Video Game Community Analysis - Destiny Clan Networks

Names: Thaminda Edirisooriya Hansohl Kim

Project Introduction

Online multiplayer gaming is a phenomenon that continues to grow in popularity. Within these games, players tend to form groups to work and play together. We are interested in analyzing how these groups form, and what kind of players tend to pull other into these groups, because in doing so, games could direct newer players or those looking for a group towards people who are likely to take them in, thus improving their experience in-game.

The subject of our project is the game Destiny - a first person shooter massively multiplayer online game. It features a variety of player versus AI and player versus player activities for users to partake in. Users can form clans, or groups, and users also have their own individual list of friends they can create and maintain. When players enter an activity, they either choose to do so with other players with whom they mutually agreed with to play together, or are randomly matched with other players in a similar stage of the game who is also interested in the same activity. The game has no inherent method of communicating, but once players are in a game with other players, they can use voice chat to communicate.

We built networks through a crawl of Destiny players for the activities they participated in. The networks reflect how many times players (nodes) interact with each other through the various in-game activities. We analyze the community structure and connectivity of these node to predict a measure of how good of a teammate a particular player is. This is a worthwhile problem to solve because it would allow players searching for teammates to have a predictor of how good their experience would likely be with player X. This could be used in conjunction with Destinys third party matchmaking tools which attempt to match players interested in completing a difficult activity, or those just looking for new people to play with.

In the process of analyzing the player interaction networks we also used the network to see how often official player clans - communities that players self-elect to join - match up to predictions made by community detection techniques. We compared the obtained clusters to the official clan listings made available through Bungie's API, allowing us to evaluate how the influence of clan membership manifests in actual day to day player interactions.

Related Work

One of the inspirations for much of the work weve done comes from Integrating and Inspecting Combined Behavioral Profiling and Social Network Models in Destiny. The authors focus on 3 different tasks - they query the full list of all Destiny players and calculate some attributes relating to the cluster of player they belong to, they do a crawl starting at a few players and exploring their friends list to create a connected component to analyze, and finally, they randomly sample 10,000 players of the game and attempt to cluster the players based on a set of dimensions they chose.

The analysis performed in this work was the clustering of players of the game into groups based on their behavior in player versus player competitive game modes, where players work with their team, against another team of players, to achieve some objective. The features chosen to perform the clustering were factors like kill/death ratio, score, and weapon preference. Through performing the clustering, 5 groups were created, which ended up somewhat separating players by skill, and by weapon preference.

The authors looked for player type clustering assignments to extend to friend networks. According to their analysis, there is little correlation between the player type clustering they performed, and the types of players that tend to be friends and play with each other. The visualizations they created serve to display nodes in the two smaller networks, as well as the player type. In the friend network visualization, player type within small clusters of friends does not seem to match or follow a pattern, confirming the analysis.

The authors didnt do much with the data they gathered besides visualize it, and note that weapon preferences and general player type did not seem to be affected by the connections in the graph. As avid players of the game, we thought to see if we could come up with deeper findings and patterns from the data we collect, to try and test some hypotheses we have as players about the general community as a whole.

We chose to use the clustering method (Clauset-Newman-Moore) described in Finding community structure in very large networks. It proposes a method for efficient community structure detection within particularly large networks. While many known community detection techniques exist, the runtimes of algorithms can sometimes become prohibitively long for larger graphs. The authors propose an optimization of a previously published greedy algorithm for community detection that reduces the runtime from $O((m+n)n)$ to only $O(md \log n)$ where d represents the height of the "dendrogram" of community structure. The paper consists primarily of a description and justification for the new algorithm, which involves repeatedly merging communities greedily.

Analysis Methods

Clustering Methods

Girvan-Newman clustering is a popular and effective way to cluster small graphs. The algorithm is based on betweeness centrality, and progressively removes bridge edges with high betweeness to obtain isolated clusters. However, this algorithm runs in $O(n^2m)$ time and is therefore inefficient for large graphs.

As mentioned previously, Clauset-Newman-Moore clustering is relatively efficient for large networks, and we chose to employ this method as well, including for larger graphs with over a thousand nodes. CNM clustering optimizes for modularity as a metric, unlike Girvan-

Newman which is based on betweeness centrality.

The third clustering method we experimented with was spectral clustering, which is based on the eigenvectors of the graph Laplacian. We implemented a variant of eigenvector clustering to produce multiple clusters.

Centrality Measures

To measure node centrality within our graph we experimented with three different metrics: degree centrality, betweeness centrality, and eigencentrality. Eigencentrality is similar to the PageRank algorithm discussed in class, which is a variant of it. Eigencentrality is defined in a recurrence, as connections to nodes with high eigencentrality contribute more than connections to those with low eigencentrality.

To determine what made a user a good teammate, we used centrality measures and looked for patterns in the measures that seemed to indicate deeper meaning than just having many edges. We use a weighted combination of degree centrality and betweeness centrality can be used to try and pick the nodes who play with a lot of people consistently, including those outside their closer friend groups consisting of clan members. We do the weighting as follows:

1. Create two sorted lists of nodes, sorted in descending order by their betweeness centrality and degree centrality

2. Define r_{id} as the position in the degree centrality list of node i, and r_{ib} as the position in the betweeness centrality list

3. Define $w_d = 1.0$ and $w_b = 3.0$. These are weights which we assign to the ranking for each of the centrality measures

4. Compute the score $s_i = r_{id}^2 * w_d + r_{ib}^2 * w_b$ for each node, and sort them based on this score. This is our final ranking of teammate quality for each node in the graph.

Data Collection

We acquired our data using the public bungie.net API. Bungie is the developer of Destiny, our topic of interest, and they allow anyone to query their API to get information about individual players, activities, and other data.

We start the process at a single player, and we acquire its unique identifier in the database, and from that the list of all recent activities they participated in of a given type. These activity types range from large Player vs. Environment (PvE) activities, to smaller Player vs. Player (PvP). Finally, from each activity, we can see which players participated, as well as how each player performed. We use this for the list of players, and thus we get a set of edges describing who the query player played with. We then recurse to a depth of our choosing to some subset of the players we touch, and build our network by storing the edges, along with weights for each edge describing the number of times two players participated in an activity together. The selection of players to recurse on can be sampled with the edge weights as a sample weighting. With the list of players and edges computed, we can load everything into SNAP and perform the analysis of our choosing.

Results and Findings

We have collected data and built networks for four game activities: 6-player PvE raids, 3-player PvE missions, 6-player PvP matches, and 3-player PvP matches. All activities aside from the 6-player PvP matches do not support in-game matchmaking and require players to actively search for and form teams. Each network was built from the same initial player. After producing the network graphs, all nodes with degree 1 were repeatedly removed. This allows us to focus on interactions that are beyond a chance one-time meeting. In addition, many of these nodes were added at the very end and we have not had a chance to properly explore their neighborhoods.

All four graphs exhibited significant clustering. Basic graph information is represented in the table below. The clustering coefficient is displayed below p, the expected clustering coefficient for an Erdos-Renyi random graph with the same average degree and number of nodes. Spectral clustering results were generally useless compared to GN or CNM clustering, for which modularity evaluations are displayed.

Table 1: Summary from player interaction networks in four game activities

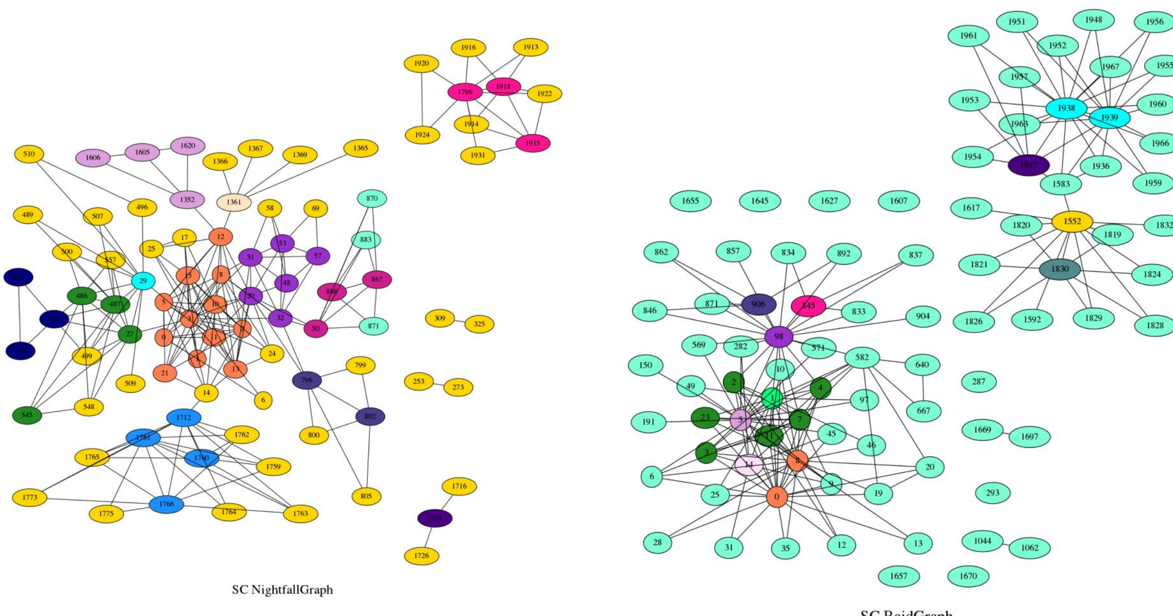
	Raid (6 PvE)	Nightfall (3 PvE)	Iron Banner (6 PvE)	Trials (3 PvE)
Matchmaking	No	No	Yes	No
Nodes (n)	192	207	794	1187
Edges (m)	451	486	1610	2631
Avg k	4.698	4.696	4.055	4.433
p	0.024	0.022	0.005	0.012
Cluster Coeff (c)	0.777	0.743	0.610	0.730
Modularity (GN)	0.710	0.753	N/A	N/A
Modularity (CNM)	0.699	0.761	0.772	0.801
Edge Bridges	0	5	2	0

These graphs display strong similarities in average node degree, clustering coefficient, and modularity, implying player interaction dynamics are similar across activities within Destiny. There are slight differences among the activities that are interesting to note. Iron Banner, for example, has the lowest clustering coefficient, which may be explained by the random introduction of connections through its matchmaking system. However, in comparison to p, the Iron Banner graph has extremely strong clustering. In addition, the modularity of the

Iron Banner graph is similar to the average among activities. The Iron Banner graph does seem to exhibit a more balanced degree distribution among nodes, however. This may indicate at least some influence from the matchmaking system, increasing connections between clustered communities. Inspection of the graph visualization of the Iron Banner network also supports this, as a near-uniform web-structure can be observed in portions of the graph, replacing the sharp concentrations of edges in the others.

By clustering coefficient, Raid and Nightfall are highly clustered and are generally accurate predictors of actual clan membership, as we will soon discuss. The Trials graph, on the other hand, is strongly clustered but is a poor predictor of clan membership using our clustering methods, as members of the same clan are often split across clusters. The primary difference between the Trials graph and the Raid/Nightfall graphs is that it is PvP. Furthermore, Trials, like Iron Banner, is considered a competitive PvP mode. This may suggest that some players involved in Trials rely on clustered communities beyond their clans. Since Trials can be highly competitive and more strongly skill-based, players may search for highly rated PvP partners online. In contrast, Raid and Nightfall PvE activities have a set difficulty level and generally rely more on group coordination and goodwill between players.

Clustering and Community Detection

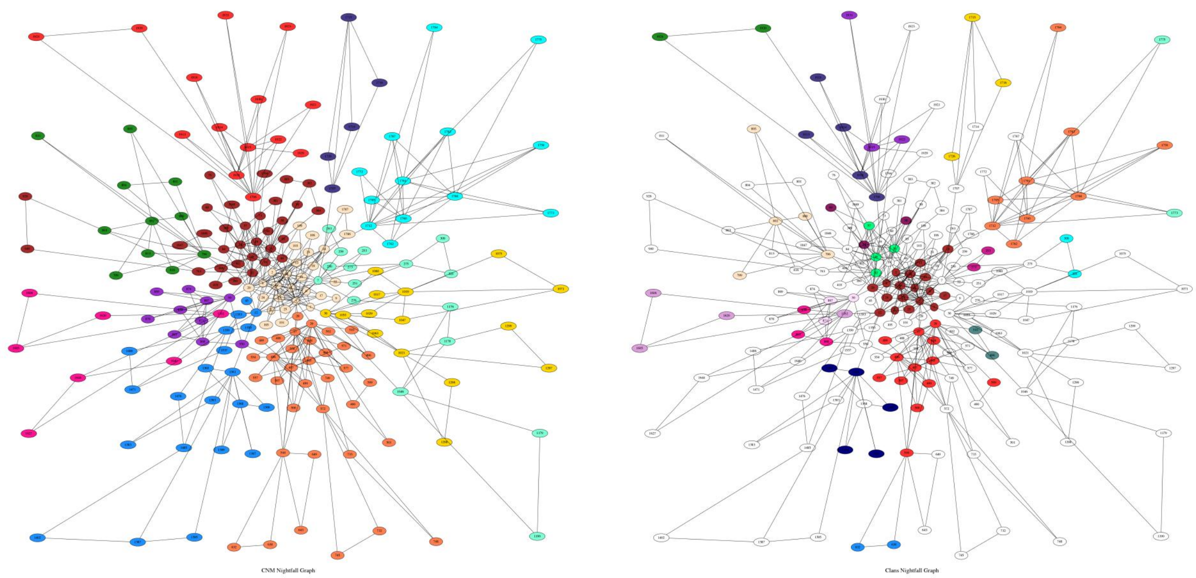


(a) Spectral Nightfall Clusters

(b) Spectral Raid Clusters

Figure 1: Spectral clustering found cores of clans, but treated most of the graph as a "remainder" category, seen here in yellow and teal

For many of the nodes in these graphs, clan membership is either private or does not exist. For those with accessible information, we used actual clan membership as one metric of our clustering results and focused on the Raid and Nightfall activity graphs. Interestingly, many of the players in our networks were the only node representing their clan. We treated such clans as trivial, in that any clustering would "correctly" group them. For clans with more than one representative, we measured whether clan members were divided across clusters. Spectral clustering proved mostly ineffective. While it was able to identify the cores of densely clustered clans, it treated the vast majority of graphs as a single remainder cluster as seen in Figure 1.

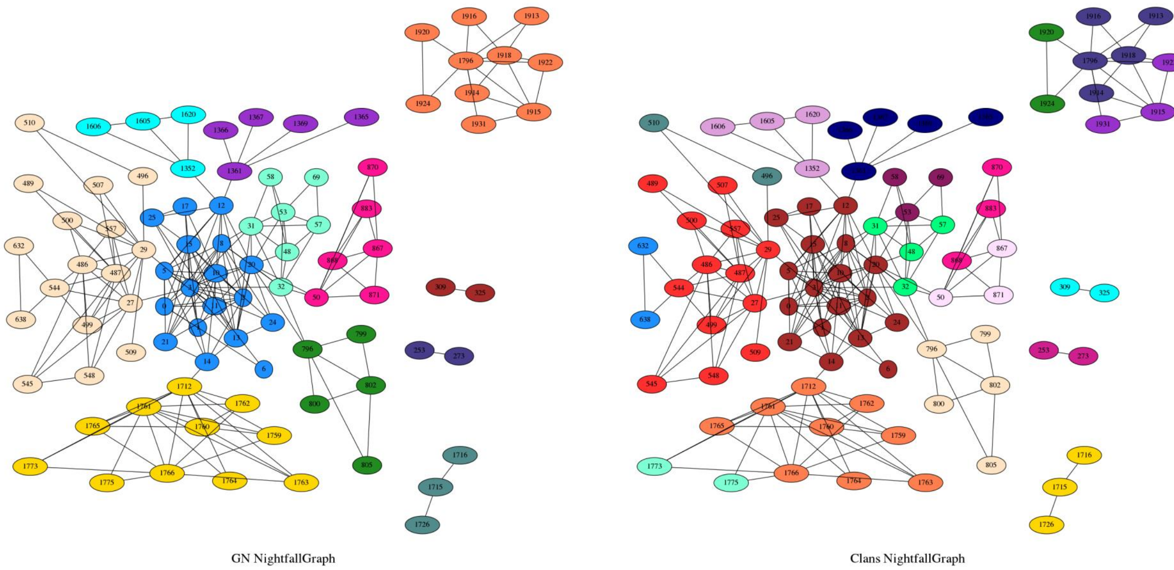


(a) CNM Nightfall Clusters

(b) Actual Nightfall Clans

Figure 2: Nightfall activity

GN and CNM clustering proved to be mostly identical for the Raid and Nightfall graphs, resulting in superclusters of interacting clans. They were also both highly accurate relative to actual clan membership: GN did not split any clans while CNM split a single clan member (who was highly connected to other clans). Figure 2 exhibits the correspondence between the clusters generated by CNM and actual clans, while Figure 3 is a graph of the connections between members of non-trivial clans only. Figure 3 also displays the GN clustering for Nightfall, which is almost identical to CNM with the exception of node 12. This view also emphasizes the interactions between known clans, and displays some level of clan superclustering, potentially indicating clans that tend to play together more frequently. We note that many of the clusters still correspond to individual clans, however.



(a) GN Nightfall Clusters

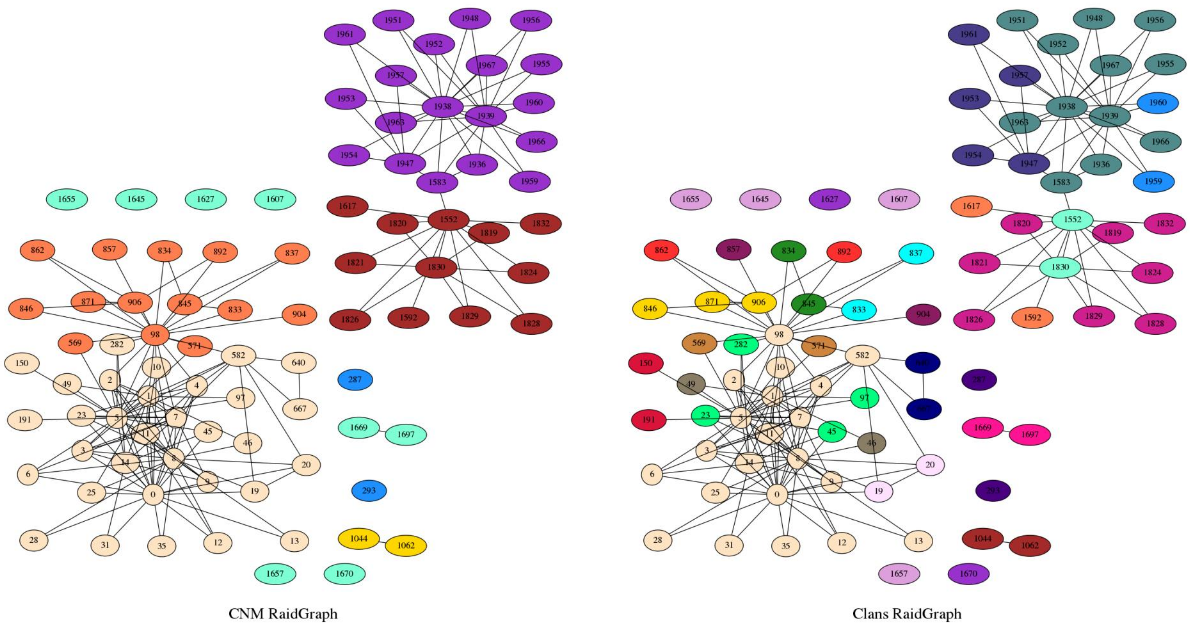
(b) Actual Nightfall Clans

Figure 3: Analysis of connections among known non-trivial clans only

GN and CNM generated identical clusters for the Raid graph. The clustering was performed over the full graph, but for clarity we present the colored graph over non-trivial clans only in Figure 4. Clan superclustering is highly visible. Despite having a similar number of nodes, the raid network exhibits far more interclan interaction.

The accuracy of both (non-spectral) clustering methods in both activities implies that clan dynamics have a real and strong impact on player interactions in Destiny. It also suggests that it may be possible to predict with reasonable accuracy the clans for the blank nodes in our graphs with unknown clan membership. While many of these players may not be officially associated with any clan, some have privacy settings obscuring the knowledge. It may also be possible to predict which clans a player is likely to join based on these clusters. The clusters of these simple player interaction networks therefore appear to be significant and highly correlated with clans in Destiny.

The prevalence of clan superclusters is also noteworthy. While Nightfall (3-player PvE) clusters correspond primarily to individual clans, Raid (6-player PvE) clusters correspond primarily to super-communities of multiple clans that frequently interact and cooperate through multiple points of contact. This may reflect the logistical difficulties in organizing a larger group to tackle a longer (usually several hours) activity, resulting in cross-clan cooperation. The fact that these super-clans can be strongly clustered implies that the external members a clan may call upon for Raids are not random, and tend to strengthen ties between clans and form regular reserves that a clan might contact. As an example, we note two clans, "Having a Time" and "Etched in Blood", that are grouped together in the same supercluster for both the Nightfall and Raid graphs, implying strong and consistent connections that carry over between activities.



(a) CNM Raid Clusters (b) Actual Raid Clans

Figure 4: Analysis of connections among known non-trivial clans only

Teammate Quality Ranking

While there are no good empirical methods to measure how good of a teammate someone is to use as a standard to compare to (thus the reason for the method we developed), we were able to compare our results against rankings posted by DestinyTracker, and independent website that computes a ranking for each player based on a multitude of different factors, such as performance in activities, number of activities completed, and other hidden values.

When we examine the top users as selected by our algorithm, we note that all of them scored very highly on the DestinyTracker website - often in the top 10% of all players of the game. Furthermore, when we examine these nodes on a visualization of our graph, we do see that they are both connected to many other players, and seem to bridge many distinct groups of players together.

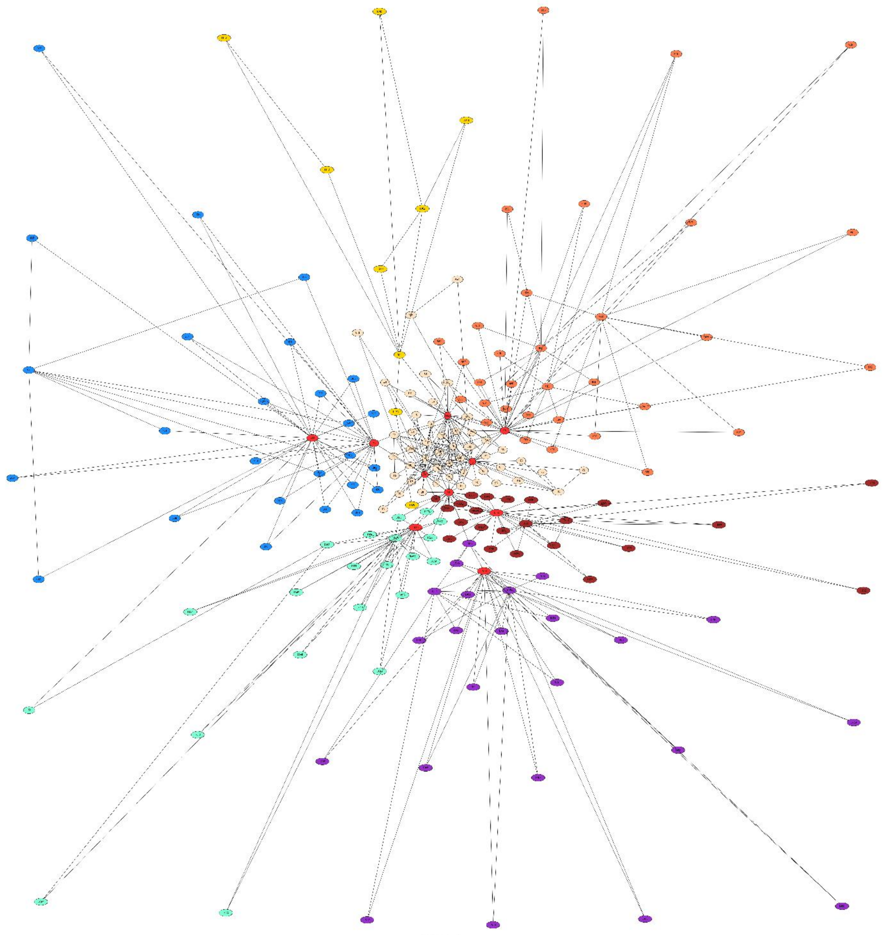


Figure 5: CNM clustering of Raid, our top-ranked picks are highlighted in red and occupy central yet also bridging positions

Name	Rank	DestinyTracker Score
Prihmal	1	55
neoleopard	2	97
overcooked_cat	3	97
Cthulhuhoop_	4	30
xXBEASTXx987	5	76
kath_in_md	6	88
aayjay	7	96
Dormous	8	53
Pork_Chop_ed1	9	71
DUCKY__4	10	22

We found that betweeness centrality captured a lot of what we were looking for: users who seem to connect large groups of people together with high weight. When computing it over the graph, a few nodes in particular exhibited extremely high betweeness centrality, and when the graph was visualized, appeared as either nodes which existed between many nodes, or as nodes within clan clusters. However, we noted that nodes that exist with high betweeness, but only within a cluster of clanmates, are not necessarily the best teammates, since they might not be as willing to play with other players and explore the rest of the network. By combining the two in our ranking, we achieved results that, as much as we are able to evaluate, appear sound.

From our analysis, we can conclude that it is indeed possible, at least within the scope of Destiny, to both predict players that may be good teammates, as well as predict the clan boundaries, just from activity data. We also built a system for scraping the Destiny player network efficiently that we could further use to explore the network for other patterns.

References

[1] Rattinger, Wallner, Drachen. Integrating and Inspecting Combined Behavioral Profiling and Social Network Models in Destiny, 2016.
https://www.researchgate.net/publication/307477035_Integrating_and_Inspecting_Combined_Behavioral_Profiling_and_Social_Network_Models_in_Destiny

[2] Clauset, Newman, Moore. Finding Community Structure in Very Large Networks, 2004.
http://snap.stanford.edu/class/cs224w-readings/clauset04community.pdf

Contributions

Thaminda: Network Crawl, Teammate Ranking
Hansohl: Multithreading Support, Clustering Analysis
Both of us worked on the report/milestone writeups