Diffusion-Generated Social Graphs Enhance Bot Detection

Alec Laprevotte[1*] **Ryan Y. Lin**[2†] **Siddhartha M. Ojha**[2†]
[1]Harvard University [2]Caltech
alaprevotte@g.harvard.edu {rylin, sojha}@caltech.edu

Abstract

Social media bot detection faces persistent data scarcity challenges, as obtaining diverse, high-quality labeled datasets becomes increasingly difficult. We introduce AURA (Augmented User-graph via Reverse-diffusion Architecture), a novel and model-agnostic pipeline that leverages graph diffusion models to generate realistic synthetic social network data for training augmentation. While we demonstrate AURA using GraphMaker, a graph-compatible diffusion architecture, our framework is compatible with any suitable generative model. By combining diffusion-based synthetic graph generation with specialized language models, AURA produces synthetic users enriched with both network structure and textual features. Through systematic evaluation on TwiBot-22 under varying levels of data scarcity, we show that synthetic augmentation via AURA consistently improves bot detection performance, delivering robust gains in accuracy, precision, and recall across all tested sample sizes. This work represents the first application of graph diffusion models to social media bot detection and establishes synthetic data generation as a promising direction for overcoming labeled data scarcity in this domain, with preliminary results suggesting increasing effectiveness as graph generation capabilities scale.

1 Introduction

Social networks have fundamentally transformed information consumption and online interaction, creating new vulnerabilities for malicious manipulation through automated bot accounts. These bots have been implicated in serious threats to democratic processes, including election interference [Bessi and Ferrara, 2016] and coordinated disinformation campaigns [Hajli et al., 2022], making their detection a critical research priority.

The effectiveness of bot detection systems depends heavily on the quantity and diversity of training data. While recent efforts have produced valuable public benchmarks [Cresci et al., 2015, Feng et al., 2021a, 2023], these datasets remain limited in scale, and questions persist regarding their informativeness [Hays et al., 2023]. This data scarcity problem is exacerbated by two key challenges: (1) increasingly sophisticated language model-powered bots that require more diverse training examples to detect, and (2) social media platforms' growing restrictions on data collection that limit researchers' access to real-world samples.

Synthetic data generation offers a promising solution to these constraints. Rather than relying solely on scarce real data, we can augment training sets with realistic synthetic examples that capture the distributional properties of bot and human behaviors. However, generating high-fidelity synthetic

*Work partially completed while a student at Caltech.
†Authors contributed equally and are listed in alphabetical order.

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: New Perspectives in Graph Machine Learning (NPGML).

social network data presents unique challenges, requiring models that can simultaneously capture complex graph structures, realistic user interactions, and contextually appropriate textual content.

We introduce AURA (Augmented User-graph via Reverse-diffusion Architecture), a novel modular pipeline that addresses these challenges through graph diffusion models. Our approach consists of three integrated components: a data sampling module that extracts representative subsets from existing datasets, a graph diffusion-based generator that synthesizes realistic user interaction patterns and network neighborhoods, and specialized text and timestamp generators that produce contextually appropriate user profiles based on bot versus human behavioral patterns. These modules are then combined to generate synthetic user data comprising textual, numerical, and relational information.

Our contributions are threefold:

- **Novel Application:** We introduce the first use of graph diffusion models for synthetic social network generation tailored to bot detection, opening a new direction for data augmentation in this domain.
- **Empirical Validation:** We show that augmentation via our framework improves classifier performance across repeated runs with varying levels of data scarcity, achieving up to 1.77% absolute accuracy gains alongside substantial improvements in precision and recall. Critically, effectiveness increases with sample size, suggesting the AURA pipeline will naturally scale as graph diffusion capabilities continue to advance.
- **Modular, Extensible Framework:** Our model-agnostic design enables systematic exploration of diverse generative architectures and data augmentation strategies, providing a flexible foundation for future research on synthetic graph-based data generation.

The remainder of this paper is organized as follows: Section 2 reviews related work in bot detection and synthetic data generation, Section 4 details our AURA pipeline, Section 5 presents experimental results, and Section 6 discusses implications and future directions.

2 Related Works

2.1 Social Network Bot Detection

Bot detection research emerged alongside Twitter's rise in the late 2000s [Yardi et al., 2009], evolving from rule-based heuristics to sophisticated deep learning approaches. Modern techniques span diverse architectures including random forests [Schnebly and Sengupta, 2019], CNNs [Faerber et al., 2019], and graph neural networks [Zhao et al., 2020], incorporating multi-modal data sources such as user features, tweet content, and network structure [Feng et al., 2023].

Current state-of-the-art methods employ graph neural networks, particularly Graph Convolutional Networks (GCNs)[Kipf and Welling, 2017], which enable learning from both node-level and network-level features[Feng et al., 2021b, Liu et al., 2024]. Recent advances focus on multi-modal integration, with methods like BotRGCN demonstrating superior performance through relational graph convolutional networks on heterogeneous graphs with distinct edge types for follower relationships [Feng et al., 2021b]. The current benchmark standard, TwiBot-22 [Feng et al., 2023], provides over 1 million annotated users and 88 million tweets while preserving crucial graph structure information.

Despite these advances, significant data quality challenges persist. Recent analyses reveal that many pre-TwiBot-22 benchmarks suffer from over-simplistic collection practices, with some datasets solvable using shallow decision trees on trivial features [Hays et al., 2023]. This highlights the fundamental data scarcity problem: existing datasets may not capture the full diversity of bot behaviors in real-world scenarios, motivating synthetic data generation as a complementary approach.

2.2 Diffusion Models

Diffusion models emerged as a powerful class of generative models through the pioneering work of Sohl-Dickstein et al. [2015], who introduced the concept of gradually corrupting data through a forward diffusion process (adding noise) and learning to reverse this corruption (the denoising) for generation. This approach, inspired by non-equilibrium thermodynamics, formulated generation as the reversal of a Markov chain that progressively adds Gaussian noise to data until it becomes

pure noise. While conceptually elegant, this discrete-time formulation required learning the reverse process through variational inference with a carefully designed evidence lower bound (ELBO).

The theoretical landscape was significantly advanced by Song et al. [2021], who unified diffusion models within the framework of stochastic differential equations (SDEs). By reformulating the discrete diffusion process as continuous-time SDEs, they demonstrated that the reverse generative process corresponds to solving a reverse-time SDE driven by the score function—the gradient of the log probability density. This continuous formulation enabled more flexible sampling procedures through predictor-corrector methods and eliminated the need for careful ELBO optimization. Moreover, the score-based method avoids the issue of directly modeling high-dimensional probability distributions, which is often the bottleneck in likelihood-based models.

The field of diffusion models has since exploded with applications spanning diverse domains including image generation [Zhang et al., 2024], audio and video synthesis [Luo et al., 2023], molecular design [Weiss et al., 2023], and text generation [Li et al., 2022]. Recent surveys have documented the rapid expansion of diffusion models across computer vision, natural language processing, medical imaging, time series analysis, text-to-speech synthesis, protein structure prediction/design, drug discovery, and bioinformatics more broadly [Yang et al., 2023, Croitoru et al., 2023, Watson et al., 2023, Corso et al., 2022, Hoogeboom et al., 2022, Yim et al., 2024]. Our work specifically pertains to the application of graph generation with diffusion models, which we discuss further in Section 2.3.

2.3 Diffusion Models for Graphs

Recent work on diffusion models has begun addressing challenges relating to graph-structured data, though most approaches have been limited to synthetic or simplified graph generation tasks. The adaptation of diffusion models from continuous domains (like images) to discrete graph structures presents unique technical challenges, particularly in handling the combinatorial nature of graph topology and the heterogeneity of node attributes.

Rather than using diffusion models for image data, Niu et al. [2020] corrupts real graphs by adding Gaussian noise to all entries of their adjacency matrix, treating the graph structure as a continuous object that can be gradually denoised. Extending this approach, Li et al. [2024] introduces the GraphMaker framework and investigates the scalability of diffusion models to large attributed graphs, adapting continuous diffusion processes to discrete combinatorial structures. However, their work is constrained to handling node attributes through categorical variables and one-hot encoding schemes.

This limitation is particularly problematic for social network data, where user attributes often include rich textual information such as profile descriptions, recent posts, and user-generated content that cannot be adequately captured through simple categorical representations. Moreover, most bot classification frameworks *require* some form of representation of a user's textual data, which introduces the need for fundamentally different processing/generation approaches. In our work, we extend and adapt the GraphMaker framework to generate realistic social network data that includes textual attributes, going beyond the categorical variable limitations of existing approaches. Our method incorporates specialized techniques for handling natural language features, enabling the generation of attributed graphs where nodes contain rich textual information representative of real social media profiles and content.To our knowledge, this is the first application of such methods for augmenting bot detection datasets.

3 Problem Setting

Bot detection on social media platforms like Twitter is fundamentally a graph-based classification problem. As a result, training effective models for this task requires careful consideration of how to represent users, their features, and their relationships within the social graph. We present a general formalization of the training problem. Let $\mathcal{D}_{\text{real}} = (\{(v_i, X_i, y_i)\}_{i=1}^{n}, E_{\text{real}})$ denote a finite, gold-standard dataset of labeled Twitter accounts where for some user account i, v_i is the node in the social graph, x_i is the user feature vector, and $y_i \in \{0, 1\}$ labels the user as either a human (0) or a bot (1), while E_{real} is the direct edge set capturing the underlying graph structure. For any two users $i, j \in \mathcal{D}_{\text{real}}$, $(v_i, v_j) \in E_{\text{real}}$ if and only if user i follows user j. We partition $\mathcal{D}_{\text{real}}$ into a train set, $\mathcal{D}_{\text{real}}^{\text{train}}$, and a held-out test set, $\mathcal{D}_{\text{real}}^{\text{test}}$.

Our objective is to learn a classifier $f_\theta : (v, X) \mapsto \{0, 1\}$ parameterized by θ that predicts the label of a user account from the node v and its features X. Using the gold-standard dataset, we train $f_{\theta,\text{real}}$ on $\mathcal{D}_{\text{real}}^{\text{train}}$ and evaluate its performance on $\mathcal{D}_{\text{real}}^{\text{test}}$. To improve generalization of the classifier, especially when $\mathcal{D}_{\text{real}}$ is limited in scope, we assume access to a black-box generative model $\mathcal{G} : \mathcal{D}_{\text{real}}^{\text{train}} \mapsto \mathcal{D}_{\text{synth}}$, where $\mathcal{D}_{\text{synth}} = (\{(v_j^{\text{synth}}, X_j^{\text{synth}}, y_j^{\text{synth}})\}_{i=1}^{m}, E_{\text{synth}})$, which mimics the format of the real data. We then use this to *augment* the original training data to create $\mathcal{D}_{\text{aug}} = \mathcal{D}_{\text{real}}^{\text{train}} \cup \mathcal{D}_{\text{synth}}$. We then train a second classifier $f_{\theta,\text{aug}}$ on $\mathcal{D}_{\text{aug}}$ and again evaluate on the same held-out test set $\mathcal{D}_{\text{real}}^{\text{test}}$.

4 AURA: Augmented User-graph via Reverse-diffusion Architecture

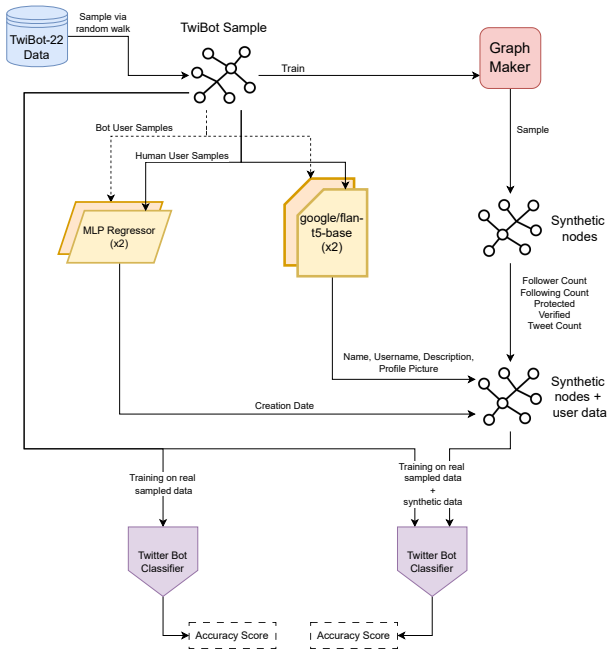

Figure 1: Training Pipeline

Our use of synthetic data generation to augment scarce labeled bot networks draws similarities with techniques employed to improve spam review detection online Stanton and A. Irissappane [2019]. We randomly sample a small number of training nodes from the larger dataset and feed this data into our training pipeline, which is depicted in Figure 1. We defer discussion of the node sampling method used to train our models to Appendix A

4.1 Classifier

Although our pipeline is model-agnostic, we use BotRGCN as the primary classifier for its simplicity and strong performance on TwiBot-22. BotRGCN remains competitive for this task, with recent methods achieving only marginal improvements [Liu et al., 2024]. BotRGCN tackles two key challenges in bot detection: bot disguise (individual bots mimicking humans) and coordinated communities (bots acting in groups). It combines multi-modal user features with relational GCNs to capture both individual and network-level patterns. Textual data (tweets, profiles) are encoded using RoBERTa, while numerical and categorical metadata are integrated directly. A heterogeneous Twitter graph is built from follower/following links, enabling relational GCNs to learn contextualized user representations that fuse content and structure.

4.2 Graph Generation

We use GraphMaker [Li et al., 2024] to generate both structural and categorical social network data. Trained on the sampled bot and human nodes from TwiBot, GraphMaker produces a synthetic

neighborhood of labeled nodes ("bot" or "human") that approximates the real network. It also generates basic user attributes, such as tweet counts and protected/verified status, yielding a set of synthetic users with unique IDs and baseline features. These are then enriched with additional user information via separate generators.

4.3 Training Pipeline

Our training process (Figure 1) consists of two components: (1) synthetic data augmentation and (2) classifier training. Our primary contribution addresses (1) through a two-phase pipeline.

Phase 1: Graph Structure Generation. We preprocess the real social graph to retain categorical features (protected flags, verified flags, and discretized tweet counts) compatible with GraphMaker [Li et al., 2024]. For numerical features like tweet counts, we apply a quantile-based discretization approach: construct an empirical CDF from training data, divide into quantile bins treated as categorical variables during generation, then recover numerical values by sampling from the conditional CDF given the assigned bin. This approach preserves the original distribution characteristics while also enabling categorical treatment during synthesis.

Phase 2: Feature Generation. We train separate models for bot and human data: two `flan-t5-base` [Chung et al., 2022] language models generate names, usernames, profile descriptions, and profile picture links, while two MLP regressors generate creation dates. Each synthetic node receives features from either bot or human models based on its label, completing the synthetic social graph. The synthetic data combines with real training data to form the augmented dataset for classifier training. We do not generate synthetic tweet content (see Appendix B for discussion).

5 Experiments

5.1 Experimental Setup

We evaluate our approach on TwiBot-22 [Feng et al., 2023], the largest publicly available graph-based Twitter bot detection benchmark, containing 1 million users, 4 entity types, and 14 relation types. Each user is labeled as human or bot, with labels obtained via a diversity-aware, weakly supervised process for higher quality than previous datasets. We keep the validation and test sets intact and sample from the 700,000-node training set. While the dataset includes multiple interaction types, we use only follower/following edges to construct our directed social network. From user information, we preserve profile attributes: names, usernames, descriptions, and profile pictures (generated by our language model); tweet counts, follower/following counts, and protected/verified status (generated by GraphMaker); and account creation dates (generated by our MLP).

5.2 Evaluation

We measure classifier performance by testing on the entire TwiBot-22 test split, with 200,000 validation nodes and 100,000 testing nodes. We report standard metrics, including accuracy, precision, recall, F1 score, and AUC. By maintaining the original test split intact and evaluating on the full dataset, we ensure that our results are directly comparable to existing bot detection literature that uses TwiBot-22 as a benchmark. Additionally, since our training involves only small sampled subsets (1k-3k nodes) from the original training data, evaluating against the full 100,000-node test set creates a significant domain gap that effectively tests whether our synthetic augmentation helps models generalize beyond their limited training distribution.

5.3 Results

To systematically evaluate our synthetic data augmentation, we conduct ablation studies across multiple sample sizes and random seeds. We test five sample sizes: 1,000, 2,000, 3,000, 4,000 and 5,000 nodes, comparing training on real data only versus real data augmented with an equal number of synthetic nodes. To account for sampling variability, we run 4 repetitions for each sample range. The average of the results are reported in Table 1 with 95% confidence intervals. Our focus on small sample regimes directly addresses the data scarcity scenarios where synthetic augmentation is most valuable, representing realistic conditions where practitioners have limited labeled data for bot detection.

Table 1: Performance Comparison: Real Data vs. Synthetic Augmentation Across Sample Sizes. Reported errors represent 95% confidence intervals.

Model	Acc	AUC	F1	Precision	Recall
2k Sampled	0.7371 ± 0.004	$\mathbf{0.6007 \pm 0.01}$	$\mathbf{0.3744 \pm 0.04}$	$\mathbf{0.6321 \pm 0.04}$	$\mathbf{0.2661 \pm 0.04}$
+ 2k Synth	$\mathbf{0.7383 \pm 0.003}$	0.5981 ± 0.07	0.3639 ± 0.02	0.6309 ± 0.05	0.2599 ± 0.04
3k Sampled	0.7184 ± 0.003	0.5603 ± 0.01	0.2687 ± 0.03	0.5707 ± 0.03	0.1757 ± 0.03
+ 3k Synth	$\mathbf{0.7357 \pm 0.002}$	$\mathbf{0.5944 \pm 0.06}$	$\mathbf{0.3584 \pm 0.04}$	$\mathbf{0.6282 \pm 0.04}$	$\mathbf{0.2507 \pm 0.03}$
4k Sampled	0.7282 ± 0.003	0.5793 ± 0.01	0.3242 ± 0.03	0.5959 ± 0.03	0.2204 ± 0.03
+ 4k Synth	$\mathbf{0.7388 \pm 0.002}$	$\mathbf{0.6004 \pm 0.05}$	$\mathbf{0.3570 \pm 0.03}$	$\mathbf{0.6310 \pm 0.04}$	$\mathbf{0.2574 \pm 0.03}$
5k Sampled	0.7211 ± 0.002	0.5630 ± 0.02	0.2804 ± 0.02	0.5835 ± 0.02	0.1860 ± 0.03
+ 5k Synth	$\mathbf{0.7388 \pm 0.002}$	$\mathbf{0.5924 \pm 0.04}$	$\mathbf{0.3526 \pm 0.02}$	$\mathbf{0.6276 \pm 0.03}$	$\mathbf{0.2466 \pm 0.02}$

Our ablation studies demonstrate consistent improvements from synthetic data augmentation across different sample sizes. As shown in Table 1, models trained with synthetic augmentation consistently achieve higher test and validation accuracy compared to their real-data-only counterparts.

Across all sample sizes, synthetic augmentation provides systematic improvements in test accuracy, with gains ranging from 0.12% to 1.77%. The improvements become more pronounced as sample size increases, suggesting that synthetic data generation becomes more effective when trained on larger initial samples. Validation accuracy improvements are consistent across all experiments, indicating enhanced generalization to unseen data. The 5k sample results are particularly compelling, showing substantial improvements across nearly all metrics. The synthetic augmentation achieves a 1.77% improvement in test accuracy, along with gains in AUC (0.0294), F1 score (0.0722), precision (0.0441), and recall (0.0606). This clear scaling trend, wherein larger samples yield proportionally stronger benefits, is especially promising given our evaluation against the full TwiBot-22 test set split, where larger training samples can better approximate the true underlying distribution.

The consistency of these improvements across multiple sample sizes and random seeds indicates that the gains are systematic rather than artifacts of particular data samples. While these preliminary results show modest improvements, their reliability and scaling trajectory demonstrate that our diffusion-based synthetic data generation successfully captures meaningful patterns in bot and human behavior. As graph diffusion technology continues to advance and can handle larger attributed graphs, we anticipate AURA will deliver increasingly substantial performance gains.

6 Conclusion

Our preliminary results provide encouraging evidence for the viability of synthetic data augmentation for bot detection, particularly in data-scarce scenarios where improvements are most impactful. The consistency of improvements across all sample sizes and the clear scaling trend, where larger samples yield proportionally stronger benefits, suggest substantial potential as graph diffusion capabilities advance.

Current limitations include scalability: our diffusion model, GraphMaker, struggles with graphs larger than approximately 13k nodes due to memory constraints. However, this constraint is instructive given our evaluation against the full TwiBot-22 test split as larger training samples increasingly approximate the true underlying distribution, making the observed scaling trajectory especially promising.

Our work establishes a foundation for synthetic data generation in social media bot detection that aligns favorably with advancing graph diffusion technology. The modular pipeline design enables systematic improvements across components, from more memory-efficient diffusion architectures to enhanced text generation and distributed sampling strategies. As synthetic data generation continues to advance, we anticipate that the modest improvements demonstrated here represent early evidence of a promising research direction that could significantly impact bot detection in resource-constrained environments.

Future work should focus on scalability optimization, optimal synthetic-to-real data ratios at larger scales, and cross-platform transferability as technological constraints diminish, paving the way for broader adoption and transformative impact in resource-constrained bot detection scenarios.

Acknowledgments and Disclosure of Funding

This research was originally conducted as part of Caltech's CS/EE 145: Projects in Networking course. We thank Prof. Adam Wierman for the conversations and feedback that greatly benefited this work, as well as the Caltech Computing + Mathematical Sciences (CMS) Department for supporting access to necessary computing resources.

References

Alessandro Bessi and Emilio Ferrara. Social bots distort the 2016 u.s. presidential election online discussion. *First Monday*, 21(11), Nov. 2016. doi: 10.5210/fm.v21i11.7090. URL https://firstmonday.org/ojs/index.php/fm/article/view/7090.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. Scaling instruction-finetuned language models, 2022. URL https://arxiv.org/abs/2210.11416.

Gabriele Corso, Hannes Stärk, Bowen Jing, Regina Barzilay, and Tommi Jaakkola. Diffdock: Diffusion steps, twists, and turns for molecular docking. *arXiv preprint arXiv:2210.01776*, 2022.

Stefano Cresci, Roberto Di Pietro, Marinella Petrocchi, Angelo Spognardi, and Maurizio Tesconi. Fame for sale: Efficient detection of fake twitter followers. *Decision Support Systems*, 80:56–71, December 2015. ISSN 0167-9236. doi: 10.1016/j.dss.2015.09.003. URL http://dx.doi.org/10.1016/j.dss.2015.09.003.

Florinel-Alin Croitoru, Vlad Hondru, Radu Tudor Ionescu, and Mubarak Shah. Diffusion models in vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(9): 10850–10869, 2023.

Michael Faerber, Agon Qurdina, and Lule Ahmedi. Identifying twitter bots using a convolutional neural network. 01 2019.

Shangbin Feng, Herun Wan, Ningnan Wang, Jundong Li, and Minnan Luo. Twibot-20: A comprehensive twitter bot detection benchmark. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, CIKM '21, page 4485–4494. ACM, October 2021a. doi: 10.1145/3459637.3482019. URL http://dx.doi.org/10.1145/3459637.3482019.

Shangbin Feng, Herun Wan, Ningnan Wang, and Minnan Luo. Botrgcn: Twitter bot detection with relational graph convolutional networks. In *Proceedings of the 2021 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, ASONAM '21, page 236–239. ACM, November 2021b. doi: 10.1145/3487351.3488336. URL http://dx.doi.org/10.1145/3487351.3488336.

Shangbin Feng, Zhaoxuan Tan, Herun Wan, Ningnan Wang, Zilong Chen, Binchi Zhang, Qinghua Zheng, Wenqian Zhang, Zhenyu Lei, Shujie Yang, Xinshun Feng, Qingyue Zhang, Hongrui Wang, Yuhan Liu, Yuyang Bai, Heng Wang, Zijian Cai, Yanbo Wang, Lijing Zheng, Zihan Ma, Jundong Li, and Minnan Luo. Twibot-22: Towards graph-based twitter bot detection, 2023. URL https://arxiv.org/abs/2206.04564.

Nick Hajli, Usman Saeed, Mina Tajvidi, and Farid Shirazi. Social bots and the spread of disinformation in social media: The challenges of artificial intelligence. *British Journal of Management*, 33(3):1238–1253, 2022. doi: https://doi.org/10.1111/1467-8551.12554. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/1467-8551.12554.

Chris Hays, Zachary Schutzman, Manish Raghavan, Erin Walk, and Philipp Zimmer. Simplistic collection and labeling practices limit the utility of benchmark datasets for twitter bot detection. In *Proceedings of the ACM Web Conference 2023*, WWW '23, page 3660–3669. ACM, April 2023. doi: 10.1145/3543507.3583214. URL http://dx.doi.org/10.1145/3543507.3583214.

Emiel Hoogeboom, Víctor Garcia Satorras, Clément Vignac, and Max Welling. Equivariant diffusion for molecule generation in 3d. *arXiv preprint arXiv:2203.17003*, 2022.

Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks, 2017. URL https://arxiv.org/abs/1609.02907.

Jure Leskovec and Christos Faloutsos. Sampling from large graphs. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '06, page 631–636, New York, NY, USA, 2006. Association for Computing Machinery. ISBN 1595933395. doi: 10.1145/1150402.1150479. URL https://doi.org/10.1145/1150402.1150479.

Mufei Li, Eleonora Kreacic, Vamsi K. Potluru, and Pan Li. Graphmaker: Can diffusion models generate large attributed graphs? *Transactions on Machine Learning Research (TMLR)*, 2024. URL https://openreview.net/forum?id=0q4zjGMKoA.

Rong-Hua Li, Jeffrey Xu Yu, Lu Qin, Rui Mao, and Tan Jin. On random walk based graph sampling. In *2015 IEEE 31st International Conference on Data Engineering*, pages 927–938, 2015. doi: 10.1109/ICDE.2015.7113345.

Xiang Lisa Li, John Thickstun, Ishaan Gulrajani, Percy Liang, and Tatsunori B. Hashimoto. Diffusion-lm improves controllable text generation, 2022. URL https://arxiv.org/abs/2205.14217.

Feng Liu, Zhenyu Li, Chunfang Yang, Daofu Gong, Haoyu Lu, and Fenlin Liu. Segcn: a subgraph encoding based graph convolutional network model for social bot detection. *Scientific Reports*, 14: 4122, February 2024. doi: 10.1038/s41598-024-54809-z. URL https://doi.org/10.1038/s41598-024-54809-z. Open Access.

Simian Luo, Chuanhao Yan, Chenxu Hu, and Hang Zhao. Diff-foley: Synchronized video-to-audio synthesis with latent diffusion models, 2023. URL https://arxiv.org/abs/2306.17203.

Chenhao Niu, Yang Song, Jiaming Song, Shengjia Zhao, Aditya Grover, and Stefano Ermon. Permutation invariant graph generation via score-based generative modeling. In Silvia Chiappa and Roberto Calandra, editors, *Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pages 4474–4484, Palermo, Italy, 26–28 Aug 2020. PMLR. URL https://proceedings.mlr.press/v108/niu20a.html.

Melike Oguz-Alper and Li-Chun Zhang. Snowball sampling from graphs, 2023. URL https://arxiv.org/abs/2003.09467.

Kristina Radivojevic, Nicholas Clark, and Paul Brenner. Llms among us: Generative ai participating in digital discourse, 2024. URL https://arxiv.org/abs/2402.07940.

James Schnebly and Shamik Sengupta. Random forest twitter bot classifier. In *2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC)*, pages 0506–0512, 2019. doi: 10.1109/CCWC.2019.8666593.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 2256–2265, Lille, France, 07–09 Jul 2015. PMLR. URL https://proceedings.mlr.press/v37/sohl-dickstein15.html.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=PxTIG12RRHS.

Gray Stanton and Athirai A. Irissappane. Gans for semi-supervised opinion spam detection. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 5204–5210. International Joint Conferences on Artificial Intelligence Organization, 7 2019. doi: 10.24963/ijcai.2019/723. URL https://doi.org/10.24963/ijcai.2019/723.

Joseph L Watson, David Juergens, Nathaniel R Bennett, Brian L Trippe, Jason Yim, Helen E Eisenach, Woody Ahern, Andrew J Borst, Robert J Ragotte, Lukas F Milles, et al. De novo design of protein structure and function with rfdiffusion. *Nature*, 620(7976):1089–1100, 2023.

T. Weiss, E. Mayo Yanes, S. Chakraborty, et al. Guided diffusion for inverse molecular design. *Nature Computational Science*, 3:873–882, 2023. doi: 10.1038/s43588-023-00532-0. URL `https://doi.org/10.1038/s43588-023-00532-0`.

Ling Yang, Zhilong Zhang, Yang Song, Shenda Hong, Runsheng Xu, Yue Zhao, Wentao Zhang, Bin Cui, and Ming-Hsuan Yang. Diffusion models: a comprehensive survey of methods and applications. *ACM Computing Surveys*, 56(4):1–39, 2023.

Sarita Yardi, Daniel Romero, Grant Schoenebeck, and danah boyd. Detecting spam in a twitter network. *First Monday*, 15(1), Dec. 2009. doi: 10.5210/fm.v15i1.2793. URL `https://firstmonday.org/ojs/index.php/fm/article/view/2793`.

Jason Yim, Brian L Trippe, Valentin De Bortoli, Emile Mathieu, Arnaud Doucet, Regina Barzilay, and Tommi Jaakkola. Diffusion models in protein structure and docking. *WIREs Computational Molecular Science*, page e1711, 2024.

Chenshuang Zhang, Chaoning Zhang, Mengchun Zhang, In So Kweon, and Junmo Kim. Text-to-image diffusion models in generative ai: A survey, 2024. URL `https://arxiv.org/abs/2303.07909`.

Chensu Zhao, Yang Xin, Xuefeng Li, Hongliang Zhu, Yixian Yang, and Yuling Chen. An attention-based graph neural network for spam bot detection in social networks. *Applied Sciences*, 10:8160, 11 2020. doi: 10.3390/app10228160.

A Node Sampling

To train our graph, we randomly select a small number of nodes from the TwiBot-22 dataset. In an effort to produce a subgraph that was representative of the larger training dataset, samples were extracted using multiple sampling techniques (particularly snowball sampling [Oguz-Alper and Zhang, 2023], forest fire sampling [Leskovec and Faloutsos, 2006], and rejection-controlled metropolis-hastings sampling [Li et al., 2015]) such that graph metrics could be computed for both samples and the full graph, and could then be contrasted (with the chosen metrics being avg degree, density, WCC fraction, SCC fraction, reciprocity, assortativity, pagerank and average path length).

Through our testing, we found that rejection-controlled metropolis-hastings samplers yielded metrics that most closely resembled those of the full graph while sampling only 1% of the full graphs nodes. This is because the RCMH algorithm performs a biased random walk on the symmetrized version of our directed social graph, with acceptance probability

$$a(u \to v) = \min\left(1, \left(\frac{\deg(u)}{\deg(v)}\right)^{\alpha}\right)$$

for transitions from node u to candidate v, where $\alpha \in [0, 1]$ controls the degree bias. For social networks like Twitter, this formulation is advantageous because it can capture both highly connected influencers and isolated accounts—both critical populations for bot detection. When the walk becomes stuck (idle for more than max_idle), the algorithm randomly jumps to explore new regions, preventing oversampling of dense communities that might obscure important behavioral patterns in bot classification.

Since TwiBot-22's training corpus exhibits a large fraction of unconnected nodes (approximately 300,000 unconnected nodes in a 700,000 node graph), we chose $\alpha = 0.95$ and max_idle = 100 after empirical validation on graph preservation metrics. The high α value biases exploration toward nodes with diverse connectivity, while the low emax_idle parameter promotes frequent exploration of different graph regions, preventing the sampler from becoming trapped in homogeneous communities and maintaining the heterogeneity essential for robust bot detection model training.

B Why No Tweets?

We find that including tweet data in the training of the BotRGCN classifier yields only marginal improvements in performance. See Table 2 for additional details.

Data Usage	Accuracy	Precision	Recall	F1
All	0.7966	0.7481	0.4680	0.5750
10%	0.7699	0.6657	0.4390	0.5291
None	0.7734	0.6882	0.4214	0.5228

Table 2: Performance of BotRGCN classifier on the test set when including varying amounts of Tweet data in the training corpus. We note the difference in classifier accuracy at these levels is $< 3\%$.

Further, in the years since the release of TwiBot-22, significant advancements in the text generation capabilities of language models have allowed bot accounts to generative more human-like content and made it more challenging to correctly distinguish real and bot account on social networks when using text data [Radivojevic et al., 2024].

C Reproducibility

All experiments and training were performed on P100 and/or V100 GPUs with model and data tensors moved to the appropriate device. Cumulatively, all experiments totaled on the order of 100 GPU hours. We plan to release the experiment code upon publication.

D Licenses For Assets Used

The experiments in this work make use of several open-source libraries, all of which are properly cited and used in accordance with their respective licenses. In particular, `PyTorch`, `scikit-learn`, `NetworkX`, and `pandas` are made available under the BSD 3-Clause license. Deep Graph Library (`DGL`) and `huggingface_hub` are available under the Apache License 2.0.