COORDINATED BOTNET DETECTION IN SOCIAL NETWORKS VIA CLUSTERING ANALYSIS

An Undergraduate Research Scholars Thesis

by

PRESTON CLENT PIERCEY

Submitted to the LAUNCH: Undergraduate Research office at
Texas A&M University
in partial fulfillment of the requirements for the designation as an

UNDERGRADUATE RESEARCH SCHOLAR

Approved by
Faculty Research Advisors: Dr. Nate Veldt
Dr. Roger Pearce

May 2023

Major: Computer Engineering

Copyright © 2023. Preston Piercey.

RESEARCH COMPLIANCE CERTIFICATION

Research activities involving the use of human subjects, vertebrate animals, and/or biohazards must be reviewed and approved by the appropriate Texas A&M University regulatory research committee (i.e., IRB, IACUC, IBC) before the activity can commence. This requirement applies to activities conducted at Texas A&M and to activities conducted at non-Texas A&M facilities or institutions. In both cases, students are responsible for working with the relevant Texas A&M research compliance program to ensure and document that all Texas A&M compliance obligations are met before the study begins.

I, Preston Piercey, certify that all research compliance requirements related to this Undergraduate Research Scholars thesis have been addressed with my Faculty Research Advisors prior to the collection of any data used in this final thesis submission.

This project did not require approval from the Texas A&M University Research Compliance & Biosafety office.

TABLE OF CONTENTS

ABSTRACT

Coordinated Botnet Detection in Social Networks via Clustering Analysis

Preston Piercey
Department of Computer Science and Engineering
Texas A&M University

Faculty Research Advisor: Dr. Nate Veldt
Department of Computer Science and Engineering
Texas A&M University

Faculty Research Advisor: Dr. Roger Pearce
Department of Computer Science and Engineering
Texas A&M University

Graphs are a widely used tool in modeling social interaction networks. In a network that consists of authors and pages with time-stamped interactions between one page and one author, we can model the network as a bipartite temporal graph. These graphs are particularly useful in modeling the temporal relationships between users and pages on social media networks such as Reddit, Twitter, or Facebook. This project lays out a three-step approach for the identification of highly coordinated behavior in these massive networks with a three step approach and applies it at scale to real-world data from the Reddit platform. Because of the scale of this data, direct computation of group interactions for all authors in the bipartite graph is too expensive. To address this problem, the bipartite temporal graph between authors and pages is projected into a one-mode weighted graph of authors by specifying a maximum and minimum time between interactions and recording how many times each author interacted on the same page as another author in that window of time. The weight of the edge between these two authors in the projected graph is the count of these interactions; higher edge weights indicate greater potential coordination. In

the second step, we query the projected graph for high edge weight triangles. This highlights triplets of authors that repeatedly interact with the same pages at the same time. Finally, after the author groups of interest have been pruned to a much smaller search space, we return to the more informative metrics considering just these authors in the bipartite graph.

DEDICATION

To my family, friends, and instructors who supported me throughout the research process.

ACKNOWLEDGMENTS

Contributors

I would like to thank my faculty advisors, Dr. Nate Veldt and Dr. Roger Pearce, as well as Dr. Trevor Steil of LLNL, for their outstanding guidance and teaching throughout the course of this research.

Thanks to my parents for their encouragement, love, and support.

Finally, thanks also go to my friends, colleagues, and the department faculty and staff for making my time at Texas A&M University a great experience.

The YGM and TriPoll codebases used for this project were provided by Dr. Roger Pearce and Dr. Trevor Steil. The Reddit data analyzed in this project was collected from the pushshift.io file archives.

All other work conducted for the thesis was completed by the student independently.

Funding Sources

Computing resources, experimentation, and testing in this project were made possible by the Lawrence Livermore National Lab Collaboration Zone Compute Clusters. This project's contents are solely the responsibility of the author and do not necessarily represent the official views of Lawrence Livermore National Lab.

1. INTRODUCTION

1.1 Undeserved Influence in Social Networks

Social networks such as Reddit, Twitter and Facebook continue to grow in size and influence as we spend more time online and increasingly rely on them as sources of news, information, and places of public discourse. Due to their growing ability to shift public opinion, these platforms have become high-value targets for malevolent parties to spread misinformation to a wider audience and sway public opinion. Each user of social media deserves the same voice, and it is unfair and unnatural for one party to have an artificial influence that is able to drown out others.

If a malevolent actor is working alone on one of these platforms, the spread of this misinformation is often limited to those who follow the account, and its efficacy is questioned due to the small amount of user interaction. However, when interaction with a page containing misinformation is spread across many accounts coordinated by a single entity it not only skews the value of the page in platform recommendation algorithms leading to the page being recommended to a wider audience, but also gives the page undeserved subjective credibility by users via the 'Bandwagon effect'.

1.2 Changing the Way We Search for Botnets

As more and more content is published on social networks in shorter periods of time and text generation bots improve to the point of being indistinguishable from human text, it becomes increasingly challenging to find these coordinated groups. When a bot comment looks just like human text it is unlikely to be reported by a human user and will never be sent to a content moderator for review. Concerns and information detailing the potential misuse of text generation models is explained in great depth by Goldstein et al. [1]. With the advent of these trainable models that can produce unique and coherent text for any directive, it is likely that we will see these 'influence campaigns' make use of these models to better slip through content moderation and further their influence in the online space.

When a large group of accounts is controlled by a single entity, commands are often issued

to and completed by the entire network of bots at the same time. This is contrary to the typical user interaction that involves single interactions that occur over a greater amount of time, as human interaction is limited by the ability to interact with the platform in the form of reading pages, forming a response, and writing the comment. The hypothesis of this project is that the structure of the coordinated behavior will be measurably different than single-user interaction, and the structure of this behavior can be used to identify malicious accounts. Rather than searching for needles in a haystack, the problem turns into a search for which pieces of hay were all cut at the same time and place.

1.3 The Approach

This coordinated behavior has been studied on Twitter in the form of retweets by Pacheco et al. in "Uncovering Coordinated Networks on Social Media" [2] . However, this analysis was targeted at a specific type of 'reshare' function of the platform solely for pages that included user-provided tags. Our project aims to develop a platform that can be used across an entire network, where interactions are comments on an existing page on the Reddit platform. No other data, such as the subreddit the page exists in or number likes or 'upvotes' on the comment, are taken into account. All of the data used was downloaded from the Pushshift Reddit Archives.*

One application of the bipartite temporal graph is in the representation of user interactions on a network where the raw data contains a record of the author, the time of the interaction, and the page that they interacted with. This makes it particularly well suited to model the user interactions that we seek to study. We consider one set of nodes as users, another set as the places that they may be able to interact with, and edges between these two sets as interactions between a user and a place with an edge label bearing the time of interaction.

We propose a three step approach in finding highly coordinated activity in social media platforms. Beginning with a bipartite temporal multigraph constructed from the raw data, we apply the following analysis:

1. Projection of the bipartite temporal graph to a weighted common interaction network

*https://files.pushshift.io/reddit/

2. Application of standard graph analysis techniques, such as triangle surveying, on the common interaction network to highlight potentially coordinated authors

3. Validating the coordinated interactions by computing hypergraph measures on the original bipartite temporal graph to verify that multi-way interactions truly occurred for the group of authors

The details of how we implemented this framework for this work are explained in much greater detail in Section 2: Methods.

In the Results section, we provide experimental data that demonstrate the success of this framework in identifying coordinated activity in data from the month of January 2020, as well as analysis of the month of October in 2016, just prior to the 2016 election. In the Conclusions section, we address the successes and shortcomings of this approach in accomplishing the project goal and directions for future research.

In summary, this project could be used to identify botnets based solely on their interaction with pages, regardless of page content or location within the network. Most current methods of identification involve finding of bots in share-reshare networks where bots are directly spreading pages on the platform (i.e. Twitter, Facebook, etc.), whereas this work will focus on botnet interaction with pages from author-comment lens based on their time of action.

2. METHODS

2.1 Background

All of the graphs that are defined below were implemented using the distributed containers of YGM [3].

2.1.1 The Bipartite Temporal Multigraph

In this specific application, we consider data from the Reddit network. Although the structure of comment trees in this network can vary widely due to the nested comment model that is used, we consider each comment to be an interaction with the page at the root of the comment tree. Because our aim is to identify coordinated behavior by the similarity of the time that users comment and pages that users comment on, there are three pieces of information from the raw data that are critical for modeling the network that could give insight into these similarities:

$$\text{user}, \text{page}, \text{comment time}. \tag{1}$$

We represent a collection of such records as a bipartite temporal multigraph (BTM). This is a multigraph rather than just a graph because a user can comment multiple times on the same page. The distinctive factor between each edge between the same author and page is the time metadata that serves as a label for the edge. These time labels are crucial in the production of the projected common interaction graph in step 1 of analysis detailed in section 2.2. Formally, the multigraph is $B = (U, P, E, t)$, where U is the set of users, P is a set pages, and E is a multiset of pairs from $U \times P$ with each pair representing a comment from a specific user (u) on a specific page (p). For each $e = (u, p) \in E$, $t(e)$ represents a timestamp for the user-page edge $e \in E$.

This representation of the data makes it easy to quantify the spatio-temporal relationship between authors on the network. For instance, if we wanted to select 5 authors and determine the number of pages that they each comment on together we can visit each author, get a list of the pages that they have commented on, and intersect these lists to get each page where they share

common interactions.

2.1.2 The Triplet Hypergraph

In this section, we detail intended outcome of the analysis. To begin, let us define a hypergraph in the context of this project, and how it relates to the bipartite temporal multigraph that is defined above. A hypergraph is a generalization of the graph structure that is widely known. A standard graph has a set of nodes and set of edges that connect exactly two of the vertices. In B we separate the set of vertices into two entirely unique sets U and P to make a distinction in the two different types of vertices of the network and highlight the unique structure that makes the graph useful in modeling user interactions, but this only makes it a special case of the standard graph. Each edge is still between only two nodes, a user and a page. The special component of a hypergraph is the ability to model relationships in groups of vertices, rather than the pairwise limitation of a standard graph. These groups of vertices are called hyperedges. This makes a hypergraph useful in representing how users may be grouped in our network, rather than just their interaction with a given page. Our end goal is the identification of coordinated groups of authors on Reddit, so we could consider each group to be vertices that are members of the same hyperedge. To model this system, we define a Hypergraph $H = (U, G)$, where U is again the set of users and G is a set of hyperedges, or groupings of the user base. If we are careful in our choice of the groups that we consider in G, then we can use the data stored in B to inform a grouping scheme for H that potentially targets coordinated behavior.

Once we have constructed the bipartite multigraph B it would be great to immediately begin recording counts for all of the possible multiway user interactions as weighted instances of the hyperedges described in the previous paragraph. However, because there are $2^{|U|} - 1$ ways to select groups out of the the set of authors U, this approach quickly becomes exceedingly computationally expensive and not feasible at the scale of a large social network, even for just a month of data. For this reason, we focus our approach to finding triplets of coordinated users. Triplets are the smallest structure in a hypergraph that still represent groups of authors rather than pairwise relationships, and these three way relationships can also be modeled well in the form of triangles in a standard

graph. Furthermore, the importance of triangles in standard graph analysis for social networks is a well documented phenomenon. There are still $O(|U|^3)$ possible triplets that can be formed from the set of authors U, but this focus allows us to make use of the fast triangle surveying system TriPoll [4] to quickly process and reduce standard graphs to a network consisting only of triangles, which we then consider for potential hyperedge triplets. These methods make the hypergraph problem much more computationally approachable but still leave the possibility for larger groups to be formed after triplets of interest have been shown to exhibit coordination.

Formally, we define a *hypergraph triplet* as a set of three users $\{x, y, z\} \subseteq U$, such that the set $T = \{(x, p), (y, p), (z, p) \in E\}$ exists for at least one page $p \in P$. In layman's terms, a triplet occurs when any three authors comment on the same page. Using this definition, we can develop a weighting scheme:

$$w_{xyz} = \text{ number of different pages where } x, y, \text{ and } z \text{ have a three-way interaction.} \tag{2}$$

2.1.3 Coordination Measures for Hypergraph Triplets

One meaningful way to measure coordinated activity among triplets of users is to just see how large w_{xyz} is for different triplets. This provides some indication of coordinated activity, but it is entirely possible that a triplet of extremely active users comment on a large number of the same pages due to a high number of pages interacted with overall, rather than a cohesive effort. To lessen the effects of extremely high single user interaction it can be useful to normalize w_{xyz} in a way that will take into account the activity of the authors. There are many ways to measure how active a user is, but the easiest way is to simply count the number of pages that a user interacts with. For a user $x \in U$, this is

$$p_x = \text{the number of pages where user } x \text{ has at least one comment.} \tag{3}$$

This leads to the following *normalized triplet coordination score*

$$C(x, y, z) = \frac{3 * w_{xyz}}{p_x + p_y + p_z}. \tag{4}$$

Because w_{xyz} is at most the minimum of p_x, p_y, and p_z and at the least could potentially be zero, for every triplet set $\{x, y, z\} \subseteq U$, it will always hold true that $C(x, y, z) \in [0, 1]$. Although this will not sift botnets with extremely widespread interaction to the top of the search list like using the direct approach with w_{xyz}, it may ensure greater focus on very targeted botnet usage where the accounts are used to influence a specific community.

The challenge is not just defining a coordination score, but also finding coordinated triplets effectively that have high values of w_{xyz} or $C(x, y, z)$. Because there is not a good way to calculate these metrics in the hypergraph directly that is not still extremely computationally expensive, we must first prune our search space to the triplets that we want to compute these values for and then return to the hypergraph representation to assess their coordination. To prune the $O(|U|^3)$ search space down to a more manageable number of interesting groups, we propose a three-step approach in finding potential groups of coordinated users. This approach, which is documented in the following sections of the Methods chapter, consists of:

1. Projection of the BTM B to a weighted common interactions network C
2. Querying high minimum edge weight triangles within the common interactions network C
3. Computing the true count of multiway interactions for authors in high edge weight triangles.

2.2 Step 1: Projecting the Bipartite Temporal Graph to Common Interaction Network

The bipartite temporal graph B is a fantastic tool for modeling user to page or page to user relationships in a network, but it is hard to directly evaluate relationships that may exist directly between authors. To get a sense of what interactions may exist between pairs of authors, we create a graph that consists of only author nodes and determine the edges and corresponding weights within the new network by computing metrics on data in the bipartite temporal graph. This notion

of a one-mode projection of the bipartite graph is similar to that of Broccatelli et al. in 2016 [5], although they generate a different graph after projection, and Wang et al. in 2016 [6].

More formally, the structure of this one-mode weighted graph is $C = (U, I, w')$, where U is again the set of users, and I is a set of pairs from $U \times U$ with each pair representing shared interactions between authors. The weighting function will be explained in detail after the explanation of temporal windowing process.

In our framework, this projection is dependent both on the time of the comment and the page that it was written to. Using temporal data to inform the structure is not a new idea, and was proposed by Moody et al in 2009 [7]. To start, we determine a window of delay between comments $(\delta_1,\ \delta_2)$ within which we are interested in looking for coordinated activity. Not only does the time window allow us to target specific types of coordinated activity, but it also reduces the overall computational load in generating the common interaction network and the size of the output. For common interaction network size and computation, a narrower difference in $(\delta_1,\ \delta_2)$ will result in a network that is smaller and takes less time to compute but is potentially more coarse in the interaction that it measures. One example where the windowing can be used to target certain types of behavior is in the extremely short interaction times of share-reshare networks. Because the sharing happens almost immediately throughout the botnet, a short time window will capture more interactions for these groups in comparison to slower moving bots generating content.

2.2.1 Weighting Scheme in the Common Interaction Network

In this projected graph, we focus on pairwise interactions. This is the common interaction network C referenced above. For fixed values (δ_1, δ_2), such that $\delta_2 > \delta_1 \geq 0$ and a pair of users $\{x, y\} \in U$ let

$$S_{xy} = \{p \in P \colon x \text{ and } y \text{ post within } \delta_2 \text{ but no closer than } \delta_1 \text{ seconds of each other in } p\}. \quad (5)$$

Then define $w'_{xy} = |S_{xy}|$. A way that we could search for potentially coordinated hypergraph triplets is to look at three nodes $\{x, y, z\}$ defining a triangle in the graph and consider edge

information. If we consider that multiway interactions within the time period of $(\delta_1,\ \delta_2)$ will be reflected in the weights of $w'_{xy}, w'_{xz}, w'_{yz}$ (along with any number of pairwise interactions between x and y, y and z, or x and z), we see that the number of multiway interactions in this time window is at most the minimum of $\{w'_{xy}, w'_{xz}, w'_{yz}\}$. If we focus on the minimum edge weight of triangles, we can be sure that both of the other edges in a triangle meet or exceed this threshold of pairwise coordination.

The same issue of extremely high user activity leading to false positives in high triangle weights can occur in the common interaction graph. To account for this high frequency of interaction in a way that is congruent to our hypergraph metric, we define a very similar metric which has a few key differences to preserve the normalization range of $[0, 1]$. One key difference is in the way that we count the number of pages interacted with for each user. When we are considering this value for an author in the common interaction graph, we would want to define it as:

$$P'_x = \text{the number of pages used to create a projection edge with } x \text{ as a vertex} \tag{6}$$

Using this definition of the page count of an author, a normalized coordination score of a common interaction graph triangle can be is defined as:

$$T(x, y, z) = 3 * \min\{w'_{xy}, w'_{yz}, w'_{xz}\}/(P'_x + P'_y + P'_z), \tag{7}$$

where w'_{xy} is the edge weight between authors x and y in the common interaction graph as defined above. The advantage of the change in definitions is that we preserve the property of equation 4, where $T(a, b, c) \in [0, 1]$ holds for every single triangle in the common interaction graph. By definition of the weighting scheme w', and because only one interaction between two authors is counted per page, the maximum value it could take is equal to the greater of the two potential number of pages that the considered authors interact on. This property ensures that $\min\{w'_{xy}, w'_{yz}, w'_{xz}\}$ is always less than or equal to $\min\{P'_x, P'_y, P'_z\}$, and that $T(x, y, z)$ will always yield a value in the range of $[0, 1]$.

To summarize this step in the process, each time that two authors place a comment on the same page within the time window $(\delta_1,\ \delta_2)$ of each other, an edge is added to the common interaction network with a weight of 1, or if the edge already exists its weight is incremented. An algorithm that reflects this process is shown in Algorithm 1.

Algorithm 1 Bipartite Graph Projection

Require: A bipartite temporal multigraph $B = (U, P, E, t)$
Require: A time window $(\delta_1,\ \delta_2)$
Ensure: An undirected common interaction graph $C = (U, I, w')$
Ensure: A list of users that records the number of pages where an interaction was considered $L = (U \times \mathbf{W})$

for $p \in P$ **do**
    $S_I =$ an empty set from $(U \times U)$
    $S_{P'} =$ an empty set from (U)
    $N =$ neighborhood(p), sorted by $t(e)$ in ascending order
    for $(x,\ p) \in N$ **do**
        for $(y,\ p) \in N \mid (t((y,\ p))\ >=\ t((x,\ p)))$ **do**
            if $\delta_1\ <=\ (t((y,\ p))\ -\ t((x,\ p)))\ <=\ \delta_2$, and $x \neq y$ **then**
                Add edge $(x,\ y)$ to S_I
    for $(x, y) \in S_I$ **do**
        Add user x to $S_{P'}$
        Add user y to $S_{P'}$
        if Edge (x, y) already exists in I **then**
            Increase the weight of (x, y) by 1
        else
            Add edge $(x,\ y)$ to I with a weight of 1
    for $x \in S_{P'}$ **do**
        if $x \in L$ **then**
            Increment the page count of x by 1
        else
            Add user x to L with a page count of 1
    return $C = (U, I, w)$, $L = (U \times \mathbf{W})$

The end result of this process is a common interaction network that notes the number of

times that any two authors commented on the same page within a known timeframe of each other. Optimally, this projection step will only have to be performed once, but if trends or networks are discovered during the later steps of analysis then it may be necessary to perform the projection again with revised parameters. For example, it can be useful to search for coordinated triplets with a small window of time, manually check their interaction behavior and content. and remove them from the original bipartite temporal multigraph and rerun the projection. However, this must be done carefully so as not to prematurely dismiss coordination that was not captured in the first time window. We could also take the opposite approach where we use a small time window to identify triplets that we are interested in understanding what coordination they may exhibit, and reproject the original Bipartite Temporal Multigraph for just this smaller group of users with a longer time window.

Because of the trade off in time windowing, w'_{xy} will always be less than or equal to the number of pages that both x or y interact on, so it is important to have some background knowledge of the data and type of coordination to be targeted before beginning analysis. If the bipartite temporal graph represents data from a low traffic network, a larger time window should be selected. If it is a network with much greater traffic then a smaller time window could be used to minimize the computational load of the projection. The size of the projected common interactions graphs can become extremely large for Reddit data with a time window of just an hour.

For example, in share-reshare networks, where one author will share a page to its followers and then several other users reshare this page, follower interactions happen in fast succession after the original action. A very short time window can be used to target these types of networks because the interactions happen almost immediately. One such example that has been studied previously in Pacheco et al.[2] are tweets and the retweet function of the Twitter platform. The 'share' would be the original tweet and the 'reshares' are bot accounts retweeting the author's work.

2.3 Step 2: Querying High Edge Weight Triangles in the Common Interaction Graph

2.3.1 The Motivation Behind Triangle Enumeration

Now that the BTM B has been projected as a graph C containing solely authors with weighted edges indicating the number of pairwise interactions between two authors, we want to determine groups of authors in our common interaction graph that may exhibit coordinated activity. Finding the important edges and structures, or the 'backbone' of a bipartite projection was examined by Neal in 2014 [8]. The simplest way to do this at very large scale is to query triangles in the common interaction graph that have high minimum edge weights. If the minimum edge weight of a triangle is N, then we know that each author shares at least N pairwise interactions with both of the other authors. While this does not guarantee that all three authors interacted on the same N pages, it does indicate that there is the possibility for N three way interactions (hypergraph triplets) that include all of the authors.

Because the common interaction graphs are often very large and cannot be analyzed using standard clustering techniques, we use the triangle metadata surveying functionality of TriPoll [4] to either find the triangles with the highest minimum edge weights in the graph, or implement a threshold that specifies the minimum edge weight for triangles that we want to consider in later steps of analysis. The metadata capabilities of TriPoll [4] also make it possible to compute the common interaction graph coordination scores for all triangles in C and threshold based on this value, or threshold on both metrics. The output of the triangle enumeration is a list of triangles in the common interaction graph that satisfy the specified cutoffs. Higher cutoffs will prune the search space of triplets in a network, but the absence of a concrete bound for hypergraph measures from common interaction data does not guarantee that cutoffs will not omit author groups of highly coordinated triplets in the resulting list of triangles. However, a comparison between the common interaction graph and hypergraph metrics is shown for different framework parameters in the results section.

2.4 Step 3: Computing Hypergraph Metrics for Triangles in the Reduced User Graph

Once we have determined triangles of authors that exhibit potential coordination, we must return to the original data to verify these multiway interactions. Now that we do not have to limit ourselves in the computation space because we have used the space and time windowing to prune away many triplets, we can return our focus to just spacial coordination. This process can be thought of as making the edges of the bipartite temporal multigraph B unique, and using the result as a bipartite incidence graph (for the hypergraph of author interactions mentioned in the background) so we can compute hyperedge metrics for author triplets. In much simpler terms, for each author triplet returned on step 3 we compute the number of pages that all three authors have commented on at least once. This is the hypergraph weighting scheme that is described in section 2.1.3. We can also compute more complex values such as equation (4). Because we have pruned our search space of triplets using steps 1 and 2, we can perform these more expensive calculations for a much smaller group of nodes. Note that number of multiway interactions could potentially be much larger or smaller than the minimum edge weight, but they have been experimentally shown to exhibit positive correlation. Yet again, the distributed containers of YGM [3] can accelerate this process by dividing up authors to be checked among several compute nodes.

When authors are ruled out of participating in coordinated activity, they can be removed from the original dataset and the process can begin again with a more honed approach. This is not an element of the analysis, but an approach in the refinement of the output. This could be used in combination with other technologies or content moderators to sift through the results.

3. RESULTS

In all following sections, the software Cytoscape is used to create network visualizations, and the Matplotlib library for Python was used to create the plots. The framework that is described in the methods section was implemented to leverage distributed computing using the containers and functionality of YGM [3]. Due to the highly parallelizable nature of each step, this cut down on computation time immensely. When the memory and computation burden could be spread across multiple compute nodes, it enabled the processing of much larger data sets and longer time windows for the projection process. Because longer windows of time can create many more edges in the common interaction graph for any given page, the projected graph tends to get much larger for longer windows of time. One situation where this is always true is when two temporal windows start at the same, but one window is longer than the other. For example, the projected common interaction graph of a given data set projected for (0, 60s) will always be smaller than or equal to the size of the projection for (0, 1 hr) on the same data. This is because all of the interactions that occur within 0 to 60 seconds also occur within 0 seconds to an hour, and the rest of the interactions that occur from 60 seconds to an hour can contribute a lot of additional edges which can require much greater space to store in memory. A potential workaround for this is using time 'buckets', where we create a projection for windows of {(0, 60s), (60s, 120s), ..., (59 min, 1 hr)}, and merging these projected graphs together at the end of the projection process.

It is important to note that some known 'helpful' bots that serve a community in roles such as automatic moderation (user 'AutoModerator') or deleted users whose usernames are replaced with '[deleted]' are not projected with the other authors for these results. Their interactions are removed from the projection step because there is no value in looking for coordination in the interactions when either 1. we already know that the way that they interact with the network and thus can rule them out of coordinated interaction ('AutoModerator'), or 2. we have nothing to learn from studying their interaction, as '[deleted]' could be any number of users. This is an example of

the process mentioned in at the end of section 2.4, where we have used background information or previous projection results to reduce the search space that we wish to study. Ideally, all known 'helpful' bots or API utilities on the platform are removed before the projection step to keep from storing unnecessary edge information in memory, but this requires a knowledge of the network prior to projection or a reprojection of the network after making these discoveries.

To compare metrics that share similar definitions in the triplet hypergraph and common interaction graph (such as $T(x, y, z)$ vs. $C(x, y, z)$), we make use of a 2D histogram (hexbin) plot. This plot is a type of data visualization that displays the frequency of occurrence for data points in a two dimensional space by representing the space in bins that are colored according to the number of data points that they contain. This can be thought of as a heat map and makes it much more intuitive to understand the density of datapoints for sections within a plot, rather than a simple scatter plot that indicates the presence of a data point. In each of our plots, we display the measured common interaction graph metric for a triplet on the x-axis and the hypergraph metric for the same triplet on the y-axis. The coloring scheme is log-scaled with empty bins left white. The log scaling prevents the extremely high counts for bins at the lower ends of each axis from completely drowning out the rest of the graph. The aim with these plots is to compare the relationships between the hypergraph and common interaction graph metrics, as well as how these relationships may change for different time windows. However, it is important to note that that we cannot make any guarantees on the relationship between hypergraph and common interaction graph metrics, only remark on any patterns that may show in the results. This is due the the lack of concrete bounds between common interaction graph triangle edge weights and hyperedge weights. On all of these plots, we add a blue line to the graph that represents $y = x$, to show where each metric has equal value.

3.1 Results for Analysis of January 2020 Reddit Comment Data

All of the results for this section come from a projected graph where the time window studied was 0 to 60s. While this was not the largest projection that was created by step 1, it was the greatest amount of data that was read by the projection step, with 138 million different

comments reviewed. We begin with by listing some interesting anecdotal findings and then move into comparisons of the concrete metrics defined in our methods.

3.1.1 GPT-2 Language Model Network

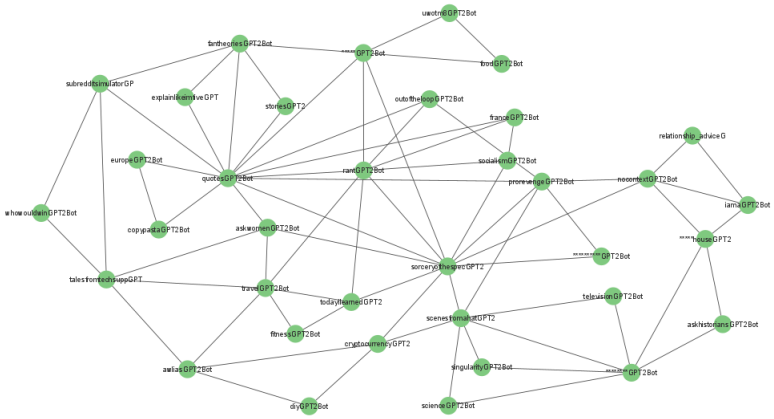

Figure 1: A network of bots using OpenAI's GPT-2 platform for unique text generation.

Using the methods described above, a network of coordinated text generation bots based on OpenAI's ChatGPT-2 platform was found. This connected component, shown in Figure 1, was from the common interaction graph. The framework parameters that yielded this network were a time window of (0s, 60s) for step 1, and a minimum triangle weight cutoff of 25 for step 2. This is one of 39 connected components that are generated from projection and thresholding for these parameters. The edge weights of this network ranged between 33 and 25, with most of the edges having weights on the lower end of the range. This indicates that a lower minimum edge weight combined with a targeted approach could potentially be used to build a more connected network, or a network that captures more of the coordinated users.

Finding this network was seen as a fundamental success for the framework, as the discovery of networks using AI text generation models to mimic human users was one of the core goals of

this project. Furthermore, it demonstrates that the process is agnostic to the content of interaction and relies solely on the spatiotemporal relationships between users.

The behavior of this network is somewhat unique. All of the bots are members of a single subreddit where the bots, and only the bots, can create a page or comment. In this subreddit, there are two distinct types of behavior. The first type is where a bot will create a page and then write many comments on its own page without other bots making comments. Because self interactions are not considered, there is no activity recorded in the common interaction graph for this type of page. The second type of page is a mixed page, where a bot will create a page and comments are authored by the page author as well as other bots in the subreddit. In this case, a subset of bots are chosen randomly from the full set to create comments. Although this would potentially drive the coordination scores of each triplet down because only small parts of the network are being used to comment on each page, we still find that this network has triplets that interact much more often than a random triplet of authors. This could also a reason that the network appears to be more sparse than a share-reshare network.

3.1.2 A Network Distributing Links for Copyright Broadcast Restreaming

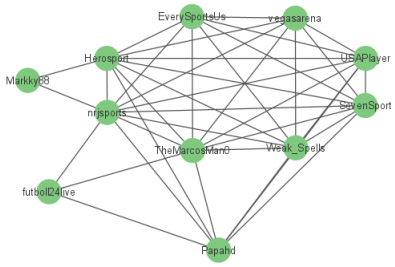

Figure 2: A network of bots redistributing illegal streams of copyright MLB broadcasts.

Figure 2 shows yet another component in the projection of January 2020 data with a time window of (0, 60s) with a minimum triangle weight cutoff of 25. After investigation into the content of the interactions, it was found to be a network of users distributing links to illegal streams of copyrighted sports broadcasts. This type of coordinated link sharing behavior has been shown to be an indicator of malevolent activity by Giglietto et al. [9]. This network has a noticeably different structure to the GPT-2 network due to the different type of behavior. In share-reshare networks, when one member creates a page, all of the other nodes in the coordinated network will immediately interact with it. This leads to dense networks, as shown by the 8-clique that is generated by the main group of users within the network. In addition, these edge weights are typically much higher than the GPT network edges. The weights of these edges range from 27 up to 91.

This was also seen as a notable success because it shows that the unique approach that we take to the projection and common interaction graph reduction which enables the processing of an entire network still highlights the botnets that are targeted by more complex and narrow (in terms of network coverage) approaches by Pacheco et al. [2] and Broccatelli et al. [5].

3.1.3 Comparison Between CI Graph Coordination Scores and Hypergraph Coordination Scores

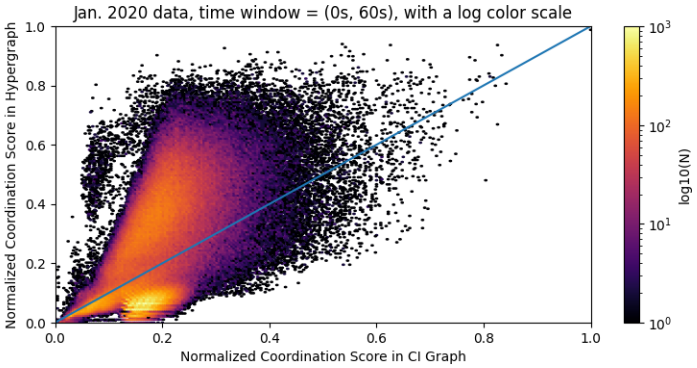

Figure 3: A 2D histogram of triplet coordination scores.

Figure 3 is a 2D hexbin plot of the coordination scores for each triplet that was recorded in the common interaction graph for the 2020 (0s, 60s) projection. The minimum triangle weight threshold that was used to limit the size of the output data was 10. On the y-axis we have the value of $C(x, y, z)$ and on the x-axis we have the value of $T(x, y, z)$ for any given triplet bin. Although there is wide variance in the trend, there appears to be a positive relationship in the values.

3.1.4 Comparison Between CI Graph Triangles and Hypergraph Triplet Weights

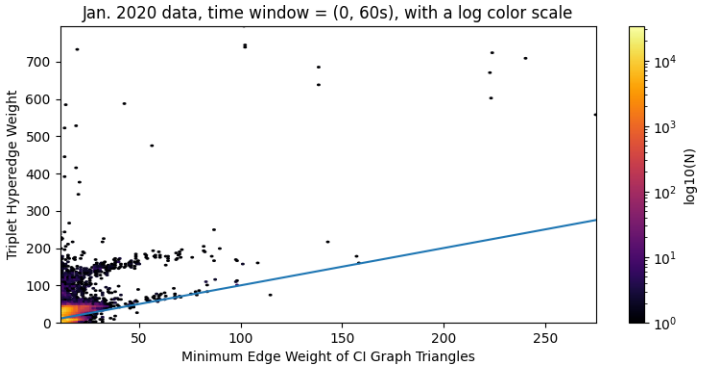

Figure 4: A 2D Histogram of Minimum Triangle Weights and Triplet Hyperedge Weights.

Figure 4 is another 2D hexbin plot, but this plot compares the hypergraph metric w_{xyz} and the minimum triangle weight for each triplet with a weight greater than 10 in the common interaction graph for the 2020 (0s, 60s) projection. On the y-axis we have the value of w_{xyz} and on the x-axis we have the value of $\min\{w'_{xy}, w'_{xz}, w'_{yz}\}$ for any given triplet bin. There again appears to be a positive correlation in the values, but interestingly there appear to be two noticeably different artifacts in the plot. These could represent two distinct types of users or behavior captured, but there is no way to be sure without further investigation into the two groups.

The triangle with the greatest minimum edge weight for this projection had edge weights of (4460, 5516, 13355). This triplet was omitted from Figure 4 in order to better show the rest of the data set. This was from a group of automated bots on the platform that automatically comment

a predetermined message in response to a specific string found in a previous user comment. In this case, the bots were commenting smiley faces ":)" in response to frowns ":(" in previous user comments.

3.2 Results for Analysis of October 2016 Reddit Comment Data

The goals in the analysis of this data were to compare similarities and differences in the analysis results for a network that is smaller but bears a similar structure to that of January 2020, to demonstrate how the relationships between hypergraph metrics and common interaction graph metrics may change for different time windows, and to study a network that has greater value to malevolent actors in influencing platform politics [10].

3.2.1 Projection With a Time Window of 0 to 1 Minute

Here we report the outcomes for the projection of a common interaction graph for October 2016 comments with a time window of (0, 60s). In both plots, we institute a minimum triangle edge weight cutoff of 10 in the common interaction graph to prune the data to a reasonable size that can be displayed.

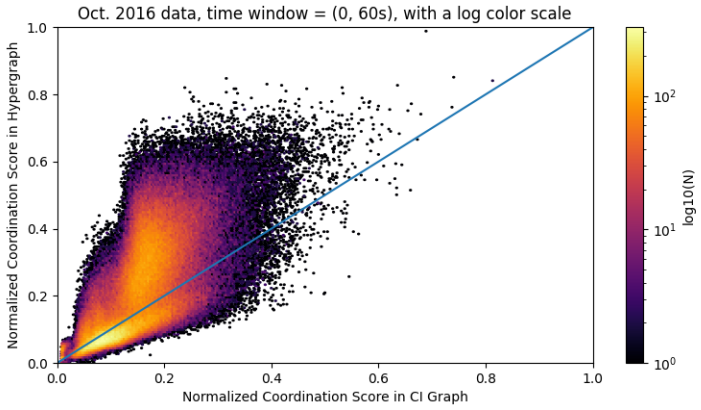

Figure 5: A 2D histogram of triplet coordination scores.

Figure 5 is a 2D hexbin plot of the coordination scores for each triangle that was recorded in the common interaction graph for the October 2016 (0s, 60s) projection. On the y-axis we have the

value of $C(x, y, z)$ and on the x-axis we have the value of $T(x, y, z)$. The analysis that yielded this plot used the same parameters as the analysis for Figure 3, which compares these on January 2020 data. Although there are some differences in the densities for each graph, there are similarities in the distributions for each month of data.

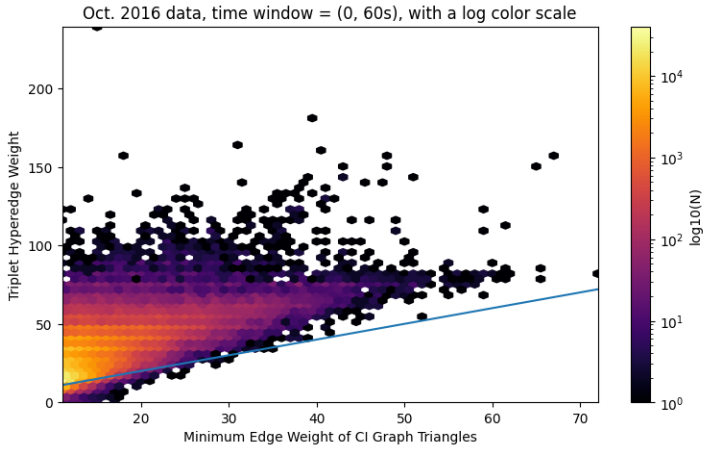

Figure 6: A 2D Histogram of triplet weights in hypergraph and common interaction graph form.

Figure 6 is a 2D hexbin plot of the hypergraph metric w_{xyz} and the minimum triangle weight for October 2016 comments. It is interesting that we do not see the two distinct lines that are apparent in January 2020 data and instead have more defined edges of the distribution.

3.2.2 Projection With a Time Window of 0 to 10 Minutes

Because the time window that produced the common interaction graph for this data is now much longer than that of January 2020, it does not serve as a comparison across the two network instances. However, we can use these plot to understand how the relationships between hypergraph and common interaction graph metrics may change with different windows of time for the same network. Again, we threshold the minimum edge weight for the triangles that we consider with a value of 10.

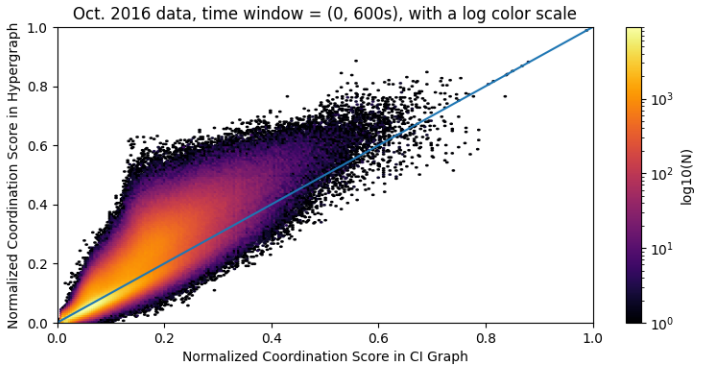

Figure 7: A 2D histogram of triplet coordination scores.

Interestingly, Figure 7 shows a much more cohesive relationship between the two coordination scores $T(x, y, z)$ and $C(x, y, z)$ when compared with the 0 to 60 second projection for 2016. While we still cannot make guarantees, this suggests that a longer time window can be helpful in bringing these two metrics together.

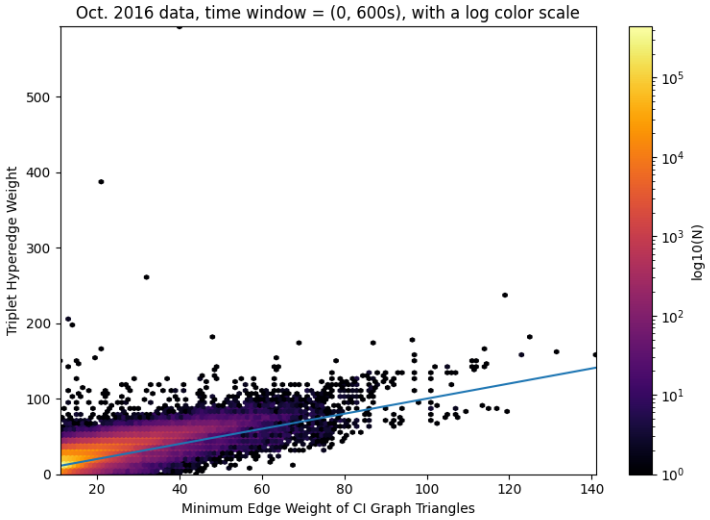

Figure 8: A 2D Histogram of triplet weights in hypergraph and common interaction graph form.

Yet again, Figure 8 demonstrates a closer relationship between hyperedge weight and minimum triangle weight for the a longer time window. However, we do still see many triplets that have a greater hyperedge weight than minimum triangle weight, which highlights the differences in what the two models are capturing. This is reasonable as shared interactions with a page may not happen within 10 minutes of each other and time thresholding is not implemented for the hyperedge counts.

3.2.3 Projection With a Time Window of 0 to 1 hour

Here we report the outcomes for the projection of a common interaction graph for October 2016 comments with a time window of 0 seconds to 1 hour. This was the largest projected graph studied. Before thresholding the edge weights in the common interaction graph, the projection had 2.95 million authors and 3.28 billion edges between them. With an edge weight threshold of 5, 315 million triangles were found. In both plots, we institute a minimum triangle edge weight cutoff of 10 in the common interaction graph to prune the data to a reasonable size that can be displayed. Although this cuts down on the size of the data, we still consider 21.2 million triplets in these graphs

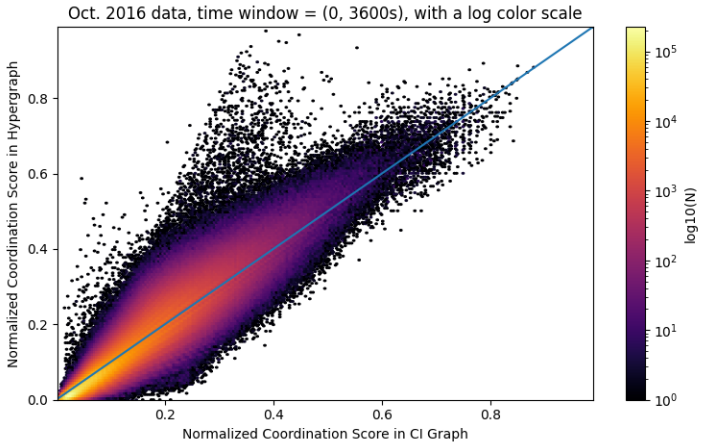

Figure 9: A 2D histogram of triplet coordination scores.

While the time window that produced Figure 9 was 0 seconds to 1 hour, it bears resemblance to Figure 7 which shows the same metrics for a time window of 0 seconds to 10 minutes. Yet again, the larger time window brings the trend closer to the 1 to 1 relationship that is anticipated at the cost of a much larger projected graph and much longer computation time. There may be some point of diminishing returns as we increase the time window; for increasingly large time windows we see less and less difference in the output data.

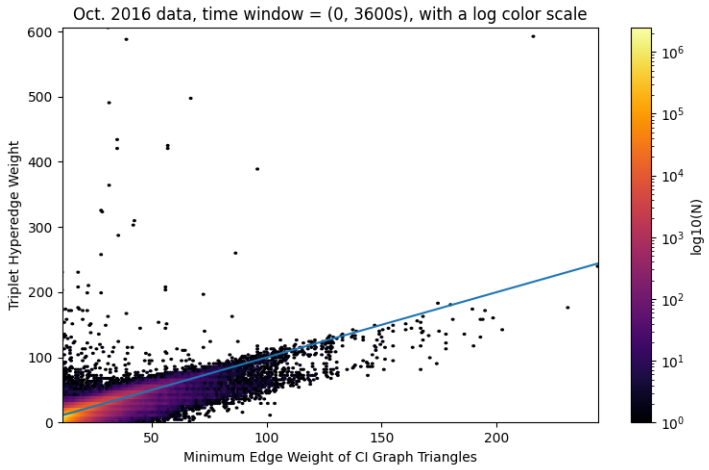

Figure 10: A 2D Histogram of triplet weights in hypergraph and common interaction graph form.

Figure 10 demonstrates how the correlation between hyperedge weight and minimum triangle weight in the common interaction graph changes for an even greater time window. Although the three weight comparison plots show similar trends, we see that the greater time windows capture more pairwise interactions within the the network at the cost of much greater computation time. A way to predict or determine the best parameters has not been studied and may be a good direction for future research.

4. CONCLUSION

4.1 Successes

The success of this project demonstrates the utility of bipartite temporal graphs in modeling user interactions and spatiotemporal relationships they may share in a social network. It has been shown to identify at least two known types of coordination, share-reshare bots and unique text generation bots. Share-reshare behavior is well-documented in literature and remains a problem, as the dissemination of misinformation is aided by networks such as these on Twitter and other platforms. These networks can also be used in the sharing of copyright media that is illegal to redistribute as seen in section 3.1.2. Often these networks can be well targeted with short time windows. The network interactions happen after a 'trigger' event, such as one member making a page containing a link to an untrustworthy article, which is immediately commented on or shared by other users in the botnet.

Another large success of this project is that it is content agnostic, meaning it does not depend on the text of user comments to identify botnets. Because the platform depends solely on the time and location of the page interactions it will be robust against increasingly powerful AI text generation models that will continue to get better at blending in with human authors. This is highlighted in the identification of the GPT2 network found in January 2020 reddit data that is showcased in section 3.1.1.

The final success we want to highlight is the application of these techniques at the scale of the entire social network. This is one of the main contributions of this work. Almost all previous endeavors have focused on specific communities where coordinated behavior is hypothesized, such as 'hastags' on Twitter [2], or studied networks that are smaller than the reddit platform. The use of distributed technologes such as YGM and TriPoll were instrumental in the ability to project these massive data sets and process the billion-edge graphs that were generated for longer time windows.

4.2 Shortcomings

One limitation of this platform is its inability to identify individual bots. However, it is important to understand that this is not the main target of the platform, as this is a process that can be performed by content moderators or existing bot detection methods. We want to identify users working together to cause a large influence in content or discussion, as this is the easiest way for a malevolent party to gain an undeservedly large platform to influence public discourse.

Another limitation of the platform is the size of groups it can find and compute metrics for. Because we limit the search space to triplets of users, we can only asses the coordination of 3 authors at a time. This will allow us to build groups after the fact, but there is no way of directly assessing coordination for groups of more that 3 authors. While this is not a challenge to implement for the hypergraph analysis, it does become difficult to find and enumerate the larger groups in the CI graph. Finding alternate metrics to identify coordination in the projected graph is an area that could be improved with further research.

The final shortcoming is the lack of temporal analysis in the third step of the framework. Because we simply consider a three way interaction with the page as a hyperedge with no limit on the time window in which all three interactions must occur, we lose provable bounds based on the common interaction graph data for the number of hyperedges that can form for a triplet. Although our current definition does indicate the coordination of hypergraph triplets, it cannot say anything about the temporal component of a three way interaction.

4.3 Directions for Future Research

In future research, we hope to address some of the shortcomings mentioned in the previous section. The first direction we hope to take future research is the study of the time-windowed hyperedges. This would allow us to target more specific types of behavior and have provable bounds for the relationship between common interaction graph triangles and triplet hyperedges. We are also interested in using more extensive network analysis tools on the common interaction network to begin the third step of analysis with larger groups of interest.

REFERENCES

[1] J. A. Goldstein, G. Sastry, M. Musser, R. DiResta, M. Gentzel, and K. Sedova, "Generative language models and automated influence operations: Emerging threats and potential mitigations," 2023.

[2] D. Pacheco, P.-M. Hui, C. Torres-Lugo, B. T. Truong, A. Flammini, and F. Menczer, "Uncovering coordinated networks on social media: Methods and case studies," *Proceedings of the International AAAI Conference on Web and Social Media*, vol. 15, pp. 455–466, May 2021.

[3] B. W. Priest, T. W. Steil, R. A. Pearce, and U. N. N. S. Administration, "Ygm," February 2019.

[4] T. Steil, T. Reza, K. Iwabuchi, B. W. Priest, G. Sanders, and R. Pearce, "Tripoll: Computing surveys of triangles in massive-scale temporal graphs with metadata," in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '21, (New York, NY, USA), Association for Computing Machinery, 2021.

[5] C. Broccatelli, M. Everett, and J. Koskinen, "Temporal dynamics in covert networks," *Methodological Innovations*, vol. 9, p. 2059799115622766, 2016.

[6] Z. Wang, T. Hou, D. Song, Z. Li, and T. Kong, "Detecting Review Spammer Groups via Bipartite Graph Projection," *The Computer Journal*, vol. 59, pp. 861–874, 06 2016.

[7] J. Moody, "Static representations of dynamic networks," in *Technical Report*, Duke Population Research Institute, Duke University Durham, 2008.

[8] Z. Neal, "The backbone of bipartite projections: Inferring relationships from co-authorship, co-sponsorship, co-attendance and other co-behaviors," *Social Networks*, vol. 39, pp. 84–97, 2014.

[9] F. Giglietto, N. Righetti, L. Rossi, and G. Marino, "Coordinated link sharing behavior as a signal to surface sources of problematic information on facebook," in *International Conference on Social Media and Society*, SMSociety'20, (New York, NY, USA), p. 85–91, Association for Computing Machinery, 2020.

[10] T. Khaund, B. Kirdemir, N. Agarwal, H. Liu, and F. Morstatter, "Social bots and their coordination during online campaigns: A survey," *IEEE Transactions on Computational Social Systems*, vol. 9, no. 2, pp. 530–545, 2022.