Heterogeneity-aware Twitter Bot Detection with Relational Graph Transformers

Shangbin Feng,[1] Zhaoxuan Tan,[1] Rui Li,[2] Minnan Luo[1]

[1]School of Computer Science and Technology, Xi'an Jiaotong University, Xi'an, China
[2]School of Continuing Education, Xi'an Jiaotong University, Xi'an, China
{wind_binteng, tanzhaoxuan}@stu.xjtu.edu.cn, {lrvberg,minnluo}@xjtu.edu.cn

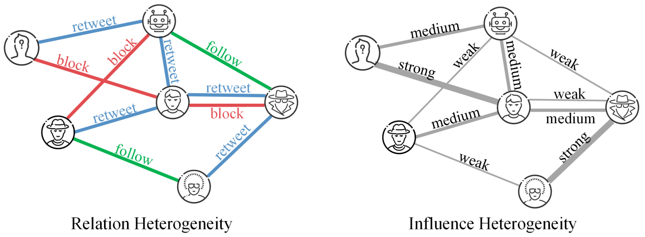
Abstract

Twitter bot detection has become an important and challenging task to combat misinformation and protect the integrity of the online discourse. State-of-the-art approaches generally leverage the topological structure of the Twittersphere, while they neglect the heterogeneity of relations and influence among users. In this paper, we propose a novel bot detection framework to alleviate this problem, which leverages the topological structure of user-formed heterogeneous graphs and models varying influence intensity between users. Specifically, we construct a heterogeneous information network with users as nodes and diversified relations as edges. We then propose relational graph transformers to model heterogeneous influence between users and learn node representations. Finally, we use semantic attention networks to aggregate messages across users and relations and conduct heterogeneity-aware Twitter bot detection. Extensive experiments demonstrate that our proposal outperforms state-of-the-art methods on a comprehensive Twitter bot detection benchmark. Additional studies also bear out the effectiveness of our proposed relational graph transformers, semantic attention networks and the graph-based approach in general.

Introduction

Twittier bots are Twitter accounts controlled by automated programs or the Twitter API. Bot operators often launch bot campaigns to pursue malicious goals, which harms the integrity of the online discourse. Over the past decade, Twitter bots were actively involved in election interference (Deb et al. 2019; Ferrara 2017), spreading misinformation (Cresci 2020) and promoting extreme ideology (Berger and Morgan 2015). Since malicious Twitter bots pose threat to online communities and induce undesirable social effects, effective Twitter bot detection measures are desperately needed.

Earlier works in Twitter bot detection generally rely on feature engineering, where an ample amount of user features are proposed and evaluated. Features extracted from tweets (Cresci et al. 2016) and user metadata (Yang et al. 2020; Lee and Kim 2013; Miller et al. 2014) were combined with traditional classifiers for bot detection. With the advent of deep learning, neural network based Twitter bot detectors were increasingly prevalent. Recurrent neural networks are adopted to encode tweets and detect bots based on their semantic content (Kudugunta and Ferrara 2018; Wei and Nguyen 2019). Self-supervised learning techniques were introduced to counter bot evolution (Feng et al. 2021b). Graph neural networks (Ali Alhosseini et al. 2019; Feng et al. 2021d) were later used to leverage the graph structure of the Twittersphere, while state-of-the-art methods are topology-aware in one way or another.



Relation Heterogeneity

Influence Heterogeneity

Figure 1: Users and bots of real-world social media interact in different ways and have varied influence over others, which result in relation and influence heterogeneity.

Despite earlier successes of leveraging the topological structure of the Twittersphere, these methods fail to recognize the intrinsic heterogeneity of Twitter and leverage it to identify subtle differences between genuine users and novel Twitter bots. Figure 1 illustrates two levels of heterogeneity that are pervasive on the real-world Twittersphere:

- **Relation Heterogeneity.** Twitter users are connected with different types of relations. For example, one user might like, comment, retweet or block another user, while these activities signal different relations between them.
- **Influence Heterogeneity.** Twitter users have different influence range and intensity over their neighbors on the Twittersphere. For example, distinguished news outlets might have a tremendous impact on the minds of many, while ordinary users generally inform close circles of their recent activities.

In this paper, we propose a novel Twitter bot detection framework that leverages the topological structure of the real-world Twittersphere, and on top of that, models pervasive heterogeneity of relation and influence to boost task performance. Specifically, we construct heterogeneous information networks with users as nodes and diversified rela-

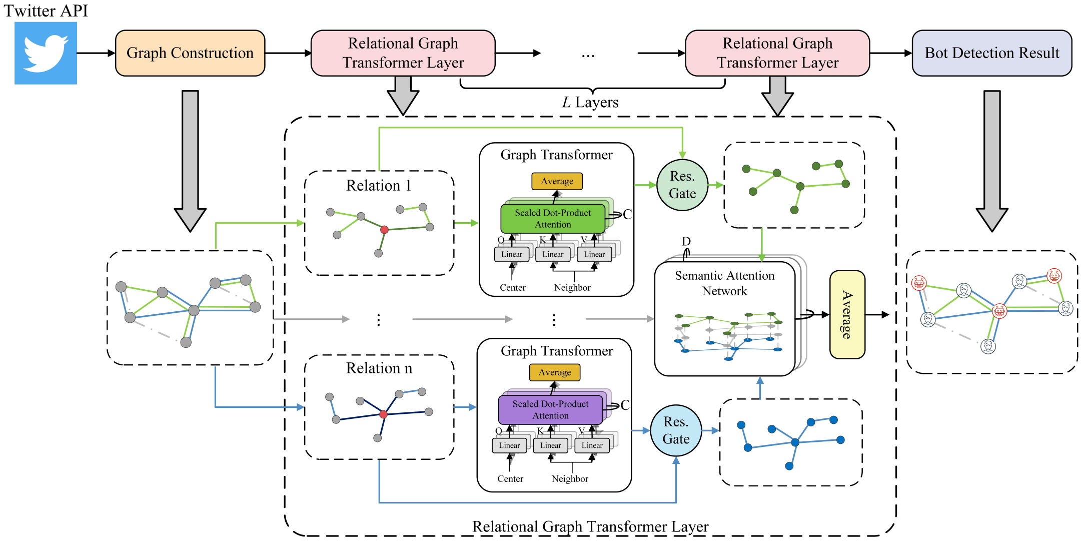


Figure 2: Overview of our graph-based and heterogeneity-aware Twitter bot detection framework.

tions as edges. We then propose relational graph transformers to model influence intensity with the attention mechanism and learn node representations. Finally, we adopt semantic attention networks to aggregate messages across users and relations and conduct bot detection. Our main contributions are summarized as follows:

- We propose to leverage relation and influence heterogeneity of the real-world Twittersphere, which enables our bot detection model to identify subtle differences between genuine users and bots and conduct robust bot detection.
- We propose a novel Twitter bot detection framework that is graph-based and heterogeneity-aware. It is an end-to-end bot detector that adopts relational graph transformers to leverage the topology and heterogeneity of the real-world Twittersphere.
- We conduct extensive experiments to evaluate our model and state-of-the-art methods on a comprehensive bot detection benchmark. Results demonstrate that our proposal consistently outperform all baseline methods. Further experiments also bear out the effectiveness of our graph-based and heterogeneity-aware approach.

Related Work

Twitter Bot Detection

Early Twitter bot detection models focus on manually designed features and combine them with traditional classifiers. These features are extracted from tweets (Cresci et al. 2016), user metadata (Yang et al. 2020; Lee and Kim 2013) or both (Miller et al. 2014). As deep learning later shows great promise and gains popularity, an increasing amount of neural network based bot detectors are proposed. Fully connected networks (Kudugunta and Ferrara 2018), recurrent neural networks (Wei and Nguyen 2019) and generative adversarial networks (Stanton and Irissappane 2019) are adopted in effective bot detection models to leverage different aspects of user information. SATAR (Feng et al. 2021b), a recently proposed framework, jointly leverages multi-modal user information with different deep architectures to improve upon these methods.

Although SATAR (Feng et al. 2021b) proposes to leverage the graph structure of the Twittersphere for bot detection, it does so in a feature engineering manner, rather than adopting state-of-the-art graph neural network architectures. Graph-based bot detectors were proposed to fill in the blanks. (Ali Alhosseini et al. 2019) views Twitter as a network of users and adopt graph convolutional networks to conduct bot detection. (Feng et al. 2021d) further constructs a heterogeneous information network to represent Twitter and uses relational GNNs for bot detection, which achieves state-of-the-art performance. However, these graph-based methods fail to incorporate the intrinsic heterogeneity of relation and influence on the real-world Twittersphere. In this paper, we build on these works and propose a heterogeneity-aware bot detector, which dynamically incorporates and leverages diversified relations and influence patterns between users.

Heterogeneous Information Networks

Real-world network data often consist of large quantities of diversified and interactive entities, which can be called heterogeneous information networks (HINs). HINs are widely adopted to model social networks (Wasserman, Faust et al. 1994; Otte and Rousseau 2002; Nguyen et al. 2020), link and graph mining (Getoor and Diehl 2005; Cook and Holder 2000) and natural language processing systems (De Cao, Aziz, and Titov 2018; Feng et al. 2021a). To effectively analyze HINs, (Schlichtkrull et al. 2018) proposes relational graph convolutional networks to extend GCN (Kipf and Welling 2016) to heterogeneous graphs. (Wang et al. 2019) proposes heterogeneous graph attention networks to extend GAT to heterogeneous graphs. In this paper, we build on these works to propose relational graph transformers and leverage Twitter heterogeneity.

Methodology

Overview

Figure 2 presents an overview of our proposed graph-based and heterogeneity-aware Twitter bot detector. Specifically, we firstly construct a heterogeneous information network with diversified relations to represent the Twittersphere. We then learn node representations under each relation with our proposed relational graph transformers. After that, we take a global view of the graph and dynamically aggregates representations across relations with semantic attention networks. Finally, we classify Twitter users into bots or genuine users and learn model parameters.

Graph Construction

We construct a heterogeneous information network (HIN) to represent the Twittersphere, which takes the relation heterogeneity into account and leverages diversified interactions between users. Specifically, we take Twitter users as nodes in the graph and we connect them with different types of edges, representing diversified relations on Twitter. We denote the set of relations in the HIN as R while our framework supports any relation settings.

Since this paper focuses on leveraging relation and influence heterogeneity to improve bot detection, we follow the same user information encoding procedure in the state-of-the-art approach (Feng et al. 2021d) for fairness. We denote user i's feature vector as x_i and transform it with a fully connected layer to serve as initial features in the GNNs, *i.e.*,

$$x_i^{(0)} = \sigma(W_I \cdot x_i + b_I) \tag{1}$$

where W_I and b_I are learnable parameters, σ denotes nonlinearity and we use leaky-relu as σ without further notice.

Relational Graph Transformers

Inspired by Transformers (Vaswani et al. 2017) and its success in natural language processing, we propose relational graph transformers, a GNN architecture that incorporates transformers and operates on HINs. We firstly obtain query, key and value for the c-th attention head with regard to relation r and node i, formulated as

$$\begin{aligned}
{q^{r}_{c,i}}^{(l)} &= {W^{r}_{c,q}}^{(l)} \cdot x_i^{(l-1)} + {b^{r}_{c,q}}^{(l)}, \\
{k^{r}_{c,j}}^{(l)} &= {W^{r}_{c,k}}^{(l)} \cdot x_j^{(l-1)} + {b^{r}_{c,k}}^{(l)}, \\
{v^{r}_{c,j}}^{(l)} &= {W^{r}_{c,v}}^{(l)} \cdot x_j^{(l-1)} + {b^{r}_{c,v}}^{(l)},
\end{aligned} \tag{2}$$

where q, k and v are query, key and value of the attention mechanism, (l) denotes the l-th layer of GNNs, all W and b are learnable parameters with regard to different relations and attention heads. We then model influence heterogeneity by calculating attention weights between different nodes by

$$ {\alpha^{r}_{c,ij}}^{(l)} = \frac{\langle {q^{r}_{c,i}}^{(l)}, {k^{r}_{c,j}}^{(l)} \rangle}{\sum_{u \in N^r(i)} \langle {q^{r}_{c,i}}^{(l)}, {k^{r}_{c,u}}^{(l)} \rangle}, \tag{3}$$

Algorithm 1: Model Learning Algorithm

```
input  : Twitter bot detection dataset $T$
output: Optimized model parameters $\theta$
initialize $\theta$;
construct Twitter HIN to obtain relations $R$;
encode user information to obtain $x_i$;
$x_i^{(0)}$ ←Equation (1);
while $\theta$ has not converged do
    for $r \in R$ do
        for each user $i \in T$ do
            find relation-based neighborhood $N^r(i)$;
            for $c \leftarrow 1$ to $C$ do
                for $j \in N^r(i)$ do
                    ${q^{r}_{c,i}}^{(l)}$, ${k^{r}_{c,j}}^{(l)}$, ${v^{r}_{c,j}}^{(l)}$ ← Equation (2);
                    ${\alpha^{r}_{c,ij}}^{(l)}$ ←Equation (3);
                end
            end
            $h_i^{r(l)}$ ← Equation (4 - 6);
        end
        for $d \leftarrow 1$ to $D$ do
            $\beta_d^r$ ←Equation (7 - 8);
        end
    end
    $x_i^{(L)}$ ← Equation (9);
    $Loss$ ← Equation (10 - 11);
    $\theta$ ← BackPropagate ($Loss$);
end
return $\theta$
```

where ${\alpha^{r}_{c,ij}}^{(l)}$ denotes the attention weight between nodes i and j, $\langle q, k \rangle = \exp(\frac{q^T k}{\sqrt{d}})$ is the exponential scale dot-product function where d is the hidden size of each attention head, $N^r(i)$ denotes node i's neighborhood with regard to relation r. We then aggregate over node neighborhood and attention heads to obtain node representation under relation r, *i.e.*,

$$u_i^{r(l)} = \frac{1}{C} \sum_{c=1}^{C} \left[\sum_{j \in N^r(i)} {\alpha^{r}_{c,ij}}^{(l)} \cdot {v^{r}_{c,j}}^{(l)} \right], \tag{4}$$

where $u_i^{r(l)}$ is the hidden representations of node i in l-th layer for relation r, C is the number of attention heads. We then apply the gate mechanism to obtained results to ensure smooth representation learning. We firstly obtain the gate level as follows

$$z_i^{r(l)} = sigmoid(W_A^r \cdot [u_i^{r(l)}, x_i^{(l)}] + b_A^r), \tag{5}$$

where $[\cdot, \cdot]$ is the concatenation operation, W_A and b_A are learnable parameters. We then apply the gate mechanism to learned representation $u_i^{r(l)}$ and input $x_i^{r(l)}$ by

$$h_i^{r(l)} = tanh(u_i^{r(l)}) \odot z_i^{r(l)} + x_i^{r(l)} \odot (1 - z_i^{r(l)}), \tag{6}$$

where $\odot$ denotes the Hadamard product operation and $h_i^{r(l)}$ is the learned representation of node i with regard to relation r in the l-th layer.

Semantic Attention Networks

After analyzing the HIN while separating different relations, we use semantic attention networks to aggregate node representations across relations while preserving the relation heterogeneity entailed in the Twitter HIN. Firstly, we obtain the importance of each relation by taking a global view of all nodes in the HIN, *i.e.*,

$$w_d^{r(l)} = \frac{1}{|V|}\sum_{i \in V} q_d^{(l)^T} \cdot tanh(W_{d,s}{}^{(l)} \cdot h_i^{r(l)} + b_{d,s}{}^{(l)}), \quad (7)$$

where $w_d^{r(l)}$ denotes the weight of relation r at the d-th attention head, V denotes the set of nodes in HIN, $q_d^{(l)}$ is the semantic attention vector at the d-th attention head in layer l, $q_d^{(l)}$, $W_{d,s}^{(l)}$ and $b_{d,s}^{(l)}$ are learnable parameters of the semantic attention network. We normalize the importance of each relation with softmax, formulated by

$$\beta_d^{r(l)} = \frac{\exp(w_d^{r(l)})}{\sum_{k \in R} \exp(w_d^{k(l)})}, \quad (8)$$

where $\beta_d^{r(l)}$ denotes the weight of relation r. We then fuse node representations under different relations with these weights as follows

$$x_i^{(l)} = \frac{1}{D}\sum_{d=1}^{D}\left[\sum_{r \in R} \beta_d^{r(l)} \cdot h_i^{r(l)}\right], \quad (9)$$

where $x_i^{(l)}$ denotes the output of layer l, $h_i^{r(l)}$ denotes the results of relational graph transformers and D is the number of attention heads in the semantic attention network.

Learning and Optimization

Each layer of GNN in our model contains a relational graph transformer and a semantic attention network. After L layers of GNNs, we obtain the final node representations $x^{(L)}$. We transform them with an output layer and a softmax layer for Twitter bot detection, *i.e.*,

$$\hat{y_i} = softmax(W_O \cdot \sigma(W_L \cdot x_i^{(L)} + b_L) + b_O), \quad (10)$$

where $\hat{y_i}$ is our model's prediction of user i, all W and b are learnable parameters. We then train our bot detector with supervised annotations and a regularization term, formulated as

$$Loss = -\sum_{i \in Y}[y_i \log(\hat{y_i}) + (1 - y_i)\log(1 - \hat{y_i})] + \lambda \sum_{w \in \theta} w^2, \quad (11)$$

where Y is the annotated user set, y_i is the ground-truth labels, θ denotes all trainable parameters in the model and λ is a hyperparameter. To sum up, Algorithm 1 presents the overall training schema of our proposed graph-based and heterogeneity-aware bot detection framework, with time complexity of $O(|E|)$ for each layer where E denotes the edge set, assuming embedding dimension and the number of relations are constants.

Experiments

Dataset

Our bot detection model is graph-based and heterogeneity-aware, which requires data sets that provide certain type of graph structure. TwiBot-20 (Feng et al. 2021c) is a comprehensive Twitter bot detection benchmark and the only publicly available bot detection dataset to provide user follow relationships to support graph-based methods. In this paper, we make use of TwiBot-20, which includes 229,573 Twitter users, 33,488,192 tweets, 8,723,736 user property items and 455,958 follow relationships. We follow the same splits provided in the benchmark so that results are directly comparable with previous works.

Baselines

We compare our graph-based and heterogeneity-aware approach with the following methods:

- **Lee *et al.*** (Lee, Eoff, and Caverlee 2011) extract features from Twitter user such as the longevity of account and combine them with random forest classifier.
- **Yang *et al.*** (Yang et al. 2020) use random forest classifier with minimal user metadata and derived features.
- **Cresci *et al.*** (Cresci et al. 2016) encode user activity sequences with strings and identify longest common substrings to identify bot groups.
- **Kudugunta *et al.*** (Kudugunta and Ferrara 2018) propose to jointly leverage user tweet semantics and user metadata.
- **Wei *et al.*** (Wei and Nguyen 2019) use recurrent neural networks to encode tweets and classify users based on their tweets.
- **Miller *et al.*** (Miller et al. 2014) extract 107 features from user tweets and metadata and frames the task of bot detection as anomaly detection.
- **Botometer** (Davis et al. 2016) is a bot detection service that leverages more than 1,000 user features.
- **SATAR** (Feng et al. 2021b) is a self-supervised representation learning framework of Twitter users that jointly leverages user tweets, metadata and neighborhood information. SATAR conducts bot detection by fine-tuning on specific bot detection data sets.
- **Alhosseini *et al.*** (Ali Alhosseini et al. 2019) use graph convolutional networks to learn user representations and conduct bot detection.
- **BotRGCN** (Feng et al. 2021d) constructs a heterogeneous graph to represent the Twittersphere and adopts relational graph convolutional networks for representation learning and bot detection. BotRGCN achieves state-of-the-art performance on the comprehensive TwiBot-20 benchmark.

Implementation

We use pytorch (Paszke et al. 2019), pytorch lightning (Falcon 2019), torch geometric (Fey and Lenssen 2019) and the transformers library (Wolf et al. 2020) for an efficient implementation of our proposed Twitter bot detection framework.

Table 1: Characteristic and performance of different Twitter bot detection methods. Deep, interactive, representative, graph-based and heterogeneity-aware denotes whether the method involves deep learning, leverages user interactions, learns user representation, involves graph neural networks or leverages Twitter heterogeneity.

Method	**Deep**	**Interactive**	**Representative**	**Graph-based**	**Heterogeneity-aware**	**Accuracy**	**F1-score**
Lee *et al.*						0.7456	0.7823
Yang *et al.*						0.8191	0.8546
Cresci *et al.*						0.4793	0.1072
Kudugunta *et al.*	✓					0.8174	0.7515
Wei *et al.*	✓					0.7126	0.7533
Miller *et al.*		✓				0.4801	0.6266
Botometer		✓				0.5584	0.4892
SATAR	✓	✓	✓			0.8412	0.8642
Alhosseini *et al.*	✓	✓	✓	✓		0.6813	0.7318
BotRGCN	✓	✓	✓	✓		0.8462	0.8707
Ours	✓	✓	✓	✓	✓	**0.8664**	**0.8821**

Table 2: Hyperparameter settings of our model. We make use of follower and following information as relations R, while we discuss more choices in heterogeneity study.

Hyperparameter	**Value**
optimizer	AdamW
learning rate	10^{-3}
L_2 regularization λ	3×10^{-5}
batch size	256
layer count L	2
dropout	0.5
size of hidden state	128
maximum epochs	40
transformer attention heads C	8
semantic attention heads D	8
relational edge set R	$\{follower, following\}$

We present our hyperparameter settings in Table 2 to facilitate reproduction. Our implementation is trained on a Titan X GPU with 12GB memory. Our implementation is publicly available on GitHub. [1].

Experiment Results

We firstly evaluate whether these methods involve deep learning, leverage user interactions, learn user representation, involve graphs and graph neural networks or leverage Twitter heterogeneity. We then benchmark these bot detection models on TwiBot-20 (Feng et al. 2021c) and present results in Table 1. It is demonstrated that:

- Our proposal consistently outperforms all baselines, including the state-of-the-art BotRGCN (Feng et al. 2021d).

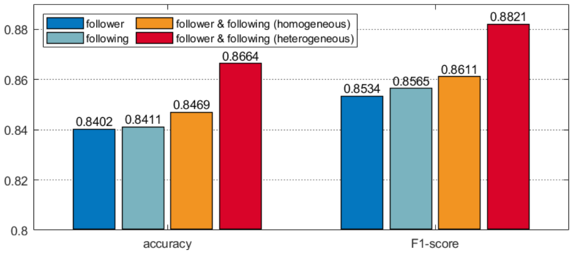


Figure 3: Ablation studying removing different parts of graph structure of our constructed Twitter HINs.

- Successful graph-based methods, such as BotRGCN (Feng et al. 2021d) and ours, generally outperform traditional approaches that do not consider the Twittersphere as graphs and networks. These results demonstrate the importance of modeling the topological structure of Twitter for bot detection.
- We propose the first heterogeneity-aware bot detection frameworks, which achieves the best performance on a comprehensive benchmark. These results bear out the necessity of leveraging Twitter heterogeneity and the effectiveness of our proposed approach.

In the following, we firstly study the role of graphs and heterogeneity in our proposed approach. We then examine the data efficiency and representation learning capability of our bot detection method.

Graph Learning Study

We propose a graph-based bot detection model, which leverages the topological structure of the Twittersphere to capture subtle patterns and better identify bots. Specifically, we adopt user follower and following relationships as two types of edges that connects users as nodes to form a HIN. To prove the effectiveness of our proposed graph construction

[1] https://github.com/BunsenFeng/BotHeterogeneity

Table 3: Ablation study of our proposed GNN architecture. RT and SA denote relational transformers and semantic attention networks respectively.

Ablation Settings	Accuracy	F1-score
full model	**0.8664**	**0.8821**
remove transformer in RT	0.8521	0.8679
remove gated residual in RT	0.8478	0.8646
replace RT with GAT	0.8571	0.8726
replace RT with GCN	0.8444	0.8619
replace RT with SAGE	0.8546	0.8687
summation as SA	0.8512	0.8654
mean pooling as SA	0.8512	0.8663
max pooling as SA	0.8495	0.8629
min pooling as SA	0.8555	0.8704

approach, we remove different types of edges and report results under these ablation settings in Figure 3. It is illustrated that the complete graph structure, with both follower and following edges, outperforms any reduced settings. These results prove the effectiveness of our constructed HIN to model relation heterogeneity on Twitter.

Upon obtaining a HIN, we propose relational graph transformers to propagate node messages and learn representations. To prove the effectiveness of our proposed GNN architecture, we conduct ablation study on relational graph transformers and report results under different settings in Table 3. It is demonstrated that transformers, the gate mechanism and the semantic attention networks are all essential parts of our proposed GNN architecture.

To sum up, both our constructed HIN and our proposed GNN architecture contribute to our model's outstanding performance, which bears out the effectiveness of our graph-based approach.

Heterogeneity Study

Our bot detection proposal models the intrinsic heterogeneity of Twitter to identify subtle anomalies of bots and conduct robust bot detetcion. We study the effects of incorporating heterogeneity and present our findings.

Relation Heterogeneity Relation heterogeneity refers to the fact that there are diversified relations between users on the real-world Twittersphere. Our bot detection model incorporates relation heterogeneity by constructing HINs and leveraging them with relational GNNs. Different HINs could be constructed with different relation sets R, thus we propose different relation heterogeneity settings and present their results in Figure 4. It is illustrated that most heterogeneous relation settings outperform their homogeneous counterpart, which proves the necessity of modeling relation heterogeneity for Twitter bot detection.

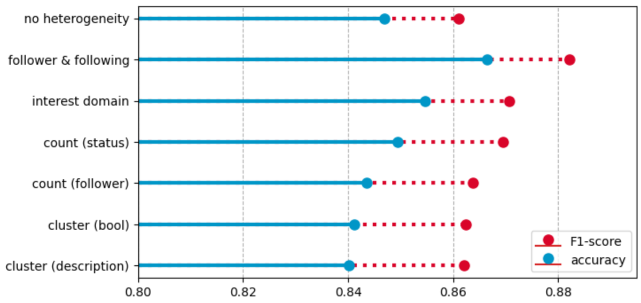


Figure 4: Performance of our proposed method with different relation heterogeneity settings. We cluster users with description and bool features, divide users with follower and status counts, leverage user interest domain in the dataset as well as following information to construct HINs.

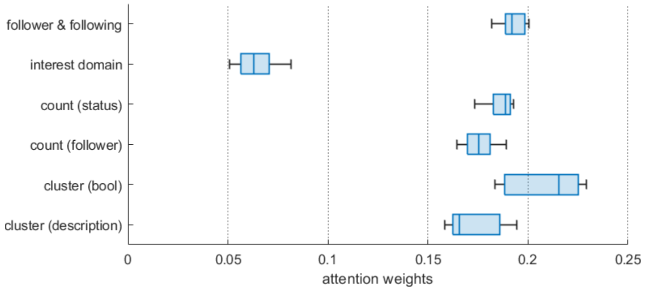


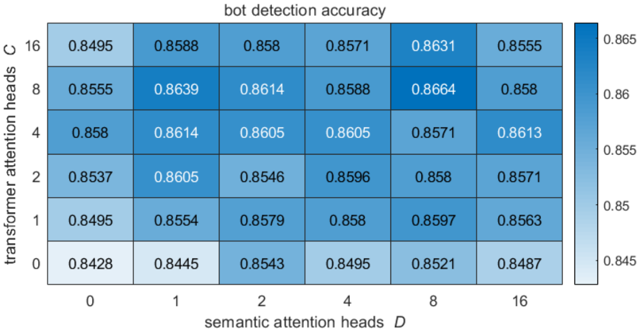
Figure 5: Attention weights of different sets of relations that co-exist on the real-world Twittersphere. We present the first, second and third quartile of results from multiple runs.

To identify relation types that are crucial in Twitter bot detection, we combine all relations to form a comprehensive graph and use weights from the semantic attention networks to identify significant relations. Experiment results in Figure 5 demonstrate that most heterogeneous relations contribute equally to our method's performance, while the user interest domain information in the data set is not as effective.

To sum up, we improve bot detection performance by incorporating relation heterogeneity and most relations are significant in our method's decision making.

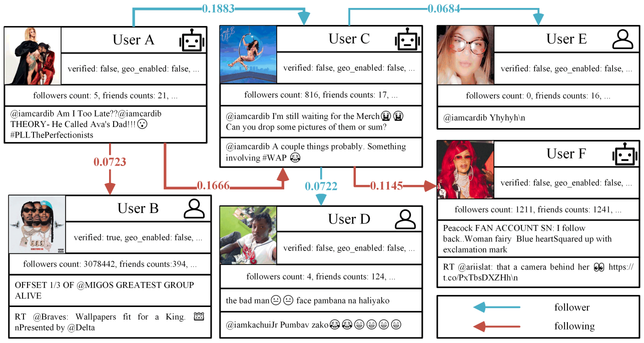
Influence Heterogeneity Influence heterogeneity refers to the fact that Twitter users have different patterns and intensity of influence over others on social media. We leverage influence heterogeneity with the multi-head attention mechanism in relational graph transformers. To validate the effectiveness of this approach, we conduct ablation study on the attention mechanism and present results in Figure 6. It is illustrated that incorporating the attention mechanism ($C > 0$, $D > 0$) outperforms methods without it ($C = 0$, $D = 0$). Besides, adopting multi-head attention networks ($C > 1$, $D > 1$) generally outperforms their single-head counterparts ($C = 1$, $D = 1$), proving the effectiveness of our design choices.

After proving the necessity of leveraging influence heterogeneity, we study a specific cluster of Twitter users and present their attention weights in Figure 7. It is illustrated that influence weights between bots are generally larger. By modeling influence heterogeneity, our method identify bots that act in groups and substantially influence each other.



transformer attention heads C \ semantic attention heads D	0	1	2	4	8	16
16	0.8495	0.8588	0.858	0.8571	0.8631	0.8555
8	0.8555	0.8639	0.8614	0.8588	0.8664	0.858
4	0.858	0.8614	0.8605	0.8605	0.8571	0.8613
2	0.8537	0.8605	0.8546	0.8596	0.858	0.8571
1	0.8495	0.8554	0.8579	0.858	0.8597	0.8563
0	0.8428	0.8445	0.8543	0.8495	0.8521	0.8487

Figure 6: Ablation study of the attention mechanism in relational graph transformers and semantic attention networks.



User A
verified: false, geo_enabled: false, ...
followers count: 5, friends counts: 21, ...
@iamcardib Am I Too Late??@iamcardib THEORY- He Called Ava's Dad!!! 😮 #PLLThePerfectionists

User C
verified: false, geo_enabled: false, ...
followers count: 816, friends counts: 17, ...
@iamcardib I'm still waiting for the Merch😭😭 Can you drop some pictures of them or sum?
@iamcardib A couple things probably. Something involving #WAP 😂

User E
verified: false, geo_enabled: false, ...
followers count: 0, friends counts: 16, ...
@iamcardib Yhyhyh\n

User F
verified: false, geo_enabled: false, ...
followers count: 1211, friends counts: 1241, ...
Peacock FAN ACCOUNT SN: I follow back..Woman fairy Blue heartSquared up with exclamation mark
RT @ariislat: that a camera behind her 👀 https://t.co/PxTbsDXZHh\n

User B
verified: true, geo_enabled: false, ...
followers count: 3078442, friends counts:394, ...
OFFSET 1/3 OF @MIGOS GREATEST GROUP ALIVE
RT @Braves: Wallpapers fit for a King. 👑 nPresented by @Delta

User D
verified: false, geo_enabled: false, ...
followers count: 4, friends counts: 124, ...
the bad man😐😐 face pambana na haliyako
@iamkachuiJr Pumbav zako😂😂😁😁😁😁

Figure 7: Example cluster of six real-world Twitter users and the attention weights between them.

To sum up, we improve bot detection performance by leveraging influence heterogeneity, while attention weights between users in the network yield valuable insights into our model's decision making.

Data Efficiency Study

Existing bot detection models are generally supervised and rely on large quantities of data annotations, while bot detection data sets are generally limited in size and labels. To examine the data efficiency of our bot detection model, we present performance with partial training sets, randomly removed edges and masked user features in Figure 8. It is illustrated that our method would still outperform the state-the-of-art BotRGCN (Feng et al. 2021d) with as little as 40% training data and is also robust to changes in user interactions. Model performance drops significantly with reduced user features, which suggest that Twitter bot detection still rely on comprehensive analysis of user information in addition to the graph structure.

Representation Learning Study

Our model, as well as few baselines, learn representation for Twitter users and identify bots with them. To examine the quality of representation learning with our proposal, we present the t-sne plot of user representation of our method and baselines in Figure 9. It is illustrated that our result shows higher levels of collocation for groups of genuine users and Twitter bots, which indicates that our method learns high-quality user representation.

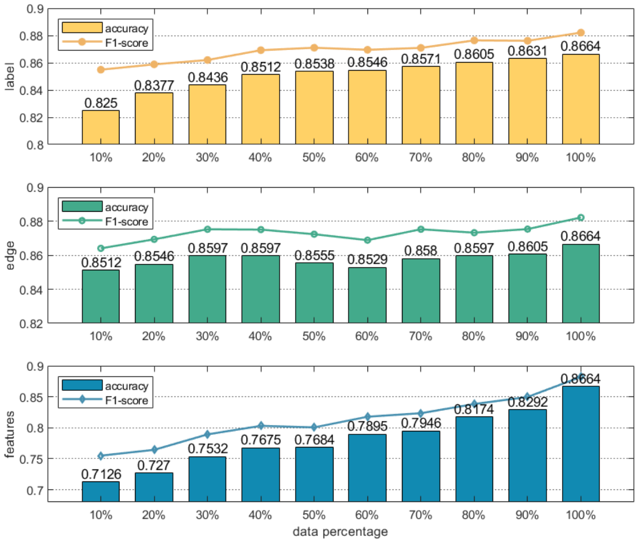


Figure 8: Model performance with limited data annotation, user interactions and user features.

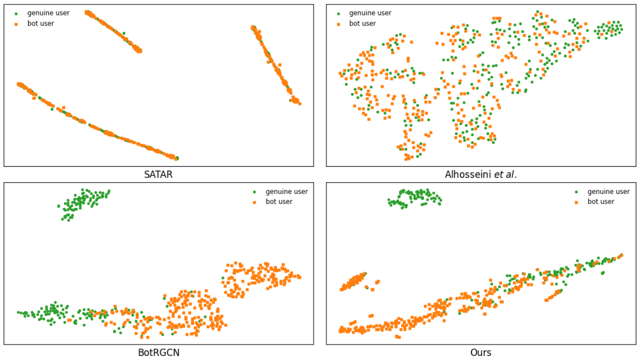


Figure 9: Plots of user representations learned with our method and different baselines.

Conclusion and Future Work

Twitter bot detection is an important and challenging task. We proposed a graph-based and heterogeneity-aware bot detection framework, which constructs HINs to represent the Twittersphere, adopt relational graph transformers and semantic attention networks for representation learning and bot detection. We conducted extensive experiments on a comprehensive benchmark, which demonstrates that our method consistently outperforms state-of-the-art baselines. Further exploration proves our method's graph learning strategy and the inclusion of Twitter heterogeneity are generally effective, while also performs well with limited data and learns high-quality representation for Twitter users. We plan to experiment with more diversified ways to model the Twittersphere as graphs and extend our graph-based bot detection approach in the future.

Acknowledgments

This work was supported by National Nature Science Foundation of China (No. 61872287, No. 62137002 and No. 62050194), CCF-AFSG Research Fund, Innovative Research Group of the National Natural Science Foundation of China (No. 61721002), Innovation Research Team of Ministry of Education (IRT_17R86), Project of China Knowledge Center for Engineering Science and Technology and The Consulting Research Project of Chinese Academy of Engineering "The Online and Offline Mixed Educational Service System for The Belt and Road" Training in MOOC China. We would also like to thank Zilong Chen for his help during the rebuttal, Herun Wan and Ningnan Wang for their contributions in preceding works, and all LUD lab members for our collaborative research environment.

References

Ali Alhosseini, S.; Bin Tareaf, R.; Najafi, P.; and Meinel, C. 2019. Detect Me If You Can: Spam Bot Detection Using Inductive Representation Learning. In *Companion Proceedings of The 2019 World Wide Web Conference*, 148–153.

Berger, J. M.; and Morgan, J. 2015. The ISIS Twitter Census: Defining and describing the population of ISIS supporters on Twitter. *The Brookings project on US relations with the Islamic world*, 3(20): 4–1.

Cook, D. J.; and Holder, L. B. 2000. Graph-based data mining. *IEEE Intelligent Systems and Their Applications*, 15(2): 32–41.

Cresci, S. 2020. A Decade of Social Bot Detection. *Commun. ACM*, 63(10): 72–83.

Cresci, S.; Di Pietro, R.; Petrocchi, M.; Spognardi, A.; and Tesconi, M. 2016. DNA-inspired online behavioral modeling and its application to spambot detection. *IEEE Intelligent Systems*, 31(5): 58–64.

Davis, C. A.; Varol, O.; Ferrara, E.; Flammini, A.; and Menczer, F. 2016. Botornot: A system to evaluate social bots. In *Proceedings of the 25th international conference companion on world wide web*, 273–274.

De Cao, N.; Aziz, W.; and Titov, I. 2018. Question answering by reasoning across documents with graph convolutional networks. *arXiv preprint arXiv:1808.09920.*

Deb, A.; Luceri, L.; Badaway, A.; and Ferrara, E. 2019. Perils and Challenges of Social Media and Election Manipulation Analysis: The 2018 US Midterms. In *Companion Proceedings of The 2019 World Wide Web Conference*, WWW '19, 237–247. New York, NY, USA: Association for Computing Machinery. ISBN 9781450366755.

Falcon, e. a., WA. 2019. PyTorch Lightning. *GitHub. Note: https://github.com/PyTorchLightning/pytorch-lightning*, 3.

Feng, S.; Chen, Z.; Li, Q.; and Luo, M. 2021a. Knowledge Graph Augmented Political Perspective Detection in News Media. *arXiv preprint arXiv:2108.03861.*

Feng, S.; Wan, H.; Wang, N.; Li, J.; and Luo, M. 2021b. SATAR: A Self-supervised Approach to Twitter Account Representation Learning and its Application in Bot Detection. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, 3808–3817.

Feng, S.; Wan, H.; Wang, N.; Li, J.; and Luo, M. 2021c. TwiBot-20: A Comprehensive Twitter Bot Detection Benchmark. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, 4485–4494.

Feng, S.; Wan, H.; Wang, N.; and Luo, M. 2021d. BotRGCN: Twitter Bot Detection with Relational Graph Convolutional Networks. *arXiv preprint arXiv:2106.13092.*

Ferrara, E. 2017. Disinformation and Social Bot Operations in the Run Up to the 2017 French Presidential Election. *CoRR*, abs/1707.00086.

Fey, M.; and Lenssen, J. E. 2019. Fast graph representation learning with PyTorch Geometric. *arXiv preprint arXiv:1903.02428.*

Getoor, L.; and Diehl, C. P. 2005. Link mining: a survey. *Acm Sigkdd Explorations Newsletter*, 7(2): 3–12.

Kipf, T. N.; and Welling, M. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907.*

Kudugunta, S.; and Ferrara, E. 2018. Deep neural networks for bot detection. *Information Sciences*, 467: 312–322.

Lee, K.; Eoff, B.; and Caverlee, J. 2011. Seven months with the devils: A long-term study of content polluters on twitter. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 5.

Lee, S.; and Kim, J. 2013. Warningbird: A near real-time detection system for suspicious urls in twitter stream. *IEEE transactions on dependable and secure computing*, 10(3): 183–195.

Miller, Z.; Dickinson, B.; Deitrick, W.; Hu, W.; and Wang, A. H. 2014. Twitter spammer detection using data stream clustering. *Information Sciences*, 260: 64–73.

Nguyen, V.-H.; Sugiyama, K.; Nakov, P.; and Kan, M.-Y. 2020. Fang: Leveraging social context for fake news detection using graph representation. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, 1165–1174.

Otte, E.; and Rousseau, R. 2002. Social network analysis: a powerful strategy, also for the information sciences. *Journal of information Science*, 28(6): 441–453.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32: 8026–8037.

Schlichtkrull, M.; Kipf, T. N.; Bloem, P.; Van Den Berg, R.; Titov, I.; and Welling, M. 2018. Modeling relational data with graph convolutional networks. In *European semantic web conference*, 593–607. Springer.

Stanton, G.; and Irissappane, A. A. 2019. GANs for semi-supervised opinion spam detection. *arXiv preprint arXiv:1903.08289.*

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *Advances in neural information processing systems*, 5998–6008.

Wang, X.; Ji, H.; Shi, C.; Wang, B.; Ye, Y.; Cui, P.; and Yu, P. S. 2019. Heterogeneous graph attention network. In *The World Wide Web Conference*, 2022–2032.

Wasserman, S.; Faust, K.; et al. 1994. Social network analysis: Methods and applications.

Wei, F.; and Nguyen, U. T. 2019. Twitter Bot Detection Using Bidirectional Long Short-term Memory Neural Networks and Word Embeddings. In *2019 First IEEE International Conference on Trust, Privacy and Security in Intelligent Systems and Applications (TPS-ISA)*, 101–109. IEEE.

Wolf, T.; Debut, L.; Sanh, V.; Chaumond, J.; Delangue, C.; Moi, A.; Cistac, P.; Rault, T.; Louf, R.; Funtowicz, M.; Davison, J.; Shleifer, S.; von Platen, P.; Ma, C.; Jernite, Y.; Plu, J.; Xu, C.; Scao, T. L.; Gugger, S.; Drame, M.; Lhoest, Q.; and Rush, A. M. 2020. Transformers: State-of-the-Art Natural Language Processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 38–45. Online: Association for Computational Linguistics.

Yang, K.-C.; Varol, O.; Hui, P.-M.; and Menczer, F. 2020. Scalable and generalizable social bot detection through data selection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 1096–1103.