Interactive Twitter Bot Detection through Network Graph Analysis

Team 018: Jon Womack, Albert Chen, Madelyn Scandlen, Anirudh Prabakaran, Shannon Isaacs*

ABSTRACT

We propose an interactive method for detecting Twitter bots that combines network graph analysis with expert human input. Automated methods for bot detection can be limited in identifying sophisticated bots that imitate human behavior. Our interactive approach uses network graph analysis to identify clusters of accounts with similar behavior, allowing an expert bot moderator to visualize and investigate individual account characteristics. Our novel combination of clustering, visualization, and user interface will accelerate the efforts of a social media to identify bots with high confidence. We evaluate our method using real Twitter data and aim to provide a valuable tool for researchers and social media analysts to understand bot behaviors.

1 BACKGROUND

In conjunction with to the rise of content generating models like ChatGPT [12] and Dalle [13], recent detection methods have utilized characteristics that are less easily subverted by adversarial manipulation, like hand-crafted node features (centrality, authority, and eccentricity) and structural embeddings [1]. In fact, the top performing models on the most comprehensive twitter datasets (Twibot-20 and Twibot-22) are graph-based approaches [4].

Botometer provides a publicly available user interface with multiple bot scores (financial, scammer, etc.) for a given account from a supervised classifier [17]. Similarly, we will have a user interface that displays relevant account information such bot-language likelihood and a profile overview.

However, rather than automate bot detection, our approach enables human annotation. A method similar to our approach is Torralba et. al scaling instance annotation by clustering segmentation masks and then propagating human assigned labels to the rest of the cluster [11]. To improve upon the random selection within a cluster to the moderator, our tool will give the moderator a choice in sample selection aided by a T-SNE embedding visualization [14].

Abbaspour and Moghaddam suggest a novel friendship preference feature, which can scale with growing followers [9]. The method also uses screen names, descriptions, counts of followers and following, and profile images. Ho et. al adopt a similar approach utilizing LSTM units to process metadata features and generate bot probabilities [8].

Another approach focuses on using the textual content of Tweets as input to deep-learning models to identify Twitter bots [3]. Wei and Nguyen improved on this deep by developing a long short-term memory model (LSTM) on sequential word-embeddings of a single Tweet to classify a user as bot [16].

The most successful deep-learning approach for bot detection uses word-embeddings and user metadata as input to a neural network that can predict the likelihood a user is a bot [10] [2]. Deep Profile-based Bot (DeeProBot) detection framework employs GLoVe word embeddings and user metadata to learn representations of users [8], which inspires our work to generate a bot-score for users which can then be used to create similar user clusters.

Using the graphical structure of accounts has proven to be a useful feature to further advance bot-detection. BotRGCN demonstrates that a relational graph convolutional network with encoded user data can detect bots [6]. A Graph Attention Network incorporates an attention mechanism to Graph Neural Networks, which allows it to better discern the significance of neighboring users when aggregating information [15]. These methods are an end-to-end bot detection framework, but it lacks the flexibility of incorporating human judgment.

2 INNOVATION

Our approach leverages content and metadata of a user's tweets and human-in-the-loop flow to classify users as bots at a large scale. Using the textual content of user's tweets with the user metadata as input to a Convolutional Neural Network, a bot-score will be calculated for all users. Users are then clustered by their graphical structure and bot score, where a human

*All authors contributed equally to this work.

annotator can view a few users in a cluster and decide whether the entire cluster is likely to consist of all bots.

We plan to use *streamlit* for the front end, and *networkx* for storing the graph structure. Depending on scaling and flexibility constraints, we may build the graph with *d3.js*. T-SNE will be the dimensionality reduction method for a moderator to easily interpret. We will leverage cloud resources to run jobs for compute intensive tasks such as embedding generation and clustering.

3 MOTIVATION

Ferrare et al. note that social bots' influence on public opinion via social media has been a concern since the early 2000s [7]. Social bots, which imitate human activity at high speeds, are tough to identify. Bots are used for malicious activity such as spam, phishing, propaganda, suppressing dissent, and infiltrating networks, which interferes with the purpose of social media platforms that prioritize authentic human interaction.

Building a successful tool will provide a solution to the problem of social media bot moderation. Our tool, which labels clusters of accounts instead of individuals, will enable moderators to be several times more effective. We will measure success by evaluating the ratio of individual accounts viewed by moderators to the number of accurately labeled bot accounts.

4 COSTS AND RISKS

There are several potential risks and challenges associated with this work. Network graph analysis and interactive visualization require significant computing resources due to the large volumes of data involved. Our project's analysis is sensitive to data quality and quantity, and we must ensure a balanced dataset with both bot and genuine accounts to prevent class imbalance issues.

A successful implementation of this work could have many payoffs. Our interactive, human-in-the-loop bot detection tool is a unique and potentially first-of-its-kind open-source solution. Our project contributes to a better understanding of bot behaviors, and the code and tools developed can be shared with the research community for further advancement. Our project may require cloud services to handle high memory requirements. Publishing our interactive tool as a webapp may entail deployment and maintenance costs.

5 EVALUATION

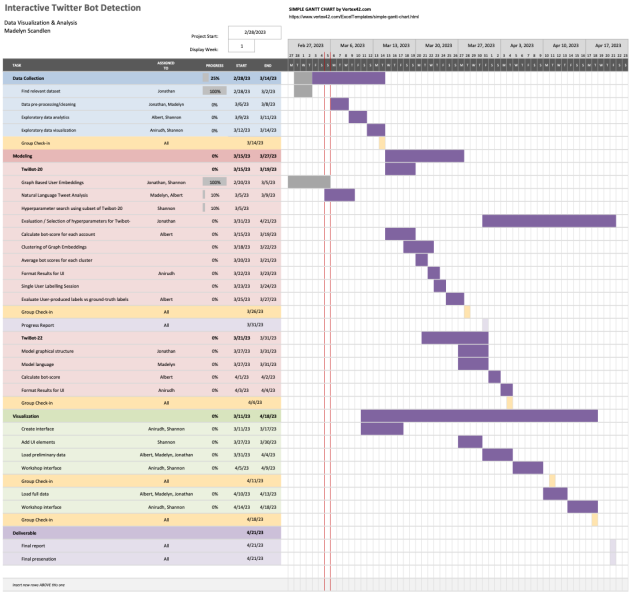
The progress report will provide updates on the modeling process of a subset of the Twibot-20 dataset and the development of the UI interface [5]. The clustering model will output a general bot score for the likelihood that all users in the cluster are bots. Then, the user of the interface will label whether the cluster seems like a bot, and this will be compared against the ground-truth labels in the Twibot-20 dataset.

The final exam deliverable will show the completed UI interface on the outputs of the modeling of the graphical structure, tweet textual content, and tweet metadata on the recent Twibot-22 data [4].

6 PLAN OF WORK

Each member will work in a different sector of the project, with the three major divisions being: modeling graphical structure (Jonathan, Shannon), modeling tweet content and metadata (Albert, Madelyn), and visualizing the users in a graph using the outputs of the models (Anirudh).

The plan of activities is in the Gantt chart below.



An overview of the flow of the task is to work on data pre-processing from March 4-14, modeling on the Twibot-20 set from March 14-29, modeling on the Twibot-22 set from March 30-April 4, and interface creation from March 11-April 18. For the project proposal, all team members have contributed a similar amount of effort.

REFERENCES

[1] Ashkan Dehghan, Kinga Siuta, Agata Skorupka, Akshat Dubey, Andrei Betlen, David Miller, Wei Xu, Bogumil Kaminski, and Pawel Pralat. 2022. Detecting Bots in Social-Networks Using Node and Structural Embeddings. (2022).

[2] David Dukić, Dominik Kea, and Dominik Stipic. 2020. Are You Human? Detecting Bots on Twitter Using BERT. *2020 IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA)* (2020), 631–636.

[3] Michael Färber, Agon Qurdina, and Lule Ahmedi. 2019. Identifying Twitter Bots Using a Convolutional Neural Network. In *Conference and Labs of the Evaluation Forum*.

[4] Shangbin Feng, Zhaoxuan Tan, Herun Wan, Ningnan Wang, Zilong Chen, Binchi Zhang, Qinghua Zheng, Wenqian Zhang, Zhenyu Lei, Shujie Yang, et al. 2022. TwiBot-22: Towards graph-based Twitter bot detection. *arXiv preprint arXiv:2206.04564* (2022).

[5] Shangbin Feng, Herun Wan, Ningnan Wang, Jundong Li, and Minnan Luo. 2021. Twibot-20: A comprehensive twitter bot detection benchmark. (2021), 4485–4494.

[6] Shangbin Feng, Herun Wan, Ningnan Wang, and Minnan Luo. 2022. 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* (Virtual Event, Netherlands) *(ASONAM '21)*. Association for Computing Machinery, New York, NY, USA, 236–239.

[7] Emilio Ferrara. 2018. Measuring Social Spam and the Effect of Bots on Information Diffusion in Social Media. *Computational Social Sciences* 10 (2018), 229–255.

[8] Kadhim Hayawi, Sujith Samuel Mathew, Neethu Venugopal, Mohammad Mehedy Masud, and Pin-Han Ho. 2022. DeeProBot: a hybrid deep neural network model for social bot detection based on user profile data. *Social Network Analysis and Mining* 12 (2022).

[9] Samaneh Hosseini Moghaddam and Maghsoud Abbaspour. 2022. Friendship Preference: Scalable and Robust Category of Features for Social Bot Detection. *IEEE Transactions on Dependable and Secure Computing* (2022), 1–1. https://doi.org/10.1109/TDSC.2022.3159007

[10] Sneha Kudugunta and Emilio Ferrara. 2018. Deep Neural Networks for Bot Detection. *ArXiv* abs/1802.04289 (2018).

[11] Dim P Papadopoulos, Ethan Weber, and Antonio Torralba. 2021. Scaling up instance annotation via label propagation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 15364–15373.

[12] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. In *Proceedings of the 2018 Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. 1–15.

[13] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. 2021. DALL·E 2: Exploring Cross-modal Embeddings for Image Generation. arXiv:2110.13147 [cs.CV]

[14] Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of machine learning research* 9, 11 (2008).

[15] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903* (2017).

[16] Feng Wei and Uyen Trang Nguyen. 2019. Twitter Bot Detection Using Bidirectional Long Short-Term Memory Neural Networks and Word Embeddings. *2019 First IEEE International Conference on Trust, Privacy and Security in Intelligent Systems and Applications (TPS-ISA)* (2019), 101–109.

[17] Kai-Cheng Yang, Emilio Ferrara, and Filippo Menczer. 2022. Botometer 101: Social bot practicum for computational social scientists. *Journal of Computational Social Science* (2022), 1–18.