Network Graph Analysis of Followers Among Twitch Streamers

Will Howe
whowe1@jhu.edu

Abstract

Twitch.TV is an online streaming platform which in recent years has branched out from its original design as a gaming website into news, politics, and entertainment streams. Unlike YouTube in which creators upload edited content, Twitch streamers' content is broadcast live. Due to the uniqueness and novelty of the platform, organic communities and charismatic figures emerge. Network graphs are useful in identifying similarities and differences between large streamers' communities. Since viewers are a finite resource, large streamers tend to form connections and rivalries with other streamers. Network graphs may provide insight into whether a streamer is at odds with another or whether they have formed an alliance of sorts.

1 Introduction

Prior to the 2016 Presidential Election, the alt-right gained a footnote on YouTube, utilizing the platform to spread ideologies ranging from anti-PC and anti-identity politics to outright fascist (Lewis, 2018). Through a careful analysis of which content creators appeared on other creators' channels on YouTube, (Lewis, 2018) managed to build a network graph of the key figures leading the alt-right movement on YouTube and enabled an understanding of how such a movement was able to gain a foothold so quickly. The YouTube algorithm as a result of its design tended to recommend to users who had ventured into the alt-right web, other members of the alt-right.

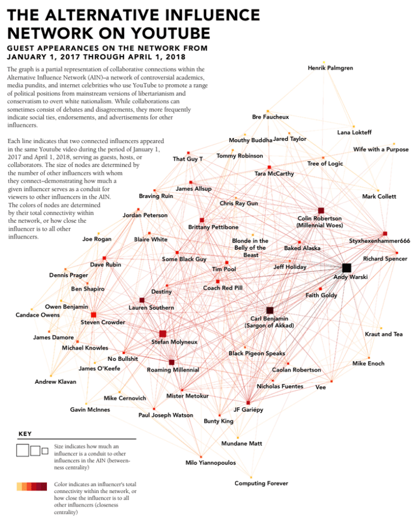
Almost in response to this phenomenon, the Twitch platform has experienced an increase in the number and size of leftist streamers. While many streamers develop positive relationships with each other and focus on community building, others start conflict and engage in so called debate-lord tactics which enable a debater to win a debate even if their arguments aren't logical (e.g. gishgalloping, ad hominem). Just as (Lewis, 2018) effectively utilized a network analysis of alt-right YouTube communities, a similar approach might be useful for studying communities on the Twitch platform.

2 Methodology

2.1 Graph Representation

To understand network interactions on the Twitch platform, there are several attack vectors. Perhaps the most effective method would be to follow (Lewis, 2018) and hand-curate a dataset using a finite-list of streamers and build an edge between two creators when one appears on the other's stream. While this would produce a very useful graph, it's not feasible for this project. In the absence of such a dataset, the next best option would be to track which users a streamer subscribes to. The *subscribe* relation on Twitch is unlike YouTube in that it costs five dollars (or more for other tiered subscription) to subscribe to a streamer. Unfortunately it is possible for streamers to hide their subscribers and the Twitch API does not make this information readily available. Another viable method would be to look at who *hosts* who on the platform. The hosting relation on Twitch means that one streamer allows another streamer's stream to broadcast from their own stream while they themselves aren't streaming. Similar to hosting is *raiding* where a streamers sends all their viewers to another streamer. These all would be good options for building edges between nodes, but they are not made available by the API or are not web-scrapeable.

- ~~Who appeared with who?~~ - Not feasible
- ~~Who subscribes to who?~~ - Not made available
- ~~Who hosts who?~~ - Not available


THE ALTERNATIVE INFLUENCE NETWORK ON YOUTUBE

GUEST APPEARANCES ON THE NETWORK FROM JANUARY 1, 2017 THROUGH APRIL 1, 2018

The graph is a partial representation of collaborative connections within the Alternative Influence Network (AIN)–a network of controversial academics, media pundits, and internet celebrities who use YouTube to promote a range of political positions from mainstream versions of libertarianism and conservatism to overt white nationalism. While collaborations can sometimes consist of debates and disagreements, they more frequently indicate social ties, endorsements, and advertisements for other influencers.

Each line indicates that two connected influencers appeared in the same Youtube video during the period of January 1, 2017 and April 1, 2018, serving as guests, hosts, or collaborators. The size of nodes are determined by the number of other influencers with whom they connect–demonstrating how much a given influencer serves as a conduit for viewers to other influencers in the AIN. The colors of nodes are determined by their total connectivity within the network, or how close the influencer is to all other influencers.

KEY

Size indicates how much an influencer is a conduit to other influencers in the AIN (betweenness centrality)

Color indicates an influencer's total connectivity within the network, or how close the influencer is to all other influencers (closeness centrality)

Figure 1: Network graph from (Lewis, 2018) on YouTube alt-right community

- ~~Who raided who?~~ - Not available
- **Who follows who?** - Feasible and available

The *who follows who* relation which is analogous to the subscription relation on YouTube is available on the Twitch API. Unlike the other metrics, though, followers are much more numerous than subscribers which can quickly cause a explosion of nodes in the queue of a webcrawler.

2.2 Procedure

Originally I set out to perform web-crawling on `https://twitchfollowing.com/` which allows the user to enter the screen name of a Twitch streamer and then display all the users they follow. This analysis focuses on big streamers and such streamers typically follow other big streamers – the mass of viewers on the platform won't be picked up by a crawler since large streamers don't typically follow their viewers – there would be too many to follow.

However, when I tried to crawl the above site with Selenium it was far too slow due to a lagging Javascript script which called the Twitch API. Bypassing this middle, I decided to do a crawling of sorts on the Twitch API. I allow a user to input a list of their favorite streamers and then I utilize API calls to get a list of the users that a given streamer follows. This is on the order of magnitude of one hundred. I provide users with two features:

- **Similarity:** If the user inputs two streamers, I can return the intersection of followers between them (i.e. the list of people that they **both** follow)
- **Network graph:** Given a list of streamers, I build a graph where each edge in the graph goes from a streamer in the user inputted list to each of the people they follow, representing the follow relation. Crucially, I only perform a depth 1 search since otherwise the node set of the graph explodes. This works best with a small number of streamers ($n < 4$).

My crawler crawls through the Twitch API and returns a list of edges from the target streamers to the people they follow. I then build a graph using the Networkx Python package for graph processing. I export my graph into the GraphML language for visualization in the Graphia graph processing software.

3 Results

Since large streamers follow somewhere on the order of magnitude of one hundred other streamers, network graphs explode very quickly. However, I generated graphs for n=2, n=3, and n=4 streamers. Intuitively, the network graph can differ in size depending on the size of the streamers provided by the user (some smaller streamers follow less than 100 people).

Crucially, my network graphs provide a useful means of interacting and probing with streamers and their communities, visualizing the size of communities and clusters between large streamers, and analyzing streamers ideologies by means of who they follow.

3.1 Qualitative Analysis of First Graph

After exporting a GraphML file to the Graphia program, I reduce the size of the nodes and edges in my graph as well as the size of the text font (down to 6pt) to enable better visualization. Nodes are sized according to their degree such that the initial user inputted seed nodes have the largest size in the graph. I apply the Louvain community detection algorithm which colors the nodes of each cluster (Blondel et al., 2008). Strictly qualitatively it is clear that the streamers *Destiny* and *HasanAbi* both follow a similar number of other streamers with HasanAbi following slightly more than Destiny. *HasanAbi* is a leftist politics streamer while *Destiny* focuses on video games, philosophy, and neoliberal politics. This information appears to be

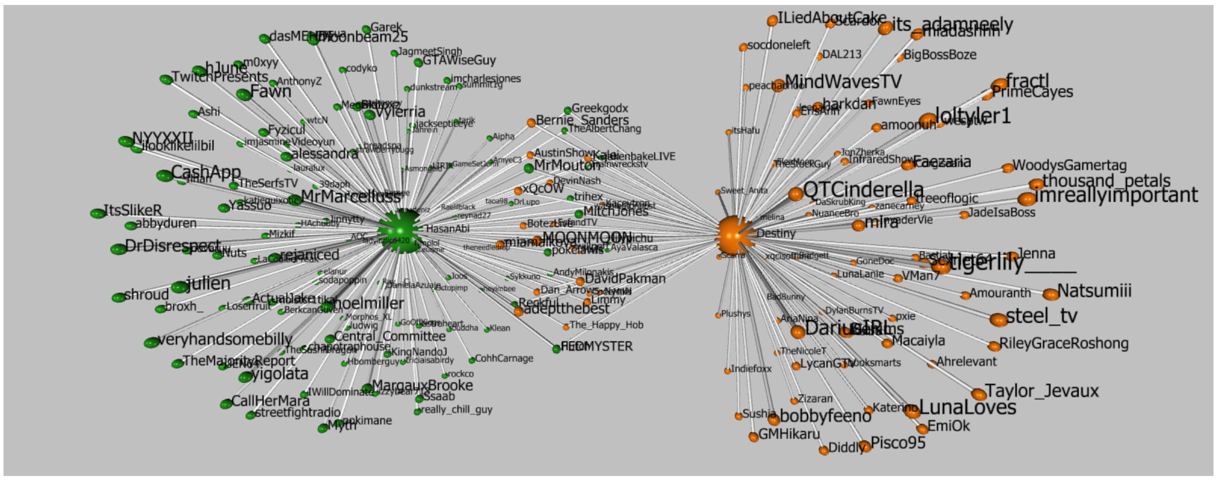


Figure 2: The network graph between who *Destiny* (orange) follows and who *HasanAbi* (green) follows.

in accordance with my network graph. For example HasanAbi follows a socialist streamer named Central_Committee that Destiny does not follow, while Destiny follows a more right leaning streamer, TreeofLogic who HasanAbi does not follow. They both follow Bernie_Sanders campaign stream. Though they have their own interests, there is a substantial cloud in the network between the two main poles signifying that there is significant overlap between the two streamers and leaving the possibility open that they might agree ideologically.

3.2 Qualitative Analysis of Second Graph

For my second graph I applied roughly the same techniques as the first graph, though I don't display node labels in my figure. For this graph though, I have increased the number of user inputted streamers from two to three thereby increasing the size of the graph. In this graph *BadBunny* has a large number of people she follows. *Central_Committee* trails shortly behind her in people he follows while *VaushVidya* follows comparatively few other streamers.

All three of these streamers are in the politics

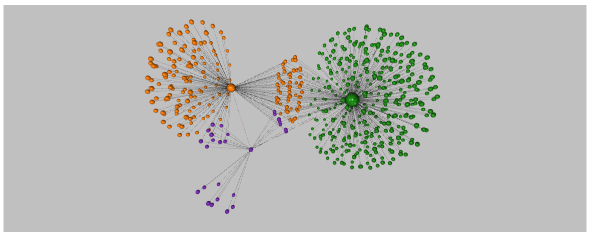


Figure 3: A network graph between *VaushVidya* (purple), *BadBunny* (green), and *Central_Committee* (orange).

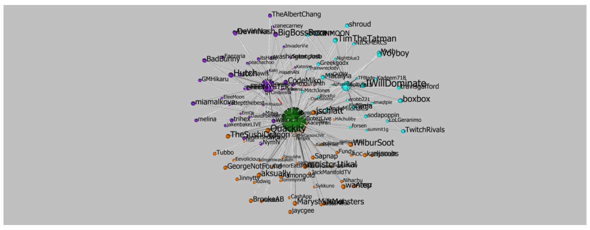


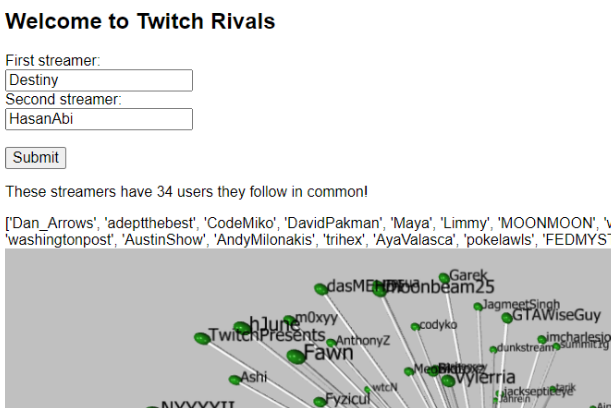
Figure 4: LolTyler1 (blue), BotezLive (green), Destiny (purple), Kaceytron (orange)

category and the green and orange clusters overlap significantly. The purple clusters overlaps with both the orange and green clusters, but much less so due to its smaller size. Interestingly, this graph might provide useful insights for VaushVidya who might be able to grow his community by leeching off streamers in the other two clusters and organizing collaborations with them.

3.3 Qualitative Analysis of Third Graph

For my third graph, I increase from three seed nodes to four which totally explodes the graph size. Here I take a new approach and remove all nodes whose in-degree is less than two in the Graphia software program. This means that all remaining nodes are followed by at least two streamers in the input list.

Following this procedure yields some interesting results. Kaceytron, who does entertainment and video games, Destiny (politics, video games), and LolTyler1 (video games) each retain distinct communities. However, the chess streamer BotezLive gets a cluster of her own – this is because she follows many streamers from each of the other communities, but doesn't have a group that she fol-


Welcome to Twitch Rivals

First streamer:
Destiny
Second streamer:
HasanAbi

Submit

These streamers have 34 users they follow in common!

['Dan_Arrows', 'adeptthebest', 'CodeMiko', 'DavidPakman', 'Maya', 'Limmy', 'MOONMOON', '
'washingtonpost', 'AustinShow', 'AndyMilonakis', 'trihex', 'AyaValasca', 'pokelawls', 'FEDMYS

Figure 5: The page result of a query to my web app. The query returns the followed streamers in common between the two inputs. Ideally, a graph like those above would be generated on the fly.

lows that is distinct from the other communities. In fact, BotezLive has the highest out-degree (105) than the other seeded streamers who have 99, 54, 31 respectively. In some sense BotezLive has a finger in every pie.

4 Web App

To demo my work, I built a toy web app in Flask, a python framework for client-server applications. This app allows users to input the names of two streamers and get back a list of common streamers they both follow. In the future, my app will build network graphs like the ones presented in this paper, but this functionality is not yet implemented.

5 Code

The Google Colab notebook used to build my web crawler and graphs as well as my Flask App are hosted on a Github repo: `https://github.com/ahowe444/info_retrieval_project`

6 Conclusion

In my project I used a web crawler, graph processing techniques, and a web application to analyze network graphs on Twitch. Just as prior work helped shed light on the growth of political movements on YouTube, I hope similar techniques can be applied to Twitch enabling new insights. For future work I plan on submitting a pull request to the owner of `https://twitchfollowing.com/` for a compare feature between two streamers which is the main feature of my web app.

References

Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. 2008. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):P10008.

Rebecca Lewis. 2018. Alternative influence: Broadcasting the reactionary right on youtube. *Data & Society*, 18.