



פרויקט מס' BS-CS-26-143

Identification and impact of Sockpuppets in Wikipedia

זיהוי והשלכות של בובות קש בוויקיפדיה

פרויקט מחקרי

דוח סופי

הוכן לשם השלמת הדרישות לקבלת

תואר ראשון במדעי הטבע B. Sc.

מאת

לידור לוטטי, חיים בורטמן

בהנחיית ד"ר דביר רוס

הוגש למחלקה למדעי המחשב

המכללה האקדמית להנדסה סמי שמעון

אשדוד



תאריך עברי

תאריך לועזי

י"ב באלול התשפ"ו

25/08/2026

Identification and impact of Sockpuppets in Wikipedia

זיהוי והשלכות של בובות קש בוויקיפדיה

פרויקט גמר

דוח מסכם

הוכן לשם השלמת הדרישות לקבלת

תואר ראשון במדעי הטבע B. Sc.

מאת

לידור לוטטי, חיים בורטמן

הוגש למחלקה למדעי המחשב

המכללה האקדמית להנדסה סמי שמעון

אשדוד

חתימת סטודנט 1: _____ תאריך: _____

חתימת סטודנט 2: _____ תאריך: _____

חתימת המנחה: _____ תאריך: _____

אישור ועדת הפרויקטים: _____ תאריך: _____

Table of Content:

Abstract	5
1. Introduction	6
2. Literature Review	8
2.1 Background	8
2.1.1 NetworkX	8
2.1.2 MediaWiki API	8
2.1.3 BeautifulSoup	8
2.1.4 In Degree	9
2.1.5 Out Degree	9
2.1.6 PageRank	9
2.1.7 Degree Centrality	10



2.1.8 Clustering Coefficient	11
2.1.9 Reciprocity	11
2.1.10 Density	12
2.1.11 Diameter	12
2.1.12 Betweenness Centrality	
	13
2.1.13 Higher Order Markov Chains	
	13
2.2 Related Work	14
2.2.1 Natural Language Processing Models	
	14
2.2.2 CheckUsers	14
2.2.3 Machine Learning bots and Tools	
	14
2.2.4 Sockpuppet Detection using Subgraph Similarity	
Matching	15



3. Methodology	16
3.1 Web-Scraping	16
3.2 Parsing Discussions	16
3.3 Discussion Page HTML Research	
_____	17
3.4 Refactoring Scraping by Wikipedia Guidelines	
_____	17
3.5 Algorithm Completion	17
3.6 Data Extraction and Preprocessing Improvements	
_____	18
4. Results	19
4.1 Data Preparation	19
4.2 Group Definitions	19
4.3 Elaboration on the Calculations Made	
_____	19
4.4 Visualization of the Most Significant Metrics	
_____	21
4.5 Findings	29
4.6 Individual User-Level Metrics	31
5. Potential Explanations	40
6. References	41



תקציר הפרויקט:

מטרת הפרויקט היא לבדוק אם קיים הבדל וויזואלי ו/או אימפרי בין דיונים של ויקיפדיה אשר כוללים "בובות קש", כאשר "בובות קש" הוא שם המתאר חשבונות נוספים של משתמשים, שבעזרתם ניתן להשפיע בצורה לא הוגנת על תוכן הערכים באתר, בין אם זאת לקדם אג'נדה מסויימת של מפעיל ה"בובות קש" ואז פגיעה כללית בתוכן האתר. אנו נבדוק את ההבדלים ע"י המרת הדיונים לגרפים והשוואת מדדים בין גרפים הכוללים משתמשי "בובות קש" וגרפים בלעדיהם. הפרויקט שואף לחקור אם המרת הדיונים לגרפים תוביל לדרך ברורה יותר לזהות "בובות קש", ולמנוע את פגיעתם בוויקיפדיה. ולשמור על טוהר ונאמנות המידע.

Abstract:

The goal of the project is to research the differences between discussions that contain "Sock Puppets" which are alternative accounts of users in Wikipedia, amongst other sites on the internet. "Sock Puppet" accounts can be used to manipulate the contents of Wikipedia, whether it is for pushing an agenda of the puppet's user or to generally damage the contents of the encyclopedia [1, 2].

This project will focus on transforming these discussions into graphs, then we will use multiple metrics to try and discern a difference between graphs originating from discussions that contain "Sock Puppets", and graphs originating from regular discussions. The goal is to empirically determine if these graphs can be distinguished from one another, and thus to be used to identify discussions that possess "Sock Puppets".

1. Introduction:

The popular internet encyclopedia known as "Wikipedia" is commonly used as a source of information for many people in their everyday lives and is based on the philosophy of total open access, where instead of a regulating



body, the editors and contributors of Wikipedia are regular people who add to the encyclopedia and moderate its contents [3]. The encyclopedia has been widely successful among students as well, so much so that many are favoring Wikipedia over established class materials [6].

Unfortunately, some users have exploited the openness of the platform to add misinformation to the website, from harmful and disingenuous “information” that damages the integrity of the encyclopedia [2, 5], to pushing political agendas by changing wording and lessening the severity about world events, as well as enforcing racial stereotypes [4].

One method of manipulating information on Wikipedia and other websites involves *sockpuppetry*, which is the practice of creating multiple accounts separate from the user’s main account [7]. A malicious user can employ sockpuppets to deceive others by acting as if these puppets are multiple people. They can then vandalize the website whilst keeping the user’s main account safe. They can also be used to over-represent an opinion, by having the main account or another sockpuppet account post an opinion in a discussion and then having the sockpuppets reinforce this viewpoint. These opinions can be false information, which from an outside perspective seem to be the consensus of numerous people, when in fact it is a single individual from many different accounts [2].

As of today, sockpuppet detection in Wikipedia is mostly based on manual detection by having a special group of users called “CheckUsers”, these users have some permissions that allow them to check and compare the IP address of two suspected sockpuppets. The sites moderators also rely on their experience with recognizing the behaviors of suspected users, and by

comparing writing styles [8]. However, not only is IP comparison a breach of privacy (since it could reveal the real-world location of the suspected “sockpuppet”), but it is also unreliable. Some Internet Service Providers have a single IP address shared across multiple people. If the suspected user is not a sockpuppet yet shares the same ISP as the other suspected sockpuppet, the IP addresses can match and lead to a false positive [9].

Besides manual detection there exists some machine learning tools, such as ClueBot NG [10] and STiki [11] as well as other proposed tools [12], which can identify vandalism. Although they cannot reveal sockpuppetry by itself, investigating multiple reports of vandalism detected by these tools can result in discovering malicious sockpuppetry. However, as previously stated, they are not designed to detect sockpuppetry, and thus still demand human investigation.

As we can see, the lack of an algorithm or a system that detects sockpuppets can result in their users manipulating information for a long time.

Developing a tool to detect sockpuppets would reduce demands from human moderators, in addition to discovering them earlier. There have been many proposed ideas for sockpuppet detection models, mainly in the form of Natural Language Processing (NLP) models [7, 8, 14].

From our review of these papers, the models can detect multiple differences between interactions that lack sockpuppets, and interactions that contain them, although these differences could be as small as one-tenth of a percent and lead to imperfect predictions. For example, when one model attempted to detect if an account is a sockpuppet with no further context, the models ROC AUC metric was 0.68 (where 0.5 is considered a “random guess” and 1.0

is a “perfect score”). The same model found much greater success in detecting if two potential sockpuppets belong to the same user, resulting in an ROC AUC metric of 0.91 [7].

We wish to highlight our reasons as to why we have decided to explore another avenue in detecting sockpuppets, which is through graph theory. We believe that due to NLP models’ reliance on textual analysis, they may overlook structural differences in these discussions we could discover using graph theory.

Turning a conversation into a directed graph has already been done numerous times [15, 16, 17]. These graphs can also be used by machine learning and deep learning models to train them [18]. There is also evidence that shows that graphs can be used to create specialized AI models [13]. We believe that there is potential in using graph theory in sockpuppet detection in Wikipedia, by turning discussions into graphs, and then comparing the graphs containing sockpuppets and graphs without them empirically using established metrics. This paper investigates whether sockpuppet detection in Wikipedia can be improved using graph theory, by analyzing Wikipedia discussions using graph-specific metrics.

2. Literature Review

2.1. Background:



We will examine some of the tools and concepts we will use to create and analyze the graphs in this project. We will require a way to interface with Wikipedia discussions, which we will sample. And a way to turn this discussion into a graph. Afterwards, we will examine the graph by calculating metrics and using them to prove or disprove our hypotheses. The metrics shown in this background will be included in our project but by no means are the only metrics which will be included.

2.1.1 **NetworkX**

NetworkX is a library for the programming language python which is used for creating and analyzing graphs. It supports many types of graphs including the directed and weighted graphs which we will use. It also supports efficient algorithms for performing analysis on those graphs, which allows us to easily calculate the metrics which we will use to empirically differentiate between sockpuppet graphs and “clean” graphs. These metrics include PageRank and Degree Centrality amongst others which will be expanded upon later [19, 20].

2.1.2 **MediaWiki API**

MediaWiki API is an open-source software developed to allow programming access to Wikipedia, including editing and querying the website. Using this API, we can write code that requests data from Wikipedia, such as the discussions we need to convert to graphs. This saves us from performing Web-scraping, which is taking the discussion

pages source code and filtering the HTML via other tools such as regular expressions [21].

2.1.3 BeautifulSoup

Beautiful Soup is a Python library commonly used in web scraping to parse and navigate HTML documents retrieved from websites. It allows us to extract structured information in HTML metadata by searching through the attributes of the code. We use BeautifulSoup to scrape Wikipedia's discussion pages, as well as the page which contained the known sockpuppet masters and their puppets. This allows us to perform operations that the MediaWiki API does not support.

2.1.4 In Degree

Measures how many incoming edges a node has; indicates how many other nodes point to it. We suspect that a suspicious account with unusually high in-degree may receive disproportionate support from coordinated accounts.

Formula:

$$d_i(v) = \sum_{u \in V} A_{uv}$$

Variables:

- $d_i(v)$: in-degree of node v
- A_{uv} : adjacency value from u to v
- V : set of all nodes



2.1.5 Out Degree

Measures how many outgoing edges a node has, sockpuppet accounts may exhibit high out-degree by repeatedly interacting with many target accounts.

Formula:

$$d_{out}(v) = \sum_{u \in V} A_{vu}$$

Variables:

- $d_{out}(v)$: out-degree of node v
- A_{vu} : adjacency value from v to u
- V : set of all nodes

2.1.6 PageRank

PageRank is an algorithm originally developed by Google that used to rank web pages by importance in the search engine. It measures the relative importance of nodes in a network using a simulated “random walk” for a user browsing on the website, calculated by first-order Markov chains. It includes following links found on websites and incorporates random teleportation, to simulate a user looking up a different topic. The result was used as a score which ranks the web page and was used to decide whether to place the web page on the front page of Google [22]. We can apply this algorithm to rank specific nodes in our graphs. We want to see if sockpuppets form their own “network” and thus can become relatively more important than actual users, or if the effect is reversed. NetworkX already possesses functionality for calculating



PageRank, so integrating it into our graphs simply requires calling the existing method.

Formula:

$$PR(v) = \frac{1-d}{N} + d \sum_{u \in B_v} \frac{PR(u)}{L(u)}$$

Variables:

- $PR(v)$: PageRank of node v
- d : damping factor
- N : total nodes
- B_v : nodes linking to v
- $L(u)$: outgoing links of u

2.1.7 Degree Centrality

Degree Centrality reflects the importance of a node based on the number of direct connections it has to other nodes. Unlike PageRank, which takes the entire graph into consideration, Degree Centrality is local and only considers directly connected nodes. Since the graphs we will make are directed, we will have two measurements under degree centrality. In-degree, which refers to how many nodes connect to this node, and out-degree, which refers to how many other nodes this node connects to [23]. This metric can allow us to spot users who only interact with others or ones that are only interacted with, which we would suspect is a characteristic of sockpuppets.



Formula:

$$C_D(v) = \frac{d(v)}{n-1}$$

Variables:

- $C_D(v)$: degree centrality
- $d(v)$: total degree of node
- n : number of nodes

2.1.8 Clustering Coefficient

The Clustering Coefficient is a measure that details the extent of which nodes are clustered. For example, if node A and node B are connected to node C, the clustering coefficient indicates the likelihood that A and B are also connected. This is a way to measure if a graph contains many interconnected groups, or more sparsely connected [24]. This metric will assist in determining whether graphs containing sockpuppets have them interact in their own isolated environment, instead of interacting with other users organically.

Formula:

$$C(v) = \frac{2e_v}{k_v(k_v-1)}$$

Variables:

- $C(v)$: clustering coefficient
- e_v : edges between neighbors of v
- k_v : number of neighbors of v



2.1.9 Reciprocity

Reciprocity is a measure of the proportion of edges that are bidirectional in a directed graph. Since our graph is directed, it is a useful metric which could indicate accounts interacting excessively with each other, though it could also show how common discussions which have two individuals directly going back and forth in regular discussions and discussions with sockpuppets.

Formula:

$$r = \frac{L_{\leftrightarrow}}{L}$$

Variables:

- r : reciprocity
- $L_{\leftrightarrow}$: bidirectional edges
- L : total edges

2.1.10 Density

Density is a measure of how many edges exist in the graph relative to the maximum possible number of edges. A high-density graph indicates thorough discussion between all users, while a low-density graph could indicate that only a few people interact with one another.



Formula:

$$D = \frac{m}{n(n-1)}$$

Variables:

- D : density
- m : number of edges
- n : number of nodes

2.1.11 **Diameter**

Diameter measures the longest shortest path between any two nodes in the graph. A graph that is dominated by sockpuppets could have small diameters due to tight coordination between them, though it could yield additional insights between infected graphs and clean ones.

Formula:

$$diam(G) = \max_{u,v \in V} d(u,v)$$

Variables:

- $diam(G)$: graph diameter
- $d(u,v)$: shortest path between u,v
- V : set of nodes



2.1.12 Betweenness Centrality

Betweenness Centrality is used to measure how often a certain node lies on the shortest path between other nodes. We want to discover if sockpuppets are “placed” in a discussion in a certain way that they interact with as many users as possible, or if they are put on the sidelines.

Formula:

$$C_B(v) = \sum_{s \neq v \neq t} \frac{\sigma_{st}(v)}{\sigma_{st}}$$

Variables:

- $C_B(v)$: betweenness centrality
- σ_{st} : shortest paths from s to t
- $\sigma_{st}(v)$: those passing through v

2.1.13 Higher Order Markov Chains

Higher-order Markov chains are a conceptual model for analyzing sequential dependence. First-order Markov chains are used to calculate the probability of landing on a specific state based solely on the previous state. Higher-order Markov chains consider the previous k states, where k refers to the order of the chain [27]. PageRank uses first-order Markov chains for its calculations. We believe we should incorporate higher-order Markov chains as an additional metric, due to our desire to glean further insights into structural patterns formed by sockpuppets.

2.2. Related Work:

In this section we will explore other methods of detecting sockpuppets including NLP models and the regular process of detection currently employed at Wikipedia.

2.2.1 Natural Language Processing Models

Natural language processing (NLP) models are models made specifically for interpreting or generating text [25]. In sockpuppet detection, there exists several attempts to create a functional model, some with more classic NLP approaches [7, 8], and others using a deep learning approach with LSTM [14]. To this day Wikipedia has yet to adopt a model specifically for sockpuppet detection. We believe that although these models have shown successful results, they could miss potential nuances that come from analyzing the structure of the conversation itself, rather than textual analysis.

2.2.2 CheckUsers

CheckUsers are Wikipedia moderators that are given special permissions that allow them to check a suspected user's IP address, as well as their browser version and operating system. These privileges are maintained solely to combat the damages malicious users can cause using their anonymity, which includes sockpuppetry [26]. These users use their experience to detect sockpuppetry by comparing writing styles and behaviours with other suspected puppets. They can also be called upon when an investigation determines a high likelihood for a user to be a puppet and or a puppeteer by other moderators [8].

2.2.3 Machine learning Bots and Tools

Wikipedia moderators employ a few machine learning tools and bots to detect vandalism in several edits. These tools use machine learning rather than heuristic algorithms that detect basic vandalism such as poor edits that are due to insufficient research and citations or simply baseless. One of these bots is ClueBot NG, an upgrade to the previous bot ClueBot, with the difference being that ClueBot NG uses machine learning and Artificial Neural Networks. According to the bots' owners, ClueBot NG catches approximately 55% of all vandalism. Although these tools are not designed to detect sockpuppetry specifically, they have been able to point towards specific problematic users which have employed sockpuppetry. This way an investigation can be issued and CheckUsers can determine sockpuppetry using their special privileges [10, 11].



2.2.4 Sockpuppet Detection using Subgraph Similarity Matching

One approach we found that uses graph-based sockpuppet detection combined machine learning and subgraph comparison. The study attempted to detect sockpuppets on a Chinese microblogging website known as Sina Weibo. They created an algorithm which generated two directed, weighted subgraphs by analyzing the social networks of a pair consisting of a sockpuppet and a suspected user. They found that a pair of related sockpuppets shared an average of 83 identical neighbors in their social networks, whereas a sockpuppet-ordinary user pair shared only 7 identical neighbors. This suggested that sockpuppet masters aim to recover previous social relationships already established with other puppets.

Another subgraph was made by analyzing the sockpuppets' interest network. The results showed that the interest network of a pair of related sockpuppets contained almost twice as much similarity than sockpuppet-ordinary user pairs (0.65 and 0.34 respectively).

Lastly, related sockpuppet pairs were considerably more likely to follow each other and build a community compared to sockpuppet-ordinary user pairs (31% and 4% respectively). The results of the study showed that utilizing social network structure similarity is highly effective in detecting sockpuppets [28].

We believe that the discussions on Wikipedia could be similar in structure to the social networks found on other websites, including the one in the

study. Since a malicious sockpuppet user typically pushes their own specific agenda, a sockpuppet group would always support each other in similar topics. From this, we can infer that sockpuppet groups could have a similar network to the interest network in the study.

3. Methodology:

Upon completion of the original literature review, we have already possessed an algorithm which used MediaWiki API to gather the discussions a certain user has been a part of, this means that we already had the basis of gathering the discussions of discovered sockpuppets. However, we did not have the capacity to generate a graph or parse the Wikipedia discussions, neither did we have a method of classification in the graph.

3.1 Web-Scraping

Our work began with scraping the Wikipedia discussion page that archives confirmed instances of "sockpuppets" and their respective "puppetmasters". Although our initial methodology indicated an API-based approach using MediaWiki, we have transitioned into working mostly using BeautifulSoup, since MediaWiki lacks parsing capabilities.



After scraping the archive we managed to construct a list of confirmed sockpuppets and their puppet-masters, though this list was filled with Wikipedia tags and other “garbage” data. Thus, we have employed filters and removed all non-username content from the list, whilst turning it into a dictionary. After filtering, this dictionary was composed from the puppet-masters username as a key, and a list of their puppets usernames as values, this dictionary was also reversed later, to allow each puppet to serve as a key and their master as the value. This was done so that classification could be performed in $O(1)$ time average case, to save computation and runtime.

3.2 Parsing Discussions

After building the basis for our classification function, we have begun to work on the algorithm that forms discussions into graphs. We decided that building the graph should be done using pairs of message authors and their recipients, which the algorithm will transform into graph edges. Specifically, we assumed that the level of indentation used for paragraphing comments can indicate the hierarchy of discussion posts such that a higher indentation level implies being directly replied to the previous post having a lower indentation. However, it was quickly discovered that in large discussions, upon reaching a sufficient level of indentation, the indentation would reset. As a result, the level of indentation cannot be treated as a reliable indication of message authors and their recipients.

3.3 Discussion Page HTML Research



After researching the structure of Wikipedia's HTML code, we discovered that each discussion is segmented into individual sections. In these sections, each message was headered by a span "data-mw-comment-start". Each of these contained an attribute called an "id" which contained metadata about who left the comment and either the recipient, or the discussion itself if the comment was made to no specific user. This proved to be the basis of extracting the graph structure and solved the unreliability of indentation.

3.4 Refactoring Scraping by Wikipedia Guidelines

During our scraping operations, we encountered automated rate-limiting from Wikipedia's servers. This prompted us to examine the Wikipedia scraping guidelines. We found that bots had to contain proper identification of their purpose, and since up until now we have used basic headers, we have breached this guideline.

As a result, we have constructed an accurate header identifying the scraper correctly, we have also made sure to go over our scraping algorithms and improve their efficiency, as well as enforcing delays and limits in data gathering to ensure that they comply with the guidelines, as well as to not send an overwhelming number of requests to the server.

3.5 Algorithm Completion

We have eventually settled on an algorithm which segments Wikipedia talk pages into discussions using the HTML sections, after which we split each discussion based on the span "data-mw-comment-start", and obtained the data necessary to create the graph from the "id" attribute metadata

Using a regex, which is a method of identifying regular expressions in text, we could separate usernames of the message author and the message recipient into usable strings. We then took these usernames as a pair of “replier” and “replied” and sent them to a function which uses NetworkX to construct a graph. When a node is constructed, it is also passed into a function which uses the previously mentioned sockpuppet dictionaries to classify it as either a puppetmaster, a puppet or a user (as in regular user).

Putting all these functions together, we have a working algorithm which correctly builds graphs based on given Wikipedia discussion pages.

3.6 Data Extraction and Preprocessing Improvements

Our initial approach assumed that Wikipedia discussion pages had a consistent HTML structure. However, whilst running the full scraping algorithm on the contributions we have gathered, we encountered issues with our parsing algorithm. We discovered that comment timestamps had an additional format in the past that we did not account for. To compensate, we have updated our parsing function to account for the second version of the timestamp.

In addition, we implemented additional filter functions that remove duplicate graphs we may have gathered whilst scraping overlapping user conversations. This ensured that any calculations we made on the dataset won't be harmed by repetitions of the same discussion.



4. Results:

4.1 Data Preparation



Before computing any metrics, we corrected the parsing inconsistencies described in Section 3.6 and removed duplicate graphs from the dataset, ensuring that the analysis below was not affected by repeated discussions. The resulting graphs were then divided into three groups for comparison.

4.2 Group Definitions

As we wanted to assess the differences between different types of discussions, we have split our dataset into 3 different groups:

- **Uninfected** - Discussions that don't contain either puppet-masters or puppets themselves.
- **Infected** - Discussions that contain at least one confirmed sock-puppet. (Note: these discussions do not have to contain the masters.)
- **Masters** - Discussions that contain at least one confirmed puppet-master, but no puppets.

By these definitions, the three groups should be distinct.

To thoroughly assess the differences in structural aspects among the discussion groups, we calculated descriptive and inferential statistics on each group. This allowed us to examine the overall network features and reveal any reoccurring patterns in each group. For our analysis, we calculated the means, medians, interquartile ranges, and the standard deviation of each metric we discussed in the background section of this paper, as well as other custom metrics we will introduce should we showcase them.

4.3 Elaboration on the Calculations Made



When a metric is calculated based on interactions with the entire graph (e.g. Reciprocity, Density etc.), we simply took the result according to NetworkX or the formula we listed in the background section of the paper.

When a metric is calculated on a specific user (e.g. In degree, out degree, Betweenness Centrality etc.), we calculated the average value of all the given nodes on the graph and used that as the graph's metric value.

It is important to note for the sake of authenticity that some metrics are correlated, we will list the metrics that contain correlations and how they are correlated:

- **In-degree and Out-degree** are perfectly correlated and produce identical values for every graph. Each directed edge adds exactly one to a node's in-degree and one to another node's out-degree, so the sum of all in-degrees always equals the sum of all out-degrees (both equal the number of edges). Because each graph's metric is the average over its nodes, the average in-degree and the average out-degree are mathematically identical. This is reflected in the descriptive statistics, where the two metrics share the exact same mean, standard deviation, median and IQR in every group (e.g. a mean of 1.577 for infected graphs).
- **Density and Degree Centrality** are perfectly correlated; the average degree centrality is always exactly twice the density. NetworkX defines a node's degree centrality as its total degree divided by $(n - 1)$, so averaging this over all n nodes yields $\frac{2m}{n(n-1)}$, which is precisely double the graph density $\frac{m}{n(n-1)}$. The numbers confirm this exact 2:1 ratio in every group (for example, infected graphs have a mean density of 0.260 and a mean



degree centrality of 0.520), and the two metrics return identical Cohen's d values in every comparison.

- **PageRank, k-order Markov probabilities and connection-graph**

PageRank are all perfectly correlated with one another. Each is a stationary distribution score whose values sum to 1 across the nodes of a graph, so once the score is averaged over a graph's n nodes, each metric reduces to exactly $1/n$. As a result, they carry identical information and behave purely as an inverse measure of discussion size rather than as a per-node importance score. The data shows this clearly, with all three reporting the same mean within each group (0.139 for infected, 0.147 for uninfected and 0.134 for masters graphs).

Lastly, we performed two statistical analyses comparisons between all 3 groups:

- **Mann-Whitney U** – Compares two independent groups and determines whether one group tends to have larger values than the other. It works by ranking all observations from both groups and calculates how often the values of the first group rank higher than the values of the second. The test returns a U-value and a p-value. The U-value reflects how separated the ranks of the two groups are, while the p-value indicates how likely it is to observe a difference at least this large if there were no difference between the groups. The result is commonly considered statistically significant when the p-value is below 0.05.
- **Cohen's d** - Measures the magnitude of the difference between two groups. Larger absolute values indicate a stronger effect, while values near zero indicate little practical difference between the groups. A



negative value simply means that the group that is compared to is larger than the main group.

4.4 Visualization of the Most Significant Metrics

To complement the numerical statistics, we plotted the distribution of each metric across the three groups using two complementary views. The first is a violin plot, in which the horizontal axis is the metric value and the width of each shape shows how many graphs fall at that value (the three dashed lines mark the lower quartile, median and upper quartile). The second is an empirical cumulative distribution function (ECDF), in which a curve positioned further to the left indicates that the group tends toward lower values of the metric.

A clear pattern emerges from both the Mann-Whitney U tests and Cohen's d values: the strongest and most consistent separation is not between Infected and Uninfected discussions, but between the Masters group and the Uninfected group. In that comparison every metric reached statistical significance with very small p-values, whereas the Infected-vs-Uninfected comparison was largely non-significant. It is equally important to note that, although these differences are statistically significant, all of the effect sizes are small (Cohen's d roughly between 0.10 and 0.32 in absolute value), meaning the groups overlap heavily and the differences are subtle in practice. The metrics below are presented in descending order of effect size, which corresponds to the metrics that most strongly distinguish the groups.

Closeness Centrality (strongest effect: Cohen's $d = -0.31$, $p \approx 1.3 \times 10^{-19}$ for Masters vs. Uninfected)

Closeness centrality produced the largest effect size of all metrics. The violin plot shows that the Masters distribution is shifted toward lower values, with

its median sitting clearly to the left of the Uninfected median, while the Infected group falls in between. The ECDF makes this ordering unmistakable: the Masters curve (blue) rises first, followed by the Infected curve (green), with the Uninfected curve (red) consistently furthest to the right. In other words, the average participant in a master-containing discussion is, on average, slightly less central and slightly further (in path length) from the rest of the participants than in a clean discussion. This is consistent with the idea that discussions involving puppet-masters are structured marginally differently, though the heavy overlap of the curves shows the effect is modest.

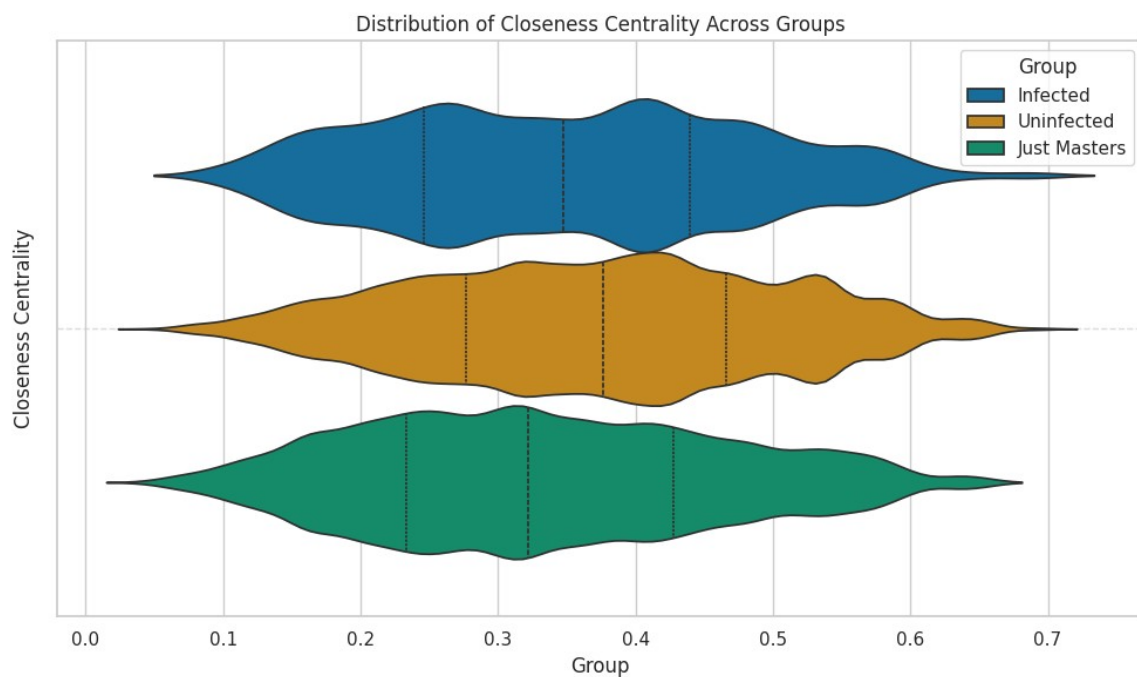


Figure 1a. Violin plot of closeness centrality across the three groups.

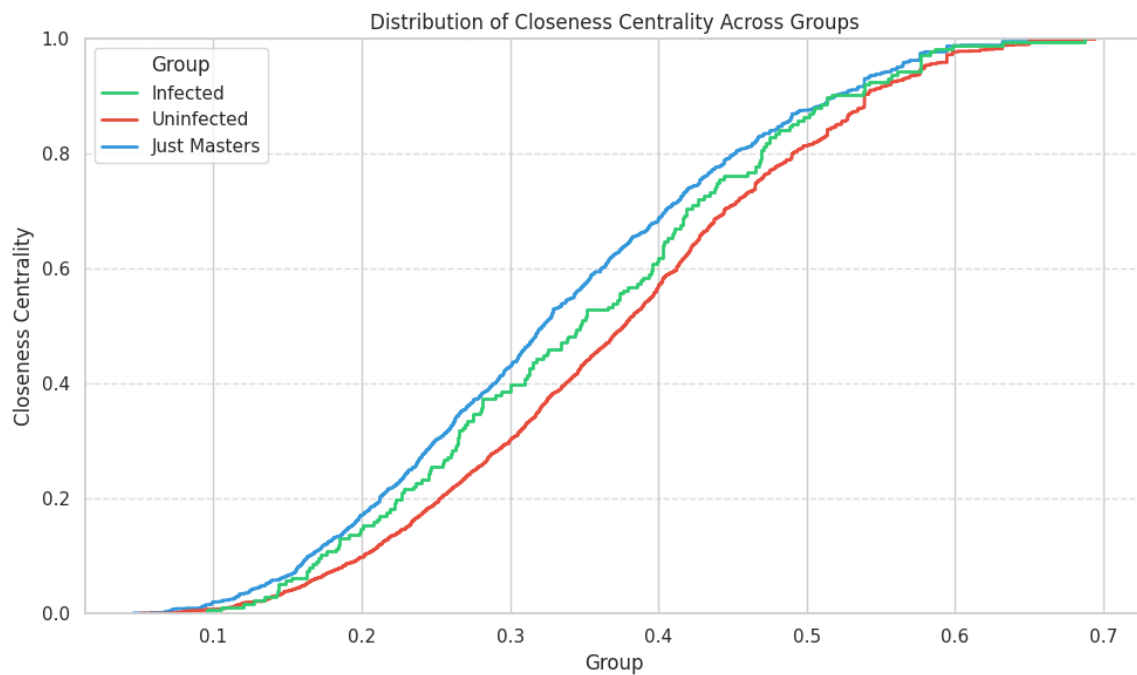


Figure 1b. ECDF of closeness centrality across the three groups.

Reciprocity (Cohen's $d = -0.30$, $p \approx 6.7 \times 10^{-18}$ for Masters vs. Uninfected)

Reciprocity measures the proportion of edges that are mutual. Both plots show the Masters group leaning toward lower reciprocity than the Uninfected group, with the Infected group again sitting close to the Masters group. In the ECDF the Masters curve (blue) is the leftmost across most of the range, indicating that master-containing discussions have proportionally fewer back-and-forth exchanges. A small spike of graphs with zero reciprocity is visible at the far left for all groups, but it is most pronounced for the Masters group. This hints that exchanges in those discussions are

somewhat more one-directional, although, as with every metric here, the distributions overlap substantially.

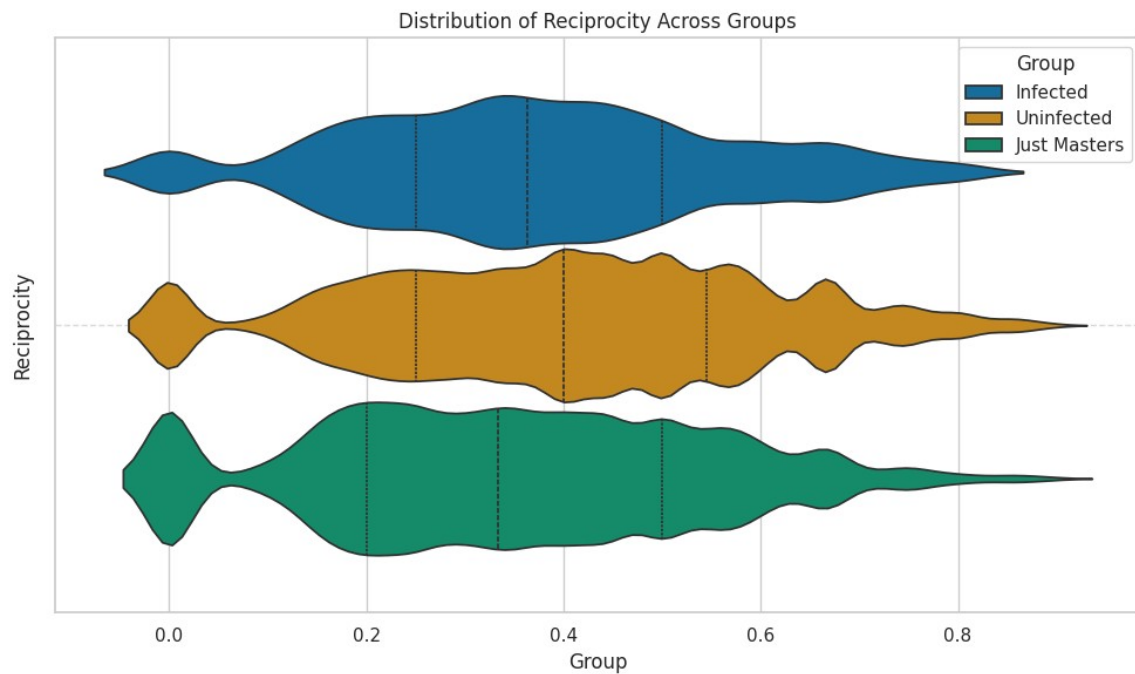


Figure 2a. Violin plot of reciprocity across the three groups.

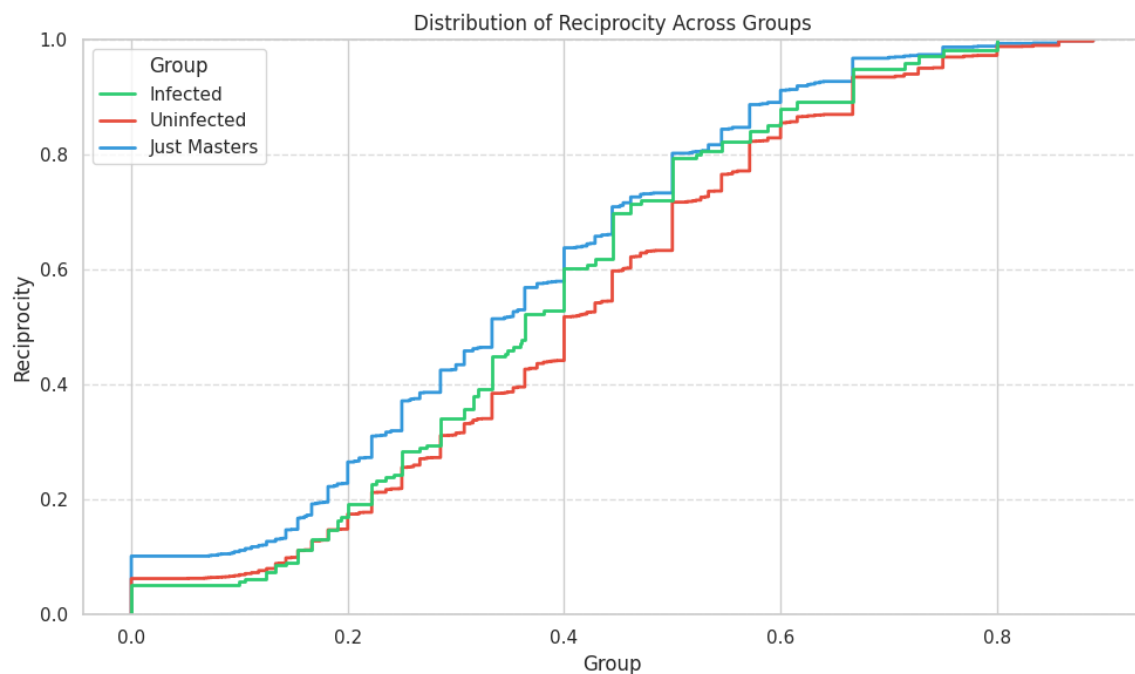


Figure 2b. ECDF of reciprocity across the three groups.

Density / Degree Centrality (Cohen's $d = -0.27$, $p \approx 7.2 \times 10^{-18}$ for Masters vs. Uninfected)



Because degree centrality is exactly twice the density (see the correlation note above), the two metrics tell the same story and only the density plots are shown. The violin plot shows the Masters distribution concentrated at slightly lower density values, and the ECDF confirms it: the Masters curve (blue) rises earliest, the Infected curve (green) is close behind, and the Uninfected curve (red) trails to the right. Lower density means that, relative to the number of possible connections, master-containing discussions contain slightly fewer actual interactions between participants, pointing to marginally sparser conversation structures.

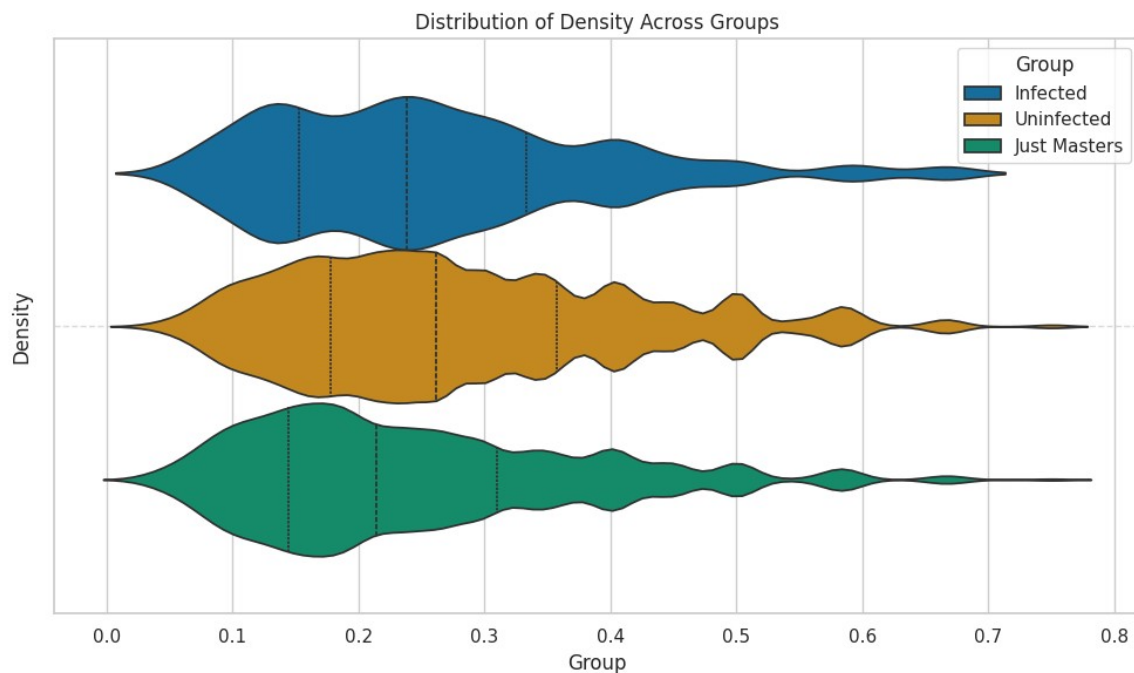


Figure 3a. Violin plot of density (degree centrality is exactly twice these values).

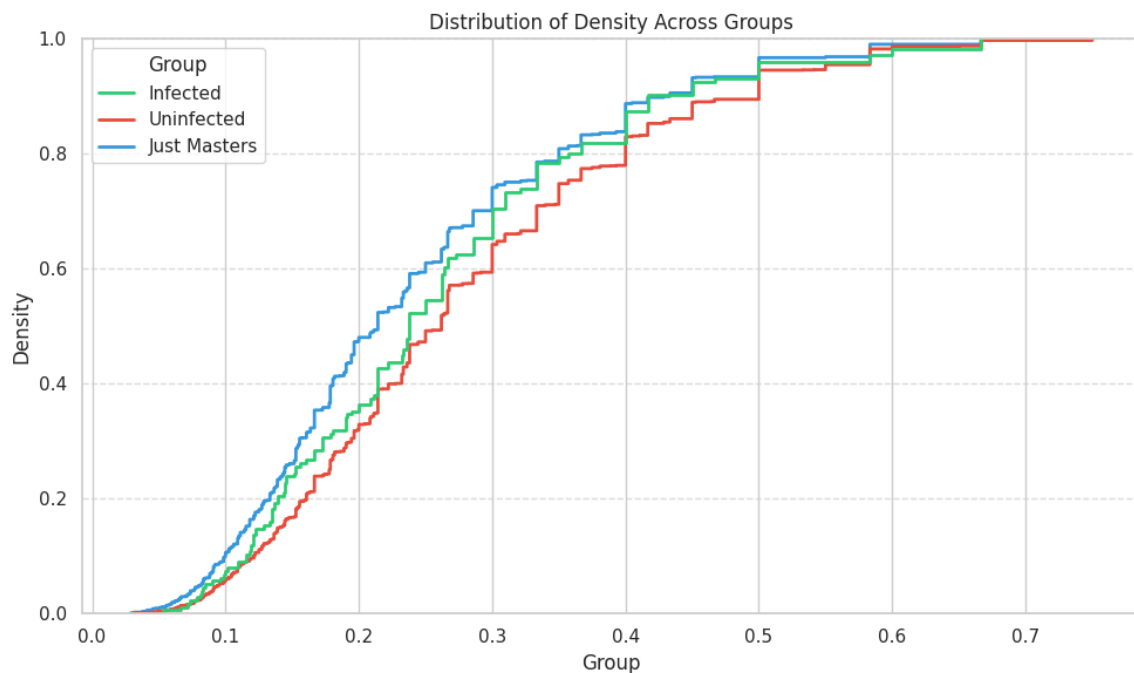


Figure 3b. ECDF of density across the three groups.

Eigenvector Centrality (Cohen's $d = -0.27$, $p \approx 1.0 \times 10^{-13}$ for Masters vs. Uninfected)

Eigenvector centrality rewards nodes that are connected to other well-connected nodes. The violin plot shows the Masters group shifted toward lower values, and the ECDF shows the now-familiar ordering, with the Masters (blue) and Infected (green) curves leading the Uninfected curve (red). The interpretation aligns with the density and closeness results: participants in master-containing discussions are embedded in slightly less interconnected neighborhoods than participants in clean discussions.

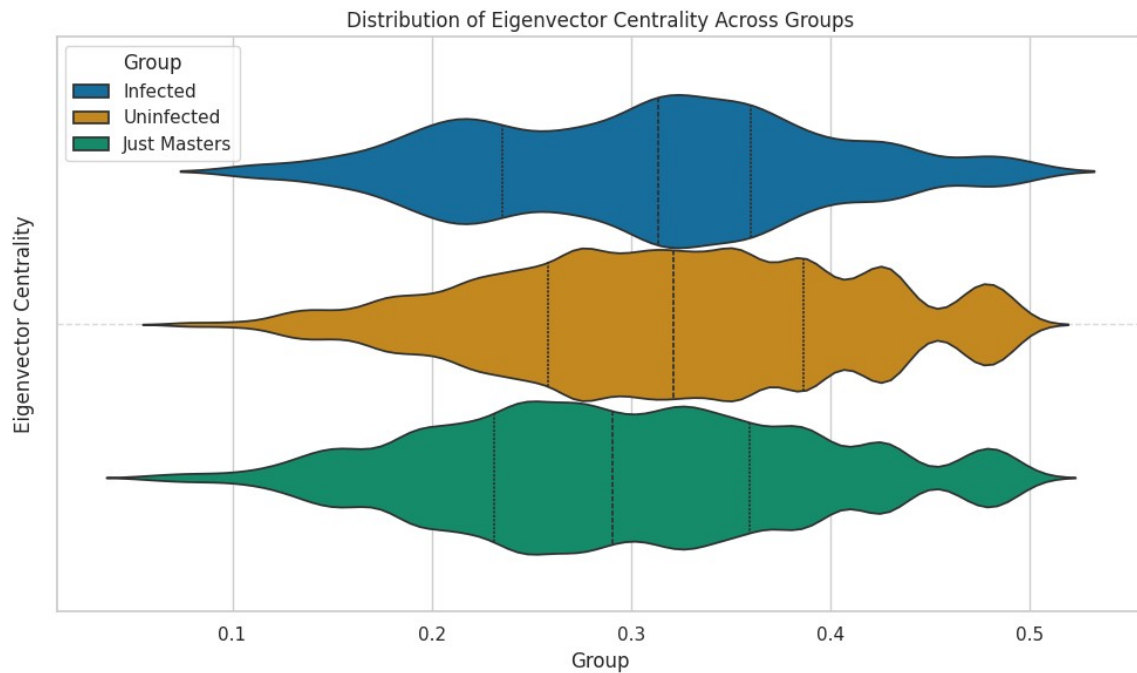


Figure 4a. Violin plot of eigenvector centrality across the three groups.

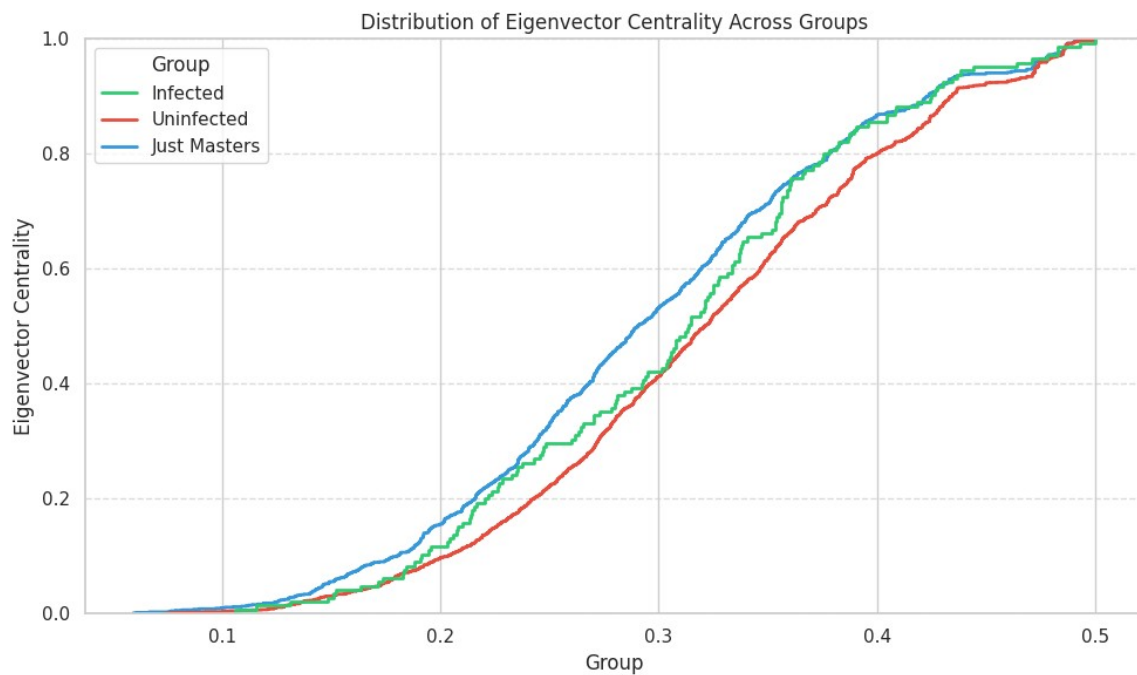


Figure 4b. ECDF of eigenvector centrality across the three groups.

PageRank (Cohen's $d = -0.25$, $p \approx 2.5 \times 10^{-12}$ for Masters vs. Uninfected)

As noted in the correlation analysis, the averaged PageRank, k-order Markov probability and connection-graph PageRank are all equal to $1/n$ and therefore act as an inverse measure of discussion size; only the PageRank

plots are shown as the others are identical. The ECDF shows the Masters curve (blue) shifted toward lower average PageRank, which (because the value equals $1/n$) means master-containing discussions tend to involve slightly more participants than clean ones. The step-like shape of these curves reflects the small, discrete number of participants per discussion. This metric should therefore be read as a proxy for conversation size rather than for any per-user importance.

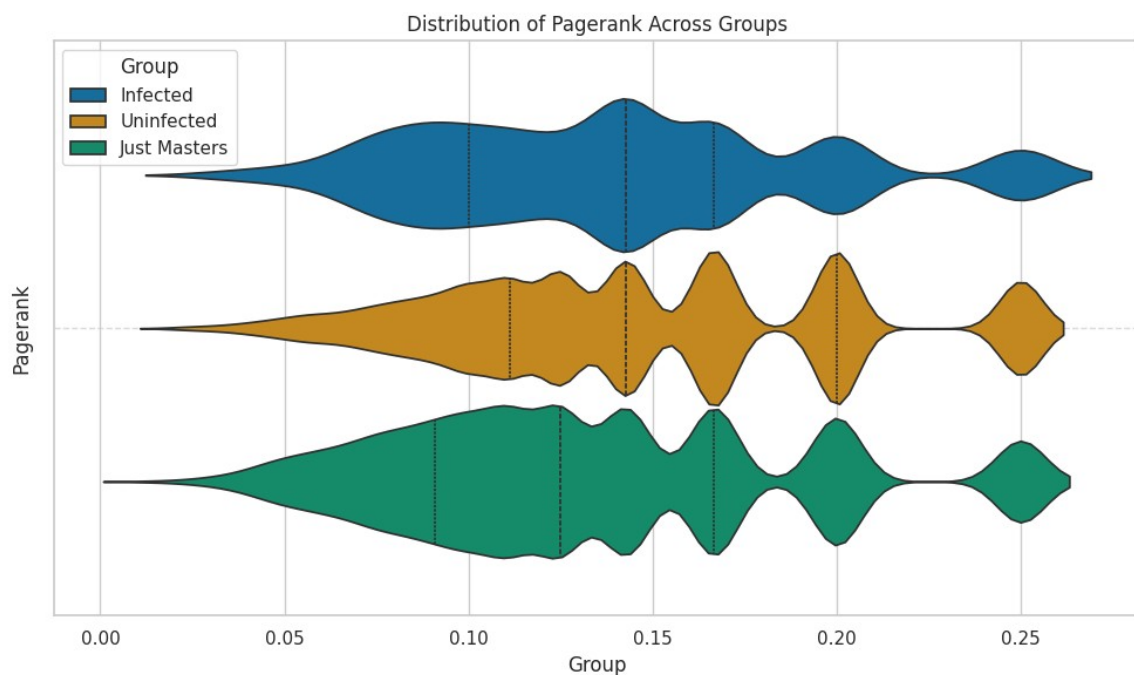


Figure 5a. Violin plot of PageRank (k -order Markov and connection-graph PageRank are identical).

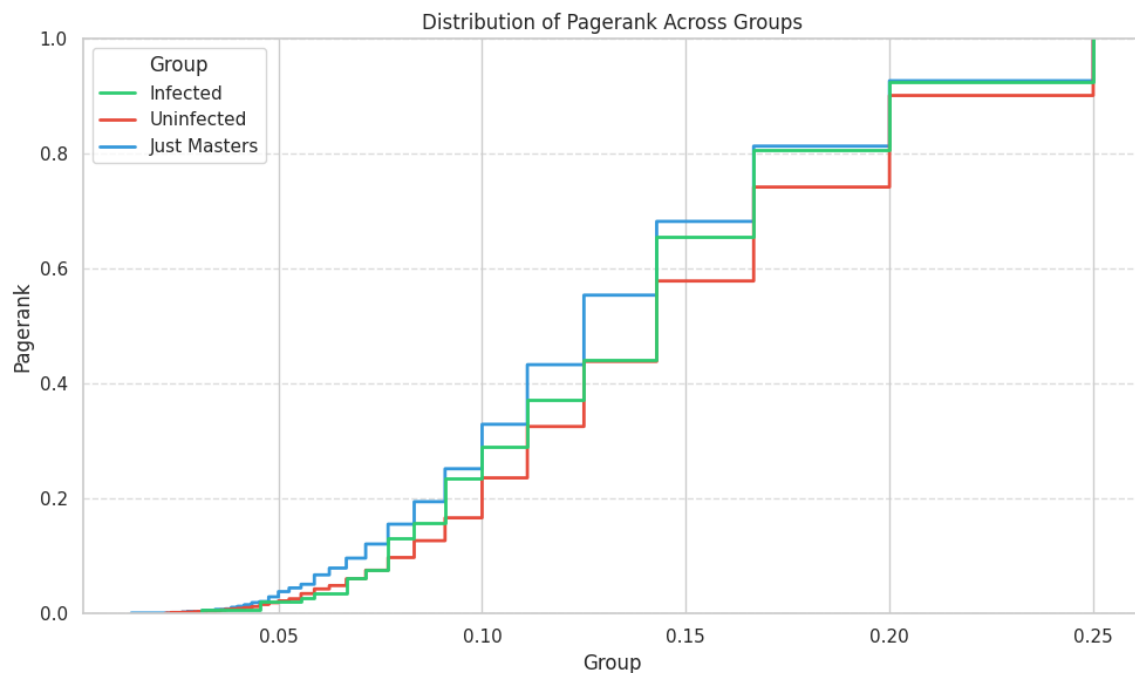


Figure 5b. ECDF of PageRank across the three groups.

4.5 Findings

By examining the results outlined previously, we have drawn the following conclusions.

Our central hypothesis was that graphs derived from discussions containing sockpuppets could be empirically distinguished from graphs derived from clean discussions. The direct test of this hypothesis – the comparison between the Infected and Uninfected groups – did not support it. Of all the metrics examined, only closeness centrality reached statistical significance

($p \approx 0.018$), and even there the effect size was very small (Cohen's $d \approx -0.19$). For every other metric the difference between discussions that contained confirmed puppets and those that did not was statistically indistinguishable. Based on the averaged graph metrics used here, the mere presence of a sockpuppet does not produce a reliably detectable change in the structure of a discussion.

The clearest and most consistent result instead emerged from the comparison between the Masters group and the Uninfected group. Here every metric reached statistical significance, with extremely small p-values, and the direction of the effect was consistent across all of them: discussions containing a confirmed puppet-master sit at the lower end of every structural measure. The strongest separations appeared in closeness centrality ($p \approx 1.3 \times 10^{-19}$, $d \approx -0.31$), reciprocity ($d \approx -0.30$), density and degree centrality ($d \approx -0.27$), and eigenvector centrality ($d \approx -0.27$). Taken together, these point toward master-containing discussions being structurally sparser, less reciprocal, and less interconnected than organic discussions – that is, conversations in which interaction is somewhat more one-directional and less evenly distributed among participants.

It is also worth noting that the Infected group consistently fell between the Masters and Uninfected groups, usually closer to the Masters group, on nearly every metric. This suggests that the structural effect is graded rather than binary, and that the presence of a puppet-master, rather than the presence of puppets alone, is associated with the strongest structural signature.

These findings must be qualified by the magnitude of the effects. Although the p-values for the Masters comparison are very small, all the



corresponding Cohen's d values are small in absolute terms (roughly between 0.10 and 0.31). The statistical significance is therefore driven in large part by the size of the dataset rather than by a large practical difference, and the distributions of the three groups overlap heavily. Consequently, while these metrics reveal a genuine and consistent structural tendency, they are not, on their own, sufficient to classify an individual discussion as containing a puppet-master.

Several limitations of our current methodology likely contribute to this. First, several of the metrics we computed are mathematically redundant: in-degree and out-degree are identical, degree centrality is exactly twice the density, and the three PageRank-style scores all reduce to the inverse of the number of participants. The set of metrics therefore contains fewer independent signals than its size suggests. Second, and more importantly, each node-level metric was reduced to a single value by averaging over all participants in the discussion. This averaging may dilute precisely the signal we are searching for, since the distinctive position of one or two sockpuppets is averaged together with the many ordinary participants around them. Finally, the diameter metric could not be used, as it is only defined for strongly connected graphs and most of our discussion graphs were not strongly connected.

Considering these results, we conclude that graph structure does carry information that distinguishes master-containing discussions from clean ones, but that the averaged whole-graph metrics used in this stage are too coarse to exploit that information directly. This motivates the next stage of our work, in which we intend to move beyond simple averages toward node-level and distributional features, and toward subgraph-level analysis, to

capture the local structural signatures that whole-graph averaging appears to obscure.

4.6 **Individual User-Level Metrics**

The analysis in the previous sections examined each discussion graph as a whole, with node-level metrics reduced to a single graph-wide average. As

noted in the Findings above, this averaging may dilute the very signal that distinguishes puppet accounts from other participants. To address this directly, we repeated the analysis at the level of individual users rather than individual graphs: for every user appearing in the dataset, we collected their metric values from every discussion in which they participated, and averaged these values per user rather than per graph. Users were then grouped by their classification as Sockpuppets, Puppet Masters, or Regular Users, and the same descriptive and inferential statistics used in Section 4.3 were recalculated for each group.

Of the users considered, 653 participated in more than five discussions and were treated as sufficiently represented for this analysis. Users whose per-user metrics could not be computed for at least one discussion in which they appeared (334 users in total, including 1 sockpuppet and 2 puppet masters) were excluded from the comparisons below, since an incomplete metric history could not be reliably averaged.

When Sockpuppets were compared against Regular Users, none of the eleven individual-level metrics reached statistical significance (all $p > 0.05$); the largest effect size in this comparison was for betweenness centrality (Cohen's $d \approx 0.24$, $p \approx 0.078$), and every other effect was smaller still. In other words, once graph structure is examined at the level of the individual participant rather than as a group average across an entire graph, confirmed sockpuppet accounts were statistically indistinguishable from ordinary Wikipedia editors across every metric tested.

A markedly different picture emerged when Puppet Masters were compared against Regular Users. Every metric except clustering coefficient reached statistical significance, several with very small p-values: PageRank ($p \approx$

0.0004, $d \approx 0.73$), eigenvector centrality ($p \approx 0.003$, $d \approx 0.71$), degree centrality ($p \approx 0.002$, $d \approx 0.62$), connection graph PageRank ($p \approx 0.009$, $d \approx 0.62$), closeness centrality ($p \approx 0.009$, $d \approx 0.64$), in-degree ($p \approx 0.003$, $d \approx 0.56$), betweenness centrality ($p \approx 0.002$, $d \approx 0.53$), k-order Markov probabilities ($p \approx 0.021$, $d \approx 0.46$), and out-degree ($p \approx 0.007$, $d \approx 0.44$). Unlike the whole-graph comparisons in Section 4.4, where the Masters-vs-Uninfected effect sizes were uniformly small (all $|d|$ below 0.32), several of these individual-level effect sizes for Masters vs. Regular Users are moderate to large, indicating that confirmed puppet-masters, as individuals, occupy substantially more central and influential positions in the network than the typical participant.

Comparing Sockpuppets directly against Masters showed that the two groups differ from one another primarily in one respect: connection graph entropy ($p \approx 0.042$, $d \approx -0.63$), with sockpuppets showing lower entropy than masters. In-degree approached significance ($p \approx 0.054$, $d \approx -0.53$) but did not cross the conventional threshold. Every other metric comparison between sockpuppets and masters was non-significant, suggesting that, of all the metrics considered, connection graph entropy is the one that most consistently separates confirmed puppet accounts from the accounts said to operate them.

To complement these statistics, the figures below show the same violin plot and ECDF views used in Section 4.4, now computed over per-user averages rather than per-graph averages. The metrics are presented in descending order of effect size for the Masters vs. Regular Users comparison, with connection graph entropy included last as the one metric that significantly separated Sockpuppets from Masters.



PageRank (largest individual-level effect: Cohen's $d \approx 0.73$, $p \approx 0.0004$ for Masters vs. Regular Users)

At the individual level, PageRank produced the largest effect size of any metric. The violin plot shows the Masters distribution shifted well to the right of both other groups, with a visibly higher median, while the Sockpuppet and Regular User distributions overlap closely with one another. The ECDF makes the same pattern clear: the Masters curve (blue) trails behind the Sockpuppet (green) and Regular User (red) curves for nearly the entire range, meaning masters are disproportionately represented among the highest-PageRank individuals in the dataset. Unlike the whole-graph PageRank comparison in Section 4.4, where PageRank behaved mainly as a proxy for discussion size, this user-level PageRank reflects an individual's own position of importance within the discussions they take part in.

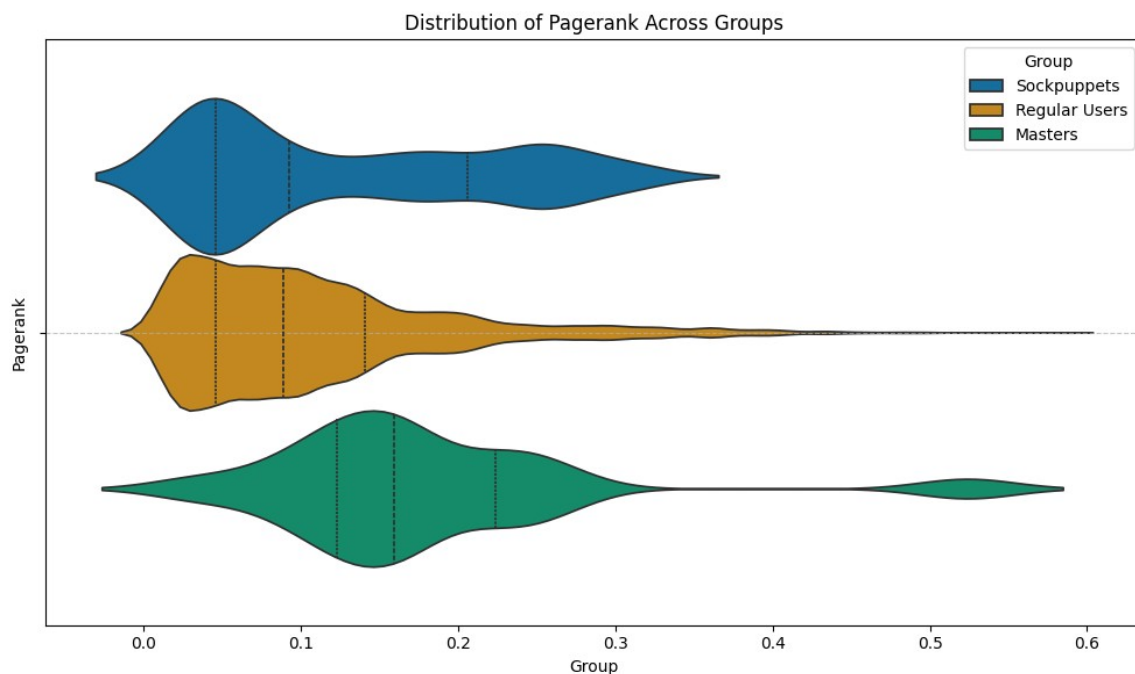


Figure 6a. Violin plot of individual-level PageRank across Sockpuppets, Regular Users, and Masters.

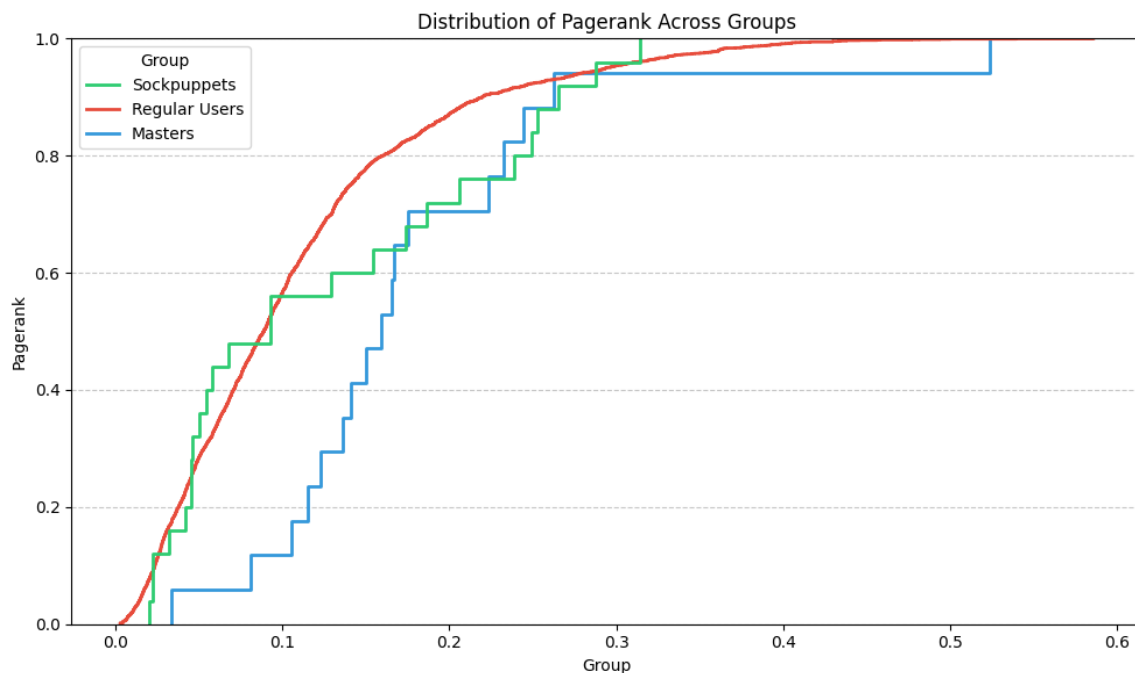


Figure 6b. ECDF of individual-level PageRank across Sockpuppets, Regular Users, and Masters.

Eigenvector Centrality (Cohen's $d \approx 0.71$, $p \approx 0.003$)

Eigenvector centrality tells a similar story to PageRank: the Masters violin is shifted toward higher values with a higher median, while the Sockpuppet and Regular User distributions again largely overlap. The ECDF shows the Masters curve (blue) positioned to the right of the other two for most of the range, meaning confirmed puppet-masters tend to be connected to other well-connected individuals more often than either sockpuppets or ordinary users.

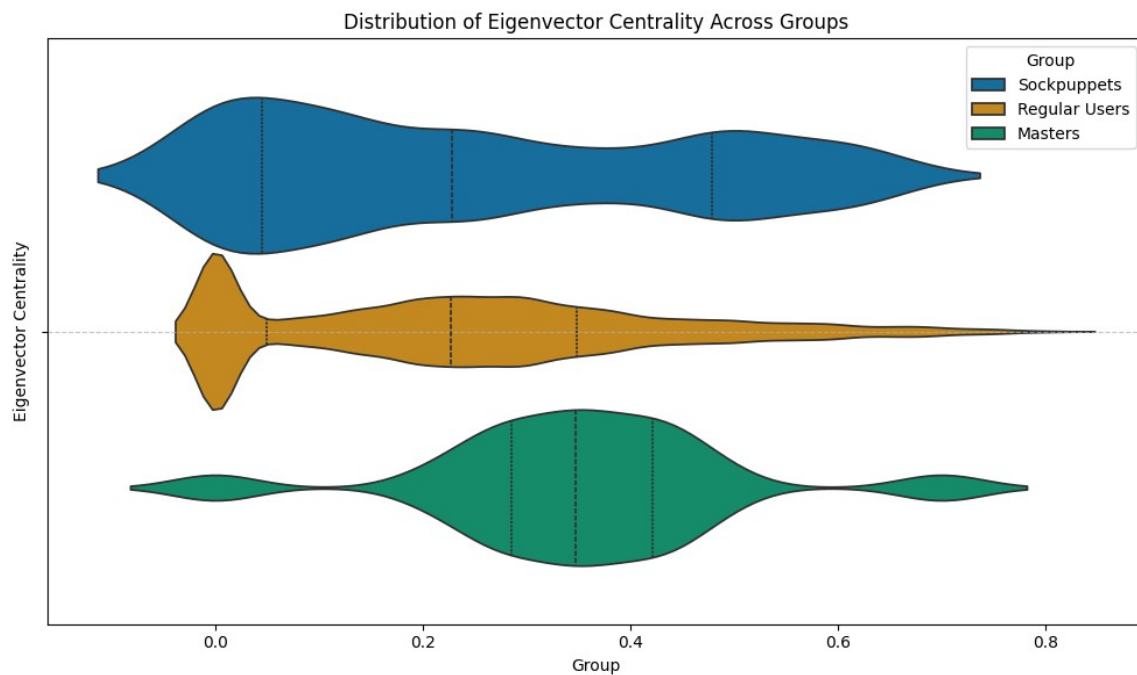


Figure 7a. Violin plot of individual-level eigenvector centrality across Sockpuppets, Regular Users, and Masters.

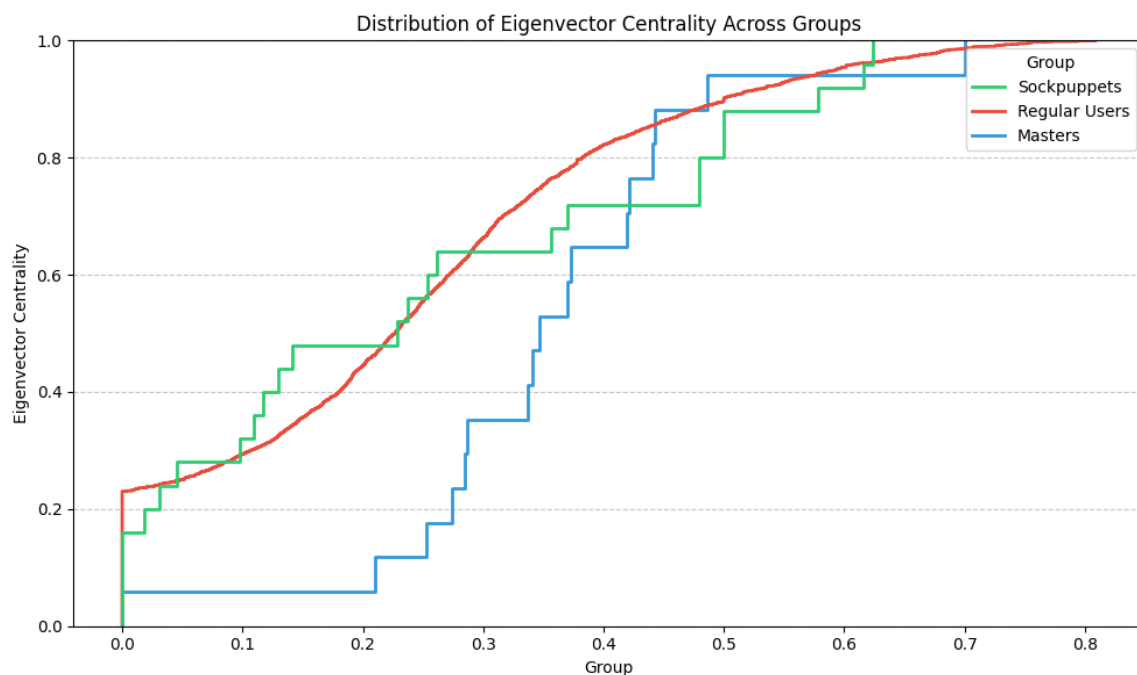


Figure 7b. ECDF of individual-level eigenvector centrality across Sockpuppets, Regular Users, and Masters.

Closeness Centrality (Cohen's $d \approx 0.64$, $p \approx 0.009$)

The Masters group again separates from the other two on closeness centrality, with a violin shifted toward higher values and an ECDF curve positioned to the right of the Sockpuppet and Regular User curves, which

remain close to one another. Individually, master accounts tend to be reachable from other participants in fewer steps than either puppets or regular users, consistent with occupying a more central position in the discussions they take part in.

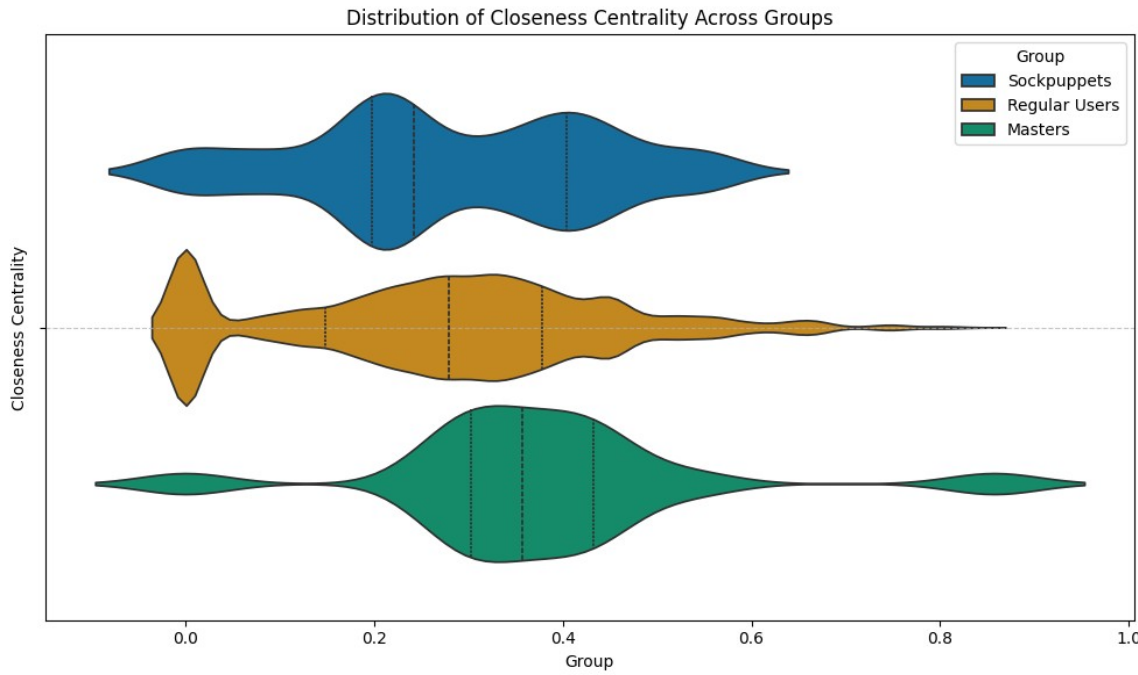


Figure 8a. Violin plot of individual-level closeness centrality across Sockpuppets, Regular Users, and Masters.

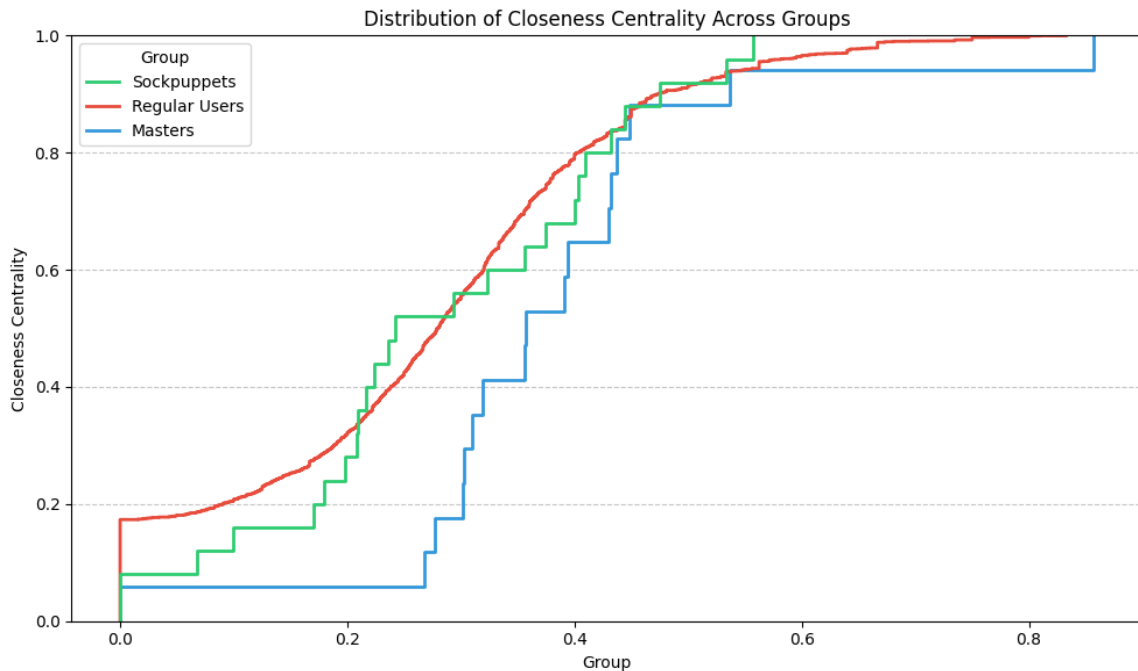


Figure 8b. ECDF of individual-level closeness centrality across Sockpuppets, Regular Users, and Masters.



Degree Centrality (Cohen's $d \approx 0.62$, $p \approx 0.002$)

Degree centrality shows the same ordering once more: the Masters distribution is shifted toward higher values relative to Sockpuppets and Regular Users, and its ECDF curve sits furthest to the right. This indicates that, on average, a puppet-master has directly interacted with a larger share of the other participants in a discussion than either a sockpuppet or a regular user has.

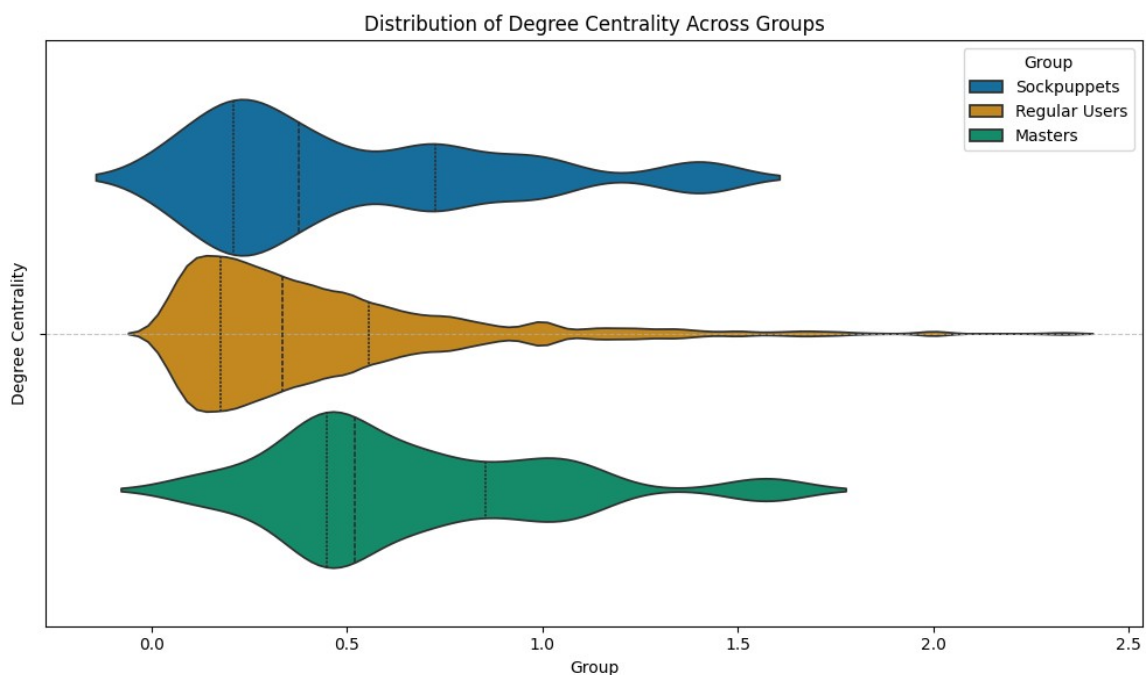


Figure 9a. Violin plot of individual-level degree centrality across Sockpuppets, Regular Users, and Masters.

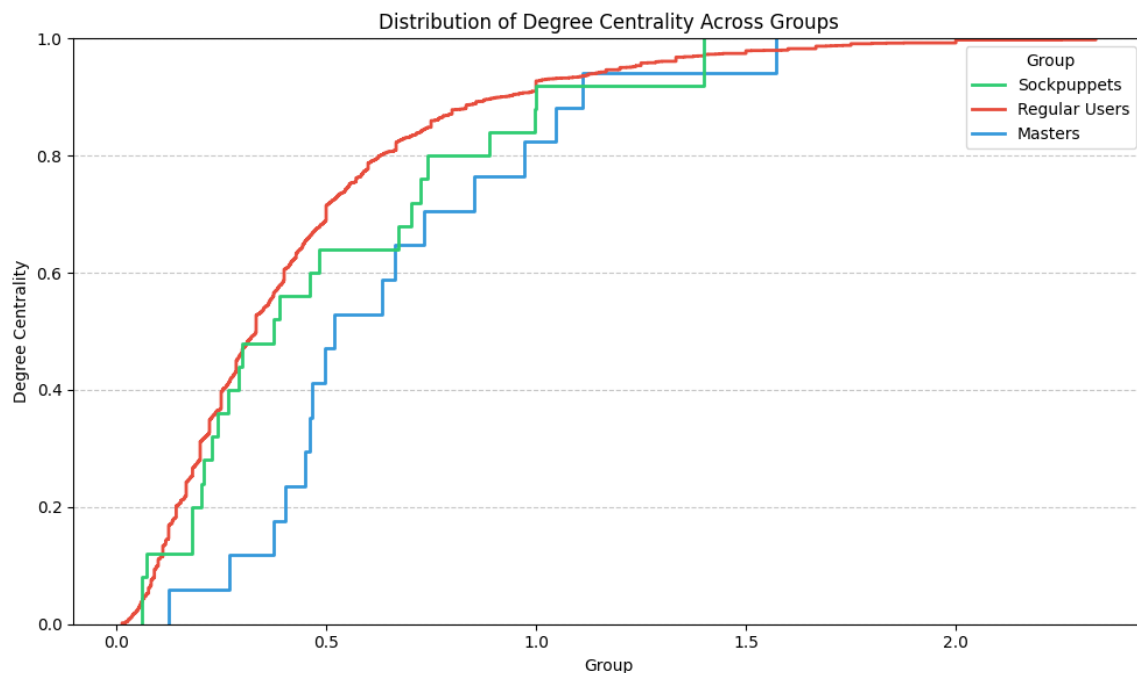


Figure 9b. ECDF of individual-level degree centrality across Sockpuppets, Regular Users, and Masters.

In-Degree (Cohen's $d \approx 0.56$, $p \approx 0.003$ for Masters vs. Regular Users; $d \approx -0.53$, $p \approx 0.054$ for Sockpuppets vs. Masters)

In-degree is notable because it is the metric that comes closest to separating Sockpuppets from Masters directly, in addition to separating Masters from Regular Users. The violin plot shows Masters skewed toward higher in-degree values, Regular Users concentrated at lower values, and Sockpuppets sitting closer to the Regular User distribution than to the Masters distribution. The ECDF shows the Masters curve (blue) trailing behind the Sockpuppet (green) and Regular User (red) curves, which nearly overlap one another for most of the range, consistent with masters receiving more replies on average than either puppets or ordinary participants.

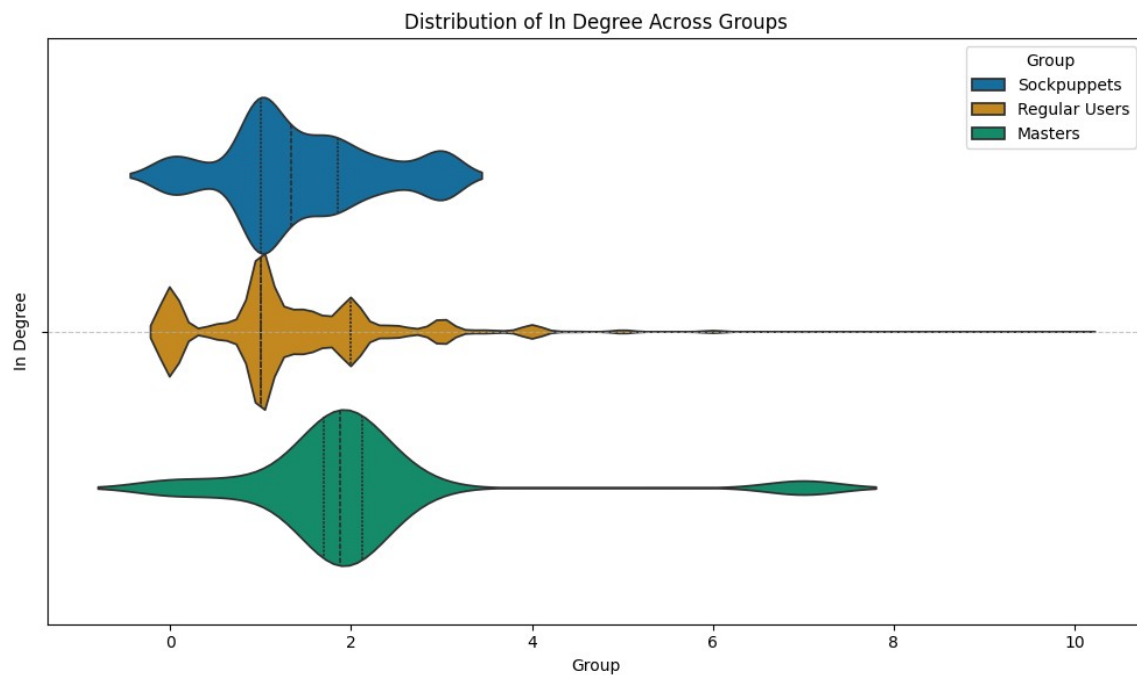


Figure 10a. Violin plot of individual-level in-degree across Sockpuppets, Regular Users, and Masters.

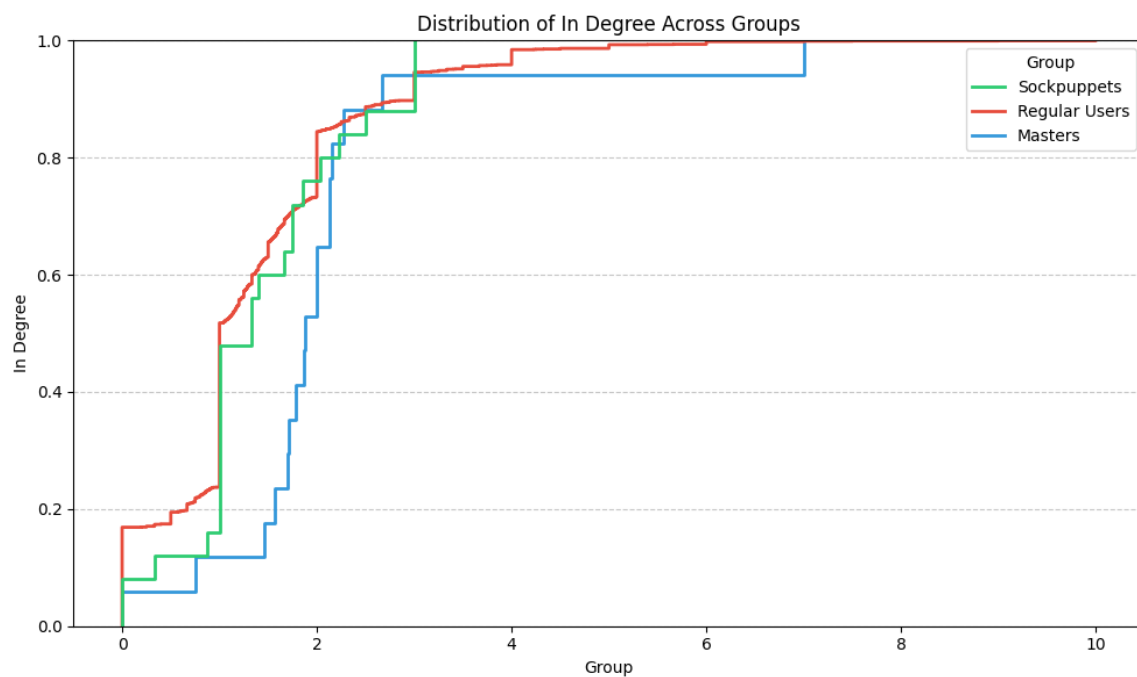


Figure 10b. ECDF of individual-level in-degree across Sockpuppets, Regular Users, and Masters.

Connection Graph Entropy (the only metric to significantly distinguish Sockpuppets from Masters: Cohen's $d \approx -0.63$, $p \approx 0.042$)

Connection graph entropy was the one metric on which Sockpuppets and Masters differed significantly from one another, with sockpuppets showing

lower entropy. The violin plot shows the Sockpuppet distribution concentrated at lower entropy values than either the Masters or Regular User distributions, while Masters show the highest and most spread-out entropy values of the three groups. Lower entropy indicates a more predictable, concentrated pattern of indirect connections, while higher entropy indicates a more diffuse and varied one; on this metric, confirmed puppet accounts appear more narrowly and predictably positioned in the network than the accounts said to operate them.

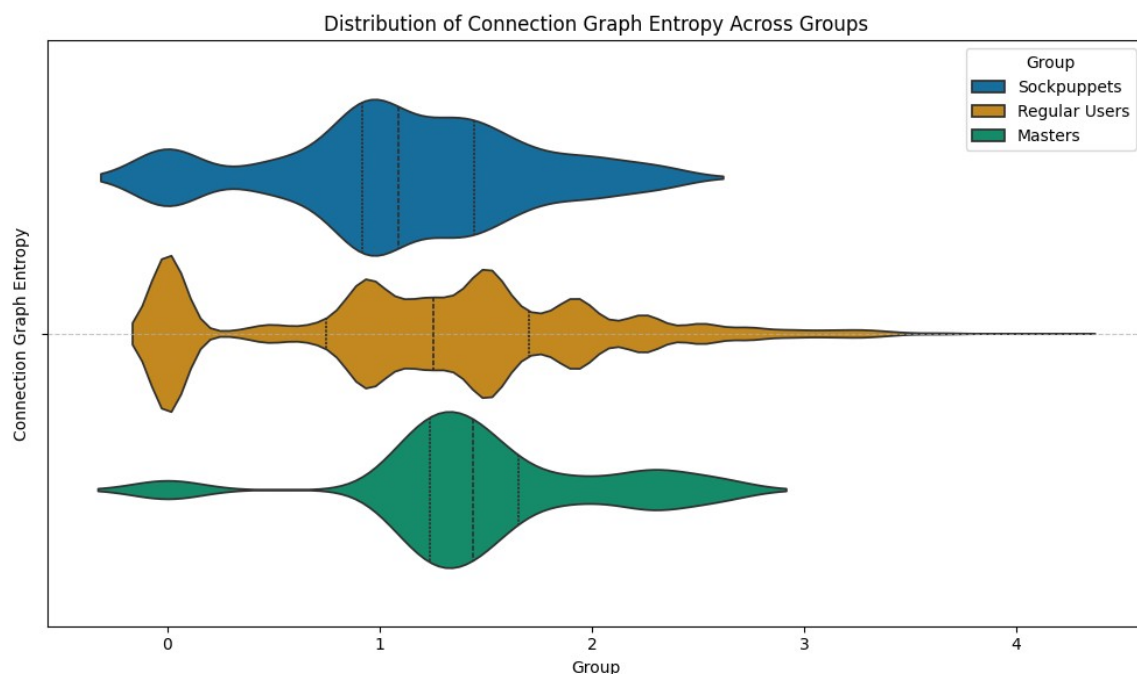


Figure 11a. Violin plot of individual-level connection graph entropy across Sockpuppets, Regular Users, and Masters.

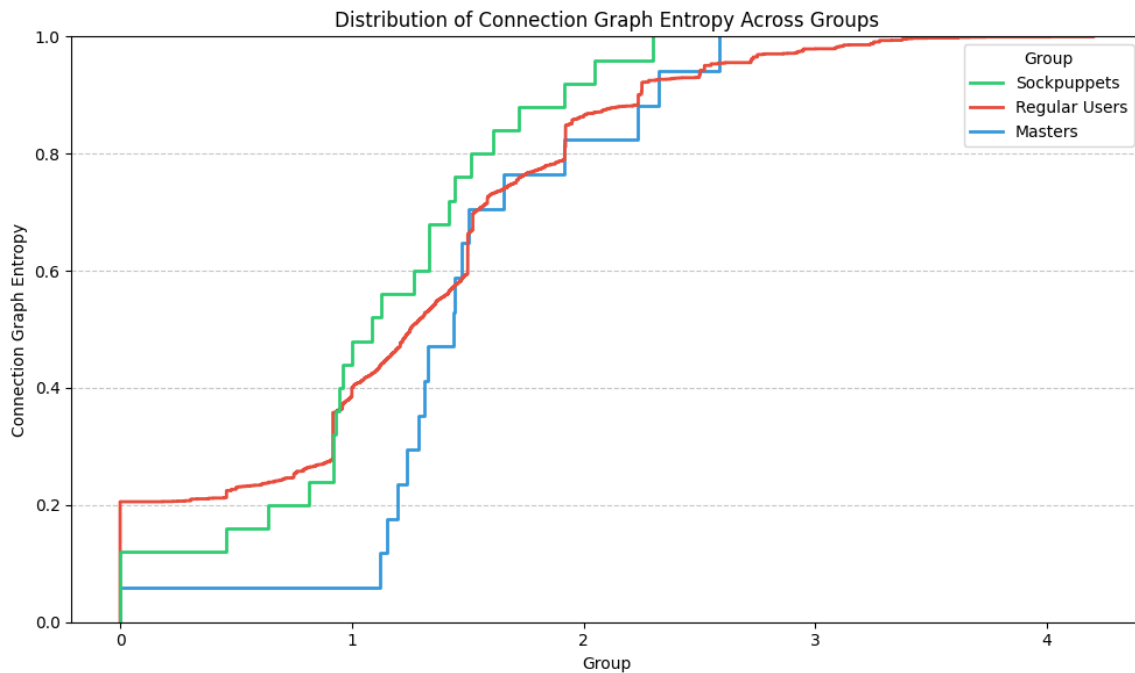


Figure 11b. ECDF of individual-level connection graph entropy across Sockpuppets, Regular Users, and Masters.

Taken together, the individual-level results echo and sharpen the whole-graph findings reported in Section 4.5. Confirmed sockpuppet accounts remain difficult to distinguish from ordinary users when examined individually, with no metric reaching statistical significance in that comparison. Puppet-masters, however, are clearly distinguishable from regular users as individuals, and the effect sizes for this comparison are considerably larger than any of the whole-graph effect sizes reported earlier, suggesting that whatever structural signal exists in this dataset is carried mainly by the master accounts rather than by the puppets themselves.

5. Potential Explanations:

The results reported above raise a natural question: why would puppet-masters occupy such distinctive positions in these discussions, while their sockpuppets remain statistically indistinguishable from ordinary users? We do not have direct evidence to confirm any single explanation, and the possibilities below are not mutually exclusive; we present them as candidate



explanations that could motivate future investigation, rather than as established conclusions.

Selection bias in how puppet-masters are identified

One possibility is that the Masters group is subject to a form of selection bias introduced by how puppet-masters come to be identified in the first place. Confirmed puppet-masters on Wikipedia are typically uncovered through CheckUser investigations or moderator scrutiny, processes that may be more likely to be triggered by accounts that are already prominent, prolific, or involved in contentious discussions. If this is the case, the Masters group in our dataset could be disproportionately composed of already-central, highly active editors, and the structural differences we observe could partly reflect this detection process rather than being a direct consequence of operating sockpuppets.

A functional division of labor between masters and puppets

A second possibility is that the pattern reflects an intentional division of roles. A puppet-master may need to remain a visible, central, and credible figure in a discussion for their opinions to carry weight, while their sockpuppets are deployed more quietly to reinforce that position without drawing attention to themselves. Under this explanation, both the central role of the master and the comparatively unremarkable structural profile of the puppets would be by design, rather than incidental.

Differences in participation frequency and statistical power

A third possibility relates to how frequently each type of account participates in discussions. If sockpuppet accounts tend to be used more sparingly, or across a narrower range of discussions, than either regular users or the masters who control them, then the per-user averages computed for

sockpuppets would be based on fewer observations than those for the other groups. Fewer observations reduce the statistical power available to detect a genuine but modest structural signal, even one of a magnitude similar to the small effects found in the whole-graph comparisons in Section 4.4.

Averaging across discussions may still dilute the signal

A fourth possibility, related to a limitation already noted in Section 4.5, is that averaging a user's metric values across every discussion in which they appear may still obscure meaningful variation in how that user behaves from one discussion to another. A sockpuppet account might behave quite differently depending on a discussion's purpose, and averaging across these different roles could wash out a signal that would be visible if individual discussions, rather than individual users, were the unit of comparison.

A narrower, more targeted pattern of connections for sockpuppets

Finally, the one metric that did significantly separate Sockpuppets from Masters, connection graph entropy, may point to a difference in how narrowly each type of account is deployed. Lower entropy for sockpuppets could indicate that these accounts are used in a more targeted, repetitive way, for example to support a specific position in a specific dispute, while masters, who may participate more broadly across the encyclopedia in their own right, would naturally accumulate a more varied and less predictable set of indirect connections.

Each of these explanations is speculative, and in most cases difficult to distinguish from the others using the metrics computed here. Confirming or ruling out any of them would likely require additional data, such as the number and diversity of discussions each account type participates in, or a

closer qualitative examination of how confirmed puppet-masters behave prior to, and independently of, their use of sockpuppets.

References:

- [1] Friedman, E., Resnick, P., & Sami, R. (2007). Manipulation-resistant reputation systems. *Algorithmic Game Theory*, 677.
- [2] Saez-Trumper, D. (2019). Online disinformation and the role of wikipedia. *arXiv preprint arXiv:1910.12596*.
- [3] Ortega Soto, J. F. (2012). Wikipedia: A quantitative analysis.
- [4] Grabowski, J., & Klein, S. (2023). Wikipedia's Intentional Distortion of the History of the Holocaust. *The Journal of Holocaust Research*, 37(2), 133-190.
- [5] Shachaf, P., & Hara, N. (2010). Beyond vandalism: Wikipedia trolls. *Journal of Information Science*, 36(3), 357-370.
- [6] Blikstad-Balas, M. (2016). "You get what you need": A study of students' attitudes towards using Wikipedia when doing school assignments. *Scandinavian journal of educational research*, 60(6), 594-608.
- [7] Kumar, S., Cheng, J., Leskovec, J., & Subrahmanian, V. S. (2017, April). An army of me: Sockpuppets in online discussion communities. In *Proceedings of the 26th international conference on world wide web* (pp. 857-866).
- [8] Solorio, T., Hasan, R., & Mizan, M. (2013). Sockpuppet detection in wikipedia: A corpus of real-world deceptive writing for linking identities. *arXiv preprint arXiv:1310.6772*.
- [9] Mandalari, A. M., Lutu, A., Dhamdhere, A., Bagnulo, M., & Claffy, K. C. (2017). Tracking the Big NAT across Europe and the US. *arXiv preprint arXiv:1704.01296*



- [10] https://tum.wikipedia.org/wiki/User:ClueBot_NG
- [11] <https://en.wikipedia.org/wiki/Wikipedia:STiki>
- [12] Kumar, S., Spezzano, F., & Subrahmanian, V. S. (2015, August). Vews: A wikipedia vandal early warning system. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 607-616)
- [13] Zhang, Z., Li, H., Zhang, Z., Qin, Y., Wang, X., & Zhu, W. (2023). Graph meets llms: Towards large graph models. *arXiv preprint arXiv:2308.14522*
- [14] Sakib, M. N., & Spezzano, F. (2022, November). Automated detection of sockpuppet accounts in wikipedia. In *2022 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)* (pp. 155-158). IEEE.
- [15] Brambilla, M., Javadian, A., & Sulistiawati, A. E. (2021, May). Conversation graphs in online social media. In *International Conference on Web Engineering* (pp. 97-112). Cham: Springer International Publishing.
- [16] Cogan, P., Andrews, M., Bradonjic, M., Kennedy, W. S., Sala, A., & Tucci, G. (2012, August). Reconstruction and analysis of twitter conversation graphs. In *Proceedings of the First ACM International Workshop on Hot Topics on Interdisciplinary Social Networks Research* (pp. 25-31).
- [17] Brambilla, M., Javadian Sabet, A., Kharmale, K., & Sulistiawati, A. E. (2022). Graph-based conversation analysis in social media. *Big Data and Cognitive Computing*, 6(4), 113.
- [18] Chami, I., Abu-El-Haija, S., Perozzi, B., Ré, C., & Murphy, K. (2022). Machine learning on graphs: A model and comprehensive taxonomy. *Journal of Machine Learning Research*, 23(89), 1-64.
- [19] Hagberg, A., & Conway, D. (2020). Networkx: Network analysis with python. URL: <https://networkx.github.io>, 1-48.

- [20] Hagberg, A., Swart, P. J., & Schult, D. A. (2008). *Exploring network structure, dynamics, and function using NetworkX* (No. LA-UR-08-05495; LA-UR-08-5495). Los Alamos National Laboratory (LANL), Los Alamos, NM (United States).
- [21] <https://www.mediawiki.org/wiki/MediaWiki>
- [22] Bianchini, M., Gori, M., & Scarselli, F. (2005). Inside pagerank. *ACM Transactions on Internet Technology (TOIT)*, 5(1), 92-128.
- [23] Freeman, L. C. (1978). Centrality in social networks conceptual clarification. *Social networks*, 1(3), 215-239.
- [24] Watts, D. J., & Strogatz, S. H. (1998). Collective dynamics of 'small-world' networks. *nature*, 393(6684), 440-442.
- [25] Chowdhary, K. (2020). Natural language processing. *Fundamentals of artificial intelligence*, 603-649.
- [26] Chen, S. L. (2010). Wikipedia: A republic of science democratized. *Alb. LJ Sci. & Tech.*, 20, 247.
- [27] Raftery, A. E. (1985). A model for high-order Markov chains. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 47(3), 528-539.
- [28] Wang, J., Zhou, W., Li, J., Yan, Z., Han, J., & Hu, S. (2018, December). An online sockpuppet detection method based on subgraph similarity matching. In *2018 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Ubiquitous Computing & Communications, Big Data & Cloud Computing, Social Computing & Networking, Sustainable Computing & Communications (ISPA/IUCC/BDCloud/SocialCom/SustainCom)* (pp. 391-398). IEEE.