Herbert Chang, Emily Chen, Goran Muric, and Jaimin Patel. Characterizing social media manipulation in the 2020 U.S. presidential election. *First Monday*, 25(11), 2020.
- [4] Alexandre Bovet and Hernán A Makse. Influence of fake news in Twitter during the 2016 US presidential election. *Nature Communications*, 10(7), 2019.
- [5] David M. J. Lazer, Matthew A. Baum, Yochai Benkler, Adam J. Berinsky, Kelly M. Greenhill, Filippo Menczer, Miriam J. Metzger, Brendan Nyhan, Gordon Pennycook, David Rothschild, Michael Schudson, Steven A. Sloman, Cass R. Sunstein, Emily A. Thorson, Duncan J. Watts, and Jonathan L. Zittrain. The science of fake news. *Science*, 359(6380), 2018.
- [6] Chengcheng Shao, Giovanni Ciampaglia, Onur Varol, Alessandro Flammini, Filippo Menczer, and Kai-Cheng Yang. The spread of low-credibility content by social bots. *Nature Communications*, 9(4787), 2018.
- [7] Soroush Vosoughi, Deb Roy, and Sinan Aral. The spread of true and false news online. *Science*, 359(6380), 2018.
- [8] Onur Varol, Emilio Ferrara, Clayton B. Davis, Filippo Menczer, and Alessandro Flammini. Online human-bot interactions: Detection, estimation, and characterization. In *Proc. ICWSM*, pages 280–289, 2017.
- [9] Luca Luceri, Ashok Deb, Silvia Giordano, and Emilio Ferrara. Evolution of bot and human behavior during elections. *First Monday*, 24(9), 2019.
- [10] Securities and Exchange Commission. Twitter, Inc. Form 10-Q for the period ending June 30, 2014. <https://sec.report/Document/0001564590-14-003474/>. Accessed at January 2022.
- [11] Haewoon Kwak, Changhyun Lee, Hosung Park, and Sue Moon. What is Twitter, a social network or a news media? In *Proc. WWW*, pages 591–600, 2010.
- [12] Meeyoung Cha, Hamed Haddadi, Fabrizio Benevenuto, and Krishna Gummadi. Measuring user influence in Twitter: The million follower fallacy. In *Proc. ICWSM*, pages 10–17, 2010.
- [13] Balachander Krishnamurthy, Phillipa Gill, and Martin Arlitt. A few chirps about Twitter. In *Proc. WOSN*, pages 19–24, 2008.
- [14] Minas Gjoka, Maciej Kurant, Carter T. Butts, and Athina Markopoulou. Practical recommendations on crawling online social networks. *IEEE Journal on Selected Areas in Communications*, 29(9), 2011.
- [15] Bruno Ribeiro and Don Towsley. Estimating and sampling graphs with multidimensional random walks. In *Proc. IMC*, pages 390–403, 2010.
- [16] Matthew J Salganik and Douglas D Heckathorn. Sampling and estimation in hidden populations using respondent-driven sampling. *Sociological Methodology*, 34(1), 2004.
- [17] Twitter. GET friends/ids. <https://developer.twitter.com/en/docs/twitter-api/v1/accounts-and-users/follow-search-get-users/api-reference/get-friends-ids>. Accessed at January 2022.
- [18] Twitter. GET followers/ids. <https://developer.twitter.com/en/docs/twitter-api/v1/accounts-and-users/follow-search-get-users/api-reference/get-followers-ids>. Accessed at January 2022.
- [19] Clayton Allen Davis, Onur Varol, Emilio Ferrara, Alessandro Flammini, and Filippo Menczer. BotOrNot: A system to evaluate social bots. In *Proc. WWW Companion*, pages 273–274, 2016.
- [20] Mohsen Sayyadiharikandeh, Onur Varol, Kai-Cheng Yang, Alessandro Flammini, and Filippo Menczer. Detection of novel social bots by ensembles of specialized classifiers. In *Proc. CIKM*, pages 2725–2732, 2020.
- [21] Salvatore A Catanese, Pasquale De Meo, Emilio Ferrara, Giacomo Fiumara, and Alessandro Provetti. Crawling facebook for social network analysis purposes. In *Proc. WIMS*, pages 1–8, 2011.
- [22] Lubos Takac and Michal Zdobovskiy. Data analysis in public social networks. In *International scientific conference and international workshop present day trends of innovations*, volume 1, 2012.
- [23] Kazuki Nakajima and Kazuyuki Shudo. Estimating properties of social networks via random walk considering private nodes. In *Proc. KDD*, pages 720–730, 2020.
- [24] Mei Fukuda, Kazuki Nakajima, and Kazuyuki Shudo. Estimating the bot population on twitter via random walk based sampling. Available online: <https://doi.org/10.1109/ACCESS.2022.3149887>. IEEE Access 2022. in press.
- [25] Jure Leskovec and Andrej Krevl. SNAP Datasets: Stanford large network dataset collection. <http://snap.stanford.edu/data>, 2014.
- [26] Jérôme Kunegis. KONECT: The koblenz network collection. In *Proc. WWW Companion*, pages 1343–1350, 2013.
- [27] Stefan Stieglitz, Florian Brachten, Davina Berthelé, Mira Schlaus, Chrissoula Venetopoulou, and Daniel Veutgen. Do social bots (still) act different to humans? - comparing metrics of social bots with those of humans. In *Proc. SCSM*, pages 379–395, 2017.
- [28] Juan Echeverria and Shi Zhou. Discovery, retrieval, and analysis of the 'Star Wars' botnet in Twitter. In *Proc. ASONAM*, pages 1–8, 2017.
- [29] Seth A. Myers, Aneesh Sharma, Pankaj Gupta, and Jimmy Lin. Information network or social network? the structure of the Twitter follow graph. In *Proc. WWW Companion*, pages 493–498, 2014.
- [30] Twitter. GET statuses/show/:id. <https://developer.twitter.com/en/docs/twitter-api/v1/tweets/post-and-engage/api-reference/get-statuses-show-id>. Accessed at January 2022.