# Twitter-Bot detection using Graph Convolution Networks

Manan Gangar
School of Information Technology
Deakin University
Burwood, Melbourne
mgangar@deakin.edu.au

Wei Luo
School of Information Technology
Deakin University
Burwood, Melbourne
wei.luo@deakin.edu.au

**Abstract**

**Twitter bots are automated accounts that pose a significant threat to the integrity of online social networks. Detecting bots is challenging due to their sophisticated nature of mimicking human behavior and evade the traditional detection methods. The aim of this report is to approach the problem using graph convolution networks (GCN) to enhance the accuracy of Twitter bot detection. Further, the report tries to find the distinguishing features of bot accounts from non-bot accounts. The graph structure of the data will help in understanding the relationships between the accounts which is a traditional method cannot do. The TwiBot-22 dataset is used to evaluate the method. Our results demonstrate that the GCN-based approach outperforms traditional methods in detecting bots accurately.**

*Keywords:* Twitter bot detection, Graph neural networks, TwiBot-22 data

## I. Introduction

The rise of social media has enabled individuals and organizations to connect and engage with each other in unprecedented ways. Twitter for instance, is recognized as the most renowned social networking platform for microblogging, which is primarily used for promptly receiving updates and breaking news [1].

However, the popularity of social media platforms has also made them a prime target for malicious actors, such as bots. Twitter, in particular, has been plagued by bots that disseminate propaganda, spam users, and manipulate public opinion.

Different research works have proposed various methods [3], [2] for detecting bots, which differ based on the definition of bot accounts, the selection of features that represent accounts, and the machine-learning algorithms used for classification. However, these methods will have limitations in identifying sophisticated bots that mimic human behavior. Machine learning techniques have shown promising results in detecting bots accurately, prompting the adoption of graph convolution networks (GCN) for Twitter bot detection.

## II. Dataset

In recent years, the detection of Twitter bots has become increasingly important, and various models have been developed based on traditional methods and graph-based models. However, the efficacy of these models has been hampered by poor dataset quality, including limited dataset scale, incomplete graph structure, and low-quality annotations [4]. This report utilizes the TwiBot-22 dataset, which is the most extensive and inclusive benchmark for detecting Twitter bots to date. Its primary objective is to address the challenges faced by previous datasets.

The TwiBot-22 dataset contains information on one million users, out of which approximately 14% are bots and the remainder are non-bots. The dataset includes metadata such as creation date, location, name, description, etc for each user as well as additional information on tweets, hashtags, and lists. However, for the purposes of this report, only the user information and their connections, including accounts followed and accounts following, will be utilized.

## III. Exploratory Data Analysis

Since the data was very huge, to start the report, the first 2000 rows were selected. The below EDA was performed on the data.

The first plot is a histogram which shows the frequency distribution of days since an account was created. The x-axis shows the number of days, and the y-axis shows the number of accounts created since those many days. Further, the histogram has been colored based on whether the account is a bot account or a human account. Looking at figure 1, we can see that majority of the bot accounts have been created recently, and human accounts are spread accross the histogram.

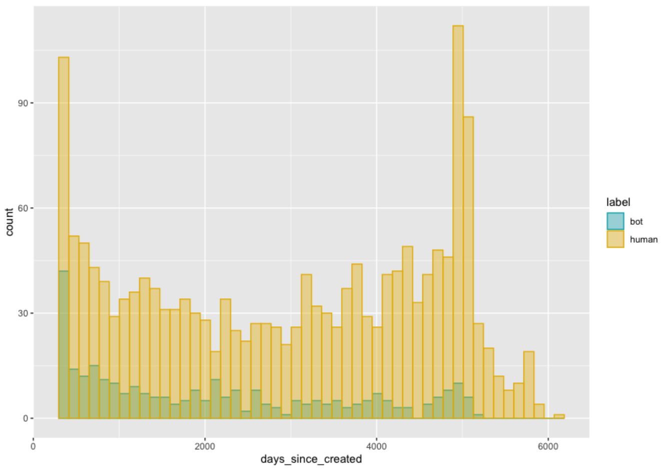


Figure 1: Histogram showing the frequency of the days it has been since the account was created

Next, the figure 2 is a point graph of tweet count vs following count. The figure shows that majority of cluster shows that accounts that follow less accounts have a high tweet count.

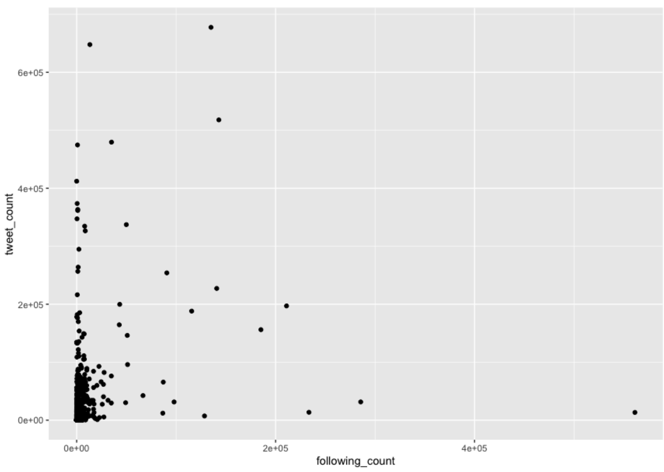


Figure 2: Point graph showing tweet count vs following count

Lastly, figure 3 shows the tweet count vs followers count. This shows how many accounts a user follows, vs how many tweets they have. Here a few outliers that are on the bottom left of the graph are non-bot accounts as they belong to celebrities as these accounts have a high number of followers, but not as many tweets.

## IV. Methodology

The process of the model formulation is as follows:

1) **Data Preprocessing:** The data comes in json format. But, for the graph neural network, the data needs to be in a graphical format. For this, the json data is first converted to csv data. Then, for the model to perform well the data is normalized. And then using the NetworkX and Torch Geometric packages the csv data is formatted to a graph data.

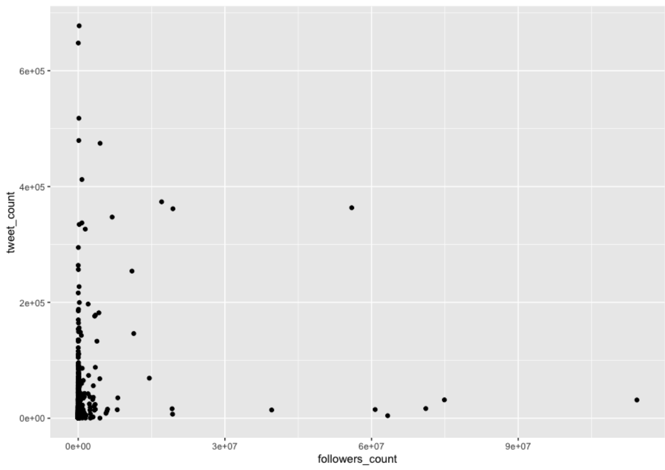


Figure 3: Point graph showing tweet count vs followers count

2) **Splitting data:** In Graph Neural Networks (GNN), a training and testing mask is created instead of simply splitting the data into training and testing sets because GNNs operate on graphs, which have complex structures and dependencies between nodes. Splitting the data randomly into training and testing sets can cause information leakage between the two sets, especially if the graph has a small number of nodes or if the nodes are highly connected. By using a training training and testing mask, the GNN ensures that the structure of the graph is preserved during training and the model learns to generalize to unseen data. Overall, using a training and testing mask is an effective way to ensure that the GNN model is trained on a representative sample of the graph and can generalize to new, unseen data. For this report, the graph is splitted as 80 : 10 : 10 for training, validation and testing mask respectively.

3) **Graph Convolution Network Training:** The GCN model inherits from the PyTorch nn.Module class and contains three GCNConv layers for message passing between nodes in the graph. The first GCNConv layer takes in the number of node features and outputs 16 features. The second and third GCNConv layers have 16 input and output features.

The forward method performs message passing between nodes in the graph using the three GCNConv layers with ReLU activation functions. The output of the final GCNConv layer is returned as the predicted node labels.

The training loop then iterates through each epoch and performs forward and backward propagation of the model on the input graph data. The loss and accuracy values for each training, validation, and testing set are computed and stored in their respective lists.

## V. Results

Along with the GCN model, decision tree and logistic regression models were also created for baseline results. The decision tree model performed with an accuracy of 80% and the logistic regression moel performed with an accuracy of 84%. Whereas, the GCN model performed with an accuracy of more than 90%. Figure 4 and Figure 5 show two graphs showing the train, validtion and test loss and train, validation and test accuracy respectively.

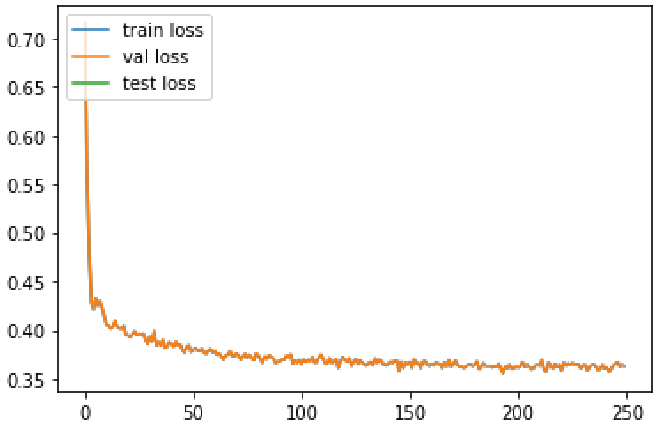


Figure 4: Point graph showing tweet count vs followers count

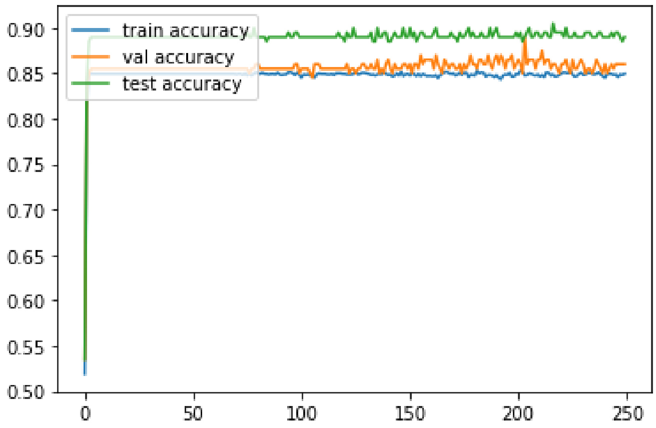


Figure 5: Point graph showing tweet count vs followers count

The graph shows that after around 100 epochs, the loss is significantly dropped, whereas just nearly after 25 epochs, the test accuracy reaches near 90%

## VI. Conclusion

As the results show from above, the GCN model outperforms the other models. The GCN model can hence be used to identify the bots more accurately.

Moreover, the model can be further tuned with respect to the hidden layers, optimization parameters and the loss function. Along with tuning these parameters, the complete data can be used to help increase the accuracy even more.

## Reflection

As a participant of the ADR Summer Project, I wish to reflect on my experience and the valuable knowledge and skills I gained during my time in the program. Firstly, I had the privilege to learn about graph neural networks, including their structure, applications, and limitations. This area of study was of great interest to me, and I was grateful to have the opportunity to apply this knowledge to a real-world problem.

Moreover, I learned how to write research reports in a professional and concise manner. This included understanding the structure of research reports and the importance of presenting findings in a logical and coherent manner. This skill will be of immense value in my academic and future career endeavors.

Additionally, I performed Exploratory Data Analysis. This involved exploring and visualizing data to identify patterns, trends, and outliers. This skill has opened up new possibilities for me to analyze data and gain insights that were previously unavailable.

Finally, I gained valuable experience working on a research project. Collaborating with my peers and supervising professor enabled me to understand the importance of teamwork, communication, and time management in a research setting. This experience provided me with the necessary skills and confidence to undertake future research projects.

Overall, my participation in this research opportunity was a valuable experience that enhanced my academic and professional skills. I am grateful for the opportunity.

## References

[1] M. Aljabri, R. Zagrouba, A. Shaahid, et al. Machine learning-based social media bot detection: a comprehensive literature review. *Social Network Analysis and Mining*, 13(1):20, 2023. doi: 10.1007/s13278-022-01020-5.

[2] E. Alothali, N. Zaki, E. A. Mohamed, and H. Alashwal. Detecting social bots on twitter: A literature review. In *2018 International Conference on Innovations in Information Technology (IIT)*, pages 175–180, 2018. doi: 10.1109/INNOVATIONS.2018.8605995.

[3] P. G. Efthimion, S. Payne, and N. Proferes. Supervised machine learning bot detection techniques to identify social twitter bots. *SMU Data Science Review*, 1(2):Article 5, 2018.

[4] S. Feng, Z. Tan, H. Wan, N. Wang, Z. Chen, B. Zhang, Q. Zheng, W. Zhang, Z. Lei, S. Yang, X. Feng, Q. Zhang, H. Wang, Y. Liu, Y. Bai, H. Wang, Z. Cai, Y. Wang, L. Zheng, Z. Ma, J. Li, and M. Luo. Twibot-22: Towards graph-based twitter bot detection, 2022. URL `https://arxiv.org/abs/2206.04564`.