Reddit Network Analysis

Subreddits interactions over the time period 2016-2017

Network Science Project, A.A. 2022/2023

Anthony Palmieri
anthony.palmieri@studenti.unipd.it

1 Introduction

In this report, we analyze the network structure of Reddit, focusing on interactions among subreddits during the period 2016/2017, time period particularly significant since it lead to the election of Donald Trump as President of the United States.
The election of Donald Trump was a significant event in American and world politics, and it generated a lot of discussion and debate across the U.S. and the globe, both offline and online. Reddit, being one of the largest online communities, was no exception.
The website is organized into communities called "subreddits", which cover a wide range of topics, from news and politics to hobbies and entertainment. Each subreddit has its own subscriber base and can be seen as separate community within the larger Reddit platform.
Using network data spanning from 2016 to 2017, our goal is to uncover any changes in the organization and dynamics of the subreddit interactions, especially regarding Trump-related subreddits, which we expect to play a relevant role given the political context of the period.

2 The Dataset

The dataset can be found on SNAP at the following link.
It is extracted from publicly available Reddit data of 2.5 years from Jan 2014 to April 2017, although for computational reasons we will be focusing solely on data from 2016/2017.
The dataset represents a Directed **Multigraph** where the nodes are subreddits and the multi-links are hyperlinks between such subreddits.
Specifically, hyperlinks are extracted from posts in the source community that link to posts in the target community.
Each hyperlink is annotated with three properties: the timestamp, the sentiment of the source post and some of its text properties.
Unfortunately we had to discard such properties in order to merge the links and turn the Multigraph into a Simple Graph.
In particular:

- we discarded hyperlinks associated with negative sentiments since they constituted less than 10% of the total number of edges;
- we discarded text properties of each hyperlink.

In this way we were able to produce a Directed Weighted Simple Graph where the weight of each link corresponds to the number of hyperlinks between the two subreddits.
One last further simplification was necessary: given the large structure of the network, we decided to turn it into an undirected network, to lighten the computational load. In particular we define the weight w^u_{ij} of the undirected link between nodes v_i and v_j as the minimum between the weights of the related directed links:

$$w^u_{ij} = w^u_{ji} = \min(w^d_{ij}, w^d_{ji}).$$

3 Network Overview

To begin our analysis we briefly report some basic network statistics:

#nodes	#edges	#self-loops	# C.C.	size G.C.	density G.C.
7969	13179	0	511	6666	12355

We note that the network is very sparse, a fact not particularly surprising considering the very nature of subreddits, which in general tend to be rather self-contained communities.
The size distribution of the connected components is shown in the table below:

size	2	3	4	5	6	7	8	9	10	19	62	6666
#	404	58	24	7	7	2	2	3	1	1	1	1

As we can see all the Connected Components except the giant one are very small, the majority of them consisting of only 2 nodes. This is the reason why we decided to restrict our analysis to the Giant Component only.

3.1 Degree Distribution

We begin the actual analysis by looking at the degree distribution of the nodes in the network. In the following we report the plots for: degree distribution, degree distribution in log-log scale and CCDF in log-log scale.

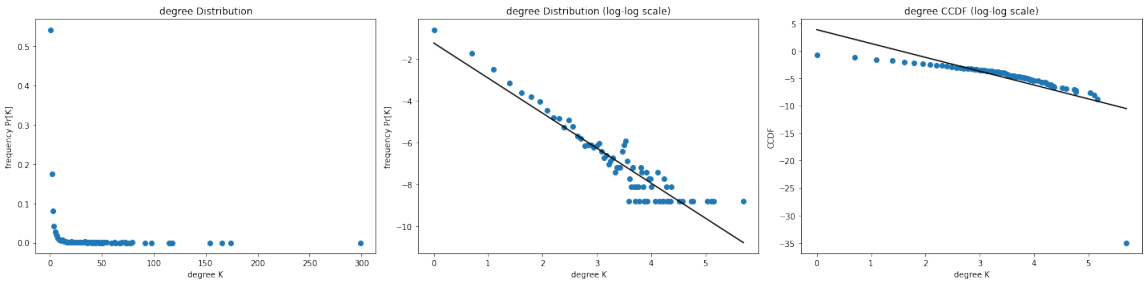


Figure 1: degree distribution

As we can see there are many nodes of degree 1, and in fact they make up about 54% of the entire network. This abundance of degree 1 nodes turned out to be quite problematic for link-prediction, as we will see later on.

Looking at the log-log scale we clearly see a linear behaviour, thus suggesting a power-law distribution. Recall that $p(k)$ is said to follow a power-law distribution if

$$p(k) = C \cdot k^{-\gamma} \tag{1}$$

with C normalizing constant and γ power exponent, main parameter of the distribution.
To estimate γ we use an Ordinary Least Square fit on the CCDF, which yields $\gamma \approx 2.52$, thus confirming a scale-free regime.
At this point we were also able to compute several other statistics ($k_{min} = 2$):

	k_{max}	C	$< k >$	$< k^2 >$	$\mathcal{K}$
theoretical	653	4.37	5.55	190.67	34.33
empirical	299	0.75	3.71	93.21	25.14

where $\mathcal{K}$ is the inhomogeneity ratio, which will play an important role in the robustness analysis.
As can be seen from the plots above, there seem to be one major hub in the network whose degree is significantly greater than the other hubs'degrees.
The node in question is *the_donald* (degree 299), subreddit dedicated to former POTUS Donald Trump. We will come back to this later when we discuss centrality measures.

3.2 Clustering Coefficients

Let's now take a look at Clustering Coefficients. Recall that for a given node v_i its clustering coefficient quantifies how close its neighbours are to being a clique (complete graph). It is therefore a measure of local density around the node. Mathematically:

$$C_i = C(v_i) = \frac{1}{k_i(k_i - 1)} \cdot \sum_{j,k} A_{ij} A_{jk} A_{ki}, \tag{2}$$

where $k_i = degree(v_i)$ and A adjacency matrix of the graph.
The Global Clustering Coefficient of the graph can then be defined as the average over all nodes:

$$C = \frac{1}{N} \cdot \sum_{i=1}^{N} C_i \tag{3}$$

which in our case turns out to be $C \approx 0.16$, meaning that in general the graph tends to be quite sparse even at a local level. Nonetheless there are still some nodes with high Clustering Coefficients, as can be seen in the plots below.
Here the first plot depicts the distribution of the Clustering Coefficients whereas the second one represents the correlation with the degrees.

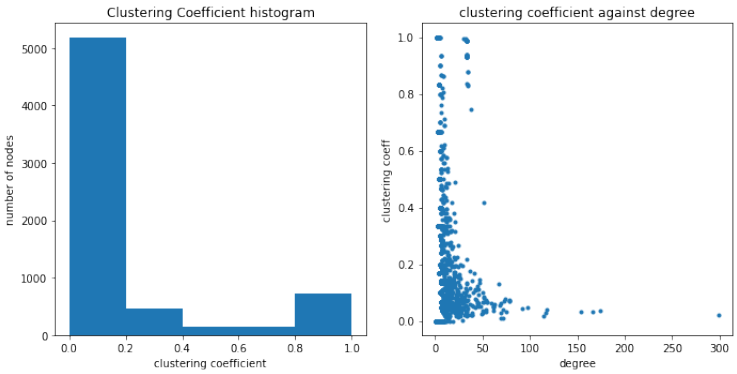


Figure 2: clustering coefficients

As can be seen, most of the neighborhoods tend to be quite sparse, although about 9.8% of the entire set of nodes have a clustering coefficient of 1. The subreddits associated with these nodes span a wide variety of topics, ranging from sports, games, trading, politics and others.
Similarly, we note that increasing degrees are associated with smaller clustering coefficients, although this is not at all surprising: the more neighbors a node has, the more unlikely it is that the neighborhood is a clique.
An interesting fact we would like to point out is that among the nodes with the highest clustering coefficient are many subreddits dedicated to various teams from different sports, such as soccer, football, basketball, and so on. The fact that this type of subreddit has a high clustering coefficient is quite predictable, since in many sports there are leagues and tournaments where teams have to compete against each other. Therefore, it is very likely that the subreddit dedicated to a specific team will also mention other opposing teams.

3.3 Assortativity

Finally, before moving on to centrality measures and community detection, let's also take a look at the assortativity of the network.
Assortativity refers to the tendency for nodes in a network to be connected to other nodes that are similar to them in some way, for example similar degrees. Assortativity can be either positive or negative. Positive assortativity refers to the phenomenon where nodes with similar characteristics tend to be connected to each other, while negative assortativity means that nodes with similar characteristics tend to avoid connecting to each other.
We can quantify the global degree assortativity of the network via the *assortativity coefficient* r, which in our case turns out to be $r \approx -0.03$. This tells us that there is no significant correlation between the degrees of the nodes and the connections between them. In other words, nodes with similar degrees are neither more nor less likely to be connected to each other than nodes with different degrees.
We can quantify assortativity also by considering the Average Degree of Neighbors function $K_{nn}(k_i)$, which computes for each node the average degree of its neighborhood. Namely:

$$K_{nn}(k_i) = \frac{1}{|N(i)|} \sum_{j \in N(i)} k_j, \tag{4}$$

with $N(i)$ neighborhood of v_i and k_j degree of v_j.
In our case we use a modified version on K_{nn} to also account for the weights in the network.

Specifically:

$$K_{nn}^{w}(k_i) = \frac{1}{s_i} \sum_{j \in N(i)} w_{ij} k_j, \tag{5}$$

with s_i weighted degree of node v_i, and w_{ij} weight of the edge that links v_i and v_j.
As we can see from the plots below, no particular pattern emerges from K_{nn}^{w}, whose behaviour remains quite stable even after considering 100 random rewiring of the edges in the graph. This confirms that the connections in the network seem to be quite random.

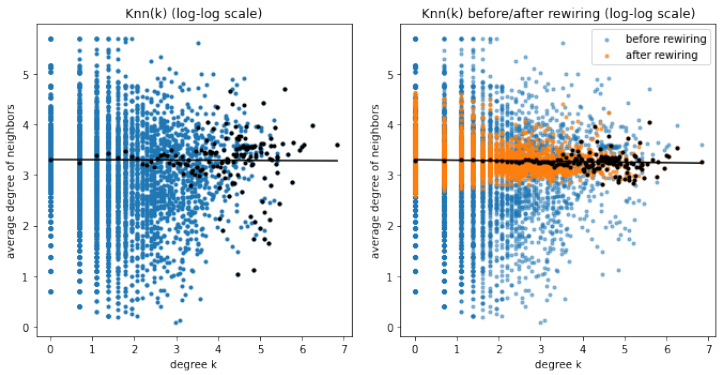


Figure 3: assortativity

4 Centrality Measures

In this section we focus on some measures of centrality. Centrality measures are metrics that quantify the importance or influence of a node in a network. They are used to identify the most influential nodes based on various criteria, such as number of connections, proximity to other nodes, or information flow through the network. The most important measures of centrality are: Degree, Betweenness, Closeness, HITS and Pagerank, each of which is based on different aspects of network structure.
These measures provide valuable insights that help us understand how information, influence or resources flow through the network.
Below we analyze the centralities based on Degree, HITS, and Pagerank. We will not discuss Betweenness and Closeness, as it was not possible to calculate them given the large size of the network.
To have a more orderly and easy-to-read discussion, we will start with degree centrality directly on the next page.

4.1 Degree Centrality

Degree centrality is just a normalized degree. Given a node v_i its centrality is defined as

$$C_d(v_i) = \frac{k_i}{N-1}. \tag{6}$$

As mentioned earlier, some of the nodes with the highest degrees are related to political topics and Trump in particular. However, the nodes with the highest degrees seem to span very different topics, including: games, sports, technical assistance, and so on. Below are two images of the network obtained using gephi. In the one on the left we have plotted only the nodes, while in the one on the right we have replaced the nodes with the actual names of the subreddits they refer to. In both images the size of each node is directly proportional to its degree centrality.

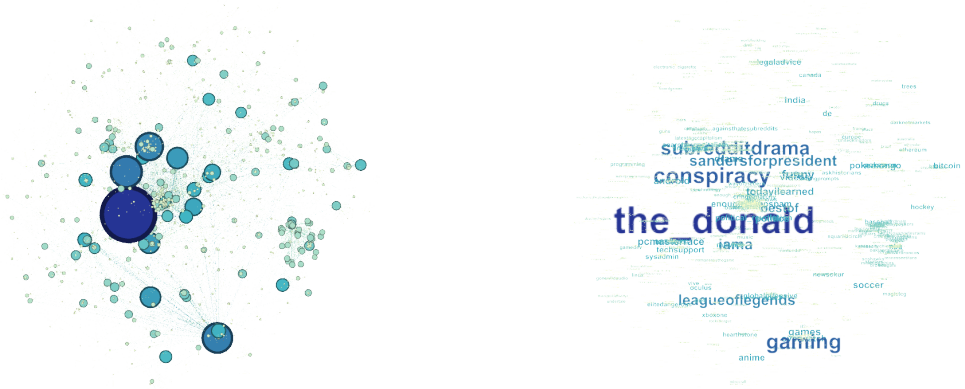
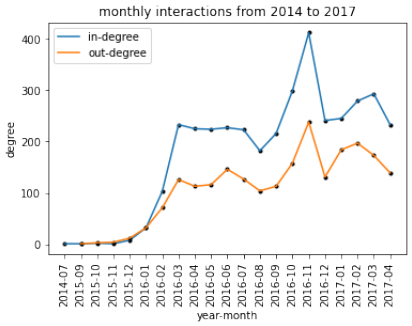


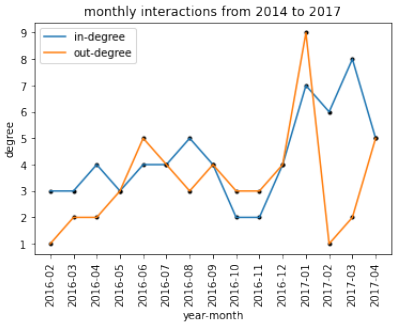
Figure 4: sizes and colors proportional to degree

As pointed out earlier, the highest-degree node turns out to be *the_donald*, subreddit dedicated to Donald Trump, former POTUS.
Using data from 2014 to 2017, we can see that after the 2016 presidential election, *the_donald* popularity increased significantly, as one would expect.
In addition, the number of subreddits whose names contain "trump" or "donald" increased from 26 in 2014-2015 to 188 in 2016-2017.
The other highest degree nodes are: *conspiracy* (174), *gaming* (166), *subredditdrama* (154), *SandersForPresident* (118), *iama* (117) *LeagueOfLegends* (115).
Interestingly one of the highest degree nodes happen to be the subreddit *conspiracy*, revolving around the discussion of conspiracy theories.
During his presidency, former U.S. President Donald J. Trump had a large base of supporters who were known for their strong loyalty to him and their conservative political views. Some of these supporters subscribed to various conspiracy theories, one of the most notable being the "QAnon" theory.
To detect any possible correlation among these two dominant subreddits we analyzed the number of monthly hyperlinks from 2014 to 2017.
Below we report two plots:

- the first one describing the monthly in-degree and out-degree for *the_donald*, from 2014 to 2017;
- the second one describing the monthly number of hyperlinks from *conspiracy* to *the_donald* (in-degree) and viceversa (out-degree).



(a) Monthly degree growth of *the_donald*



(b) *the_donald* and *conspiracy* interactions

Interestingly, the monthly interactions start from 2016, during the presidential election and they begin to spike in January 2017, month that marked the inauguration of Donald Trump as the 45th president of the United States.
This simple analysis seems to reflect the climate of unrest that surrounded Trump's election and presidency.

4.2 Pagerank centrality

The basic idea behind Pagerank is that a page that is linked to by many other pages is likely to be important, and thus should be ranked higher. Pagerank tries to solve the following eigenvector problem:

$$x = \mu S x + (1-\mu)v = \mu S x + (1-\mu)ve^T x = \left(\mu S + (1-\mu)ve^T\right)x = Gx, \tag{7}$$

where $S = (D^{-1}A)^T$, A adjacency matrix, D diagonal degree matrix, μ *damping factor*, v *teleportation vector*, e vector with all entries equal to 1, and $||x||_1$, $||v||_1 = 1$.
Pagerank solution is typically calculated using an iterative algorithm, starting from an initial set of scores and updating such scores until a stable solution is achieved.
The Pagerank algorithm describes a Markov Chain evolving on the nodes of the graph. At each iteration the damping factor μ controls the probability of flowing through adjacent nodes or jumping to a random node as specified by the teleportation vector v.
Pagerank solution is the *stationary distribution* of said Markov Chain.
As we already know, pagerank centrality tends to be quite similar to degree centrality, and this is also what emerges from the graphs below. Both degree centrality and pagerank tend to identify more or less the same set of nodes as most relevant.
In the images below, the size of each node is directly proportional to its pagerank centrality.

Figure 6: sizes and colors proportional to Pagerank scores

4.3 HITS centrality

HITS (Hyperlink-Induced Topic Search) centrality is a measure of the importance of a node in a network, based on the idea of authority and hub nodes. The HITS algorithm assigns two scores to each node in the network, an "authority" score and a "hub" score.
A node with a high authority score is considered to be a source of valuable information and is likely to be cited by other nodes in the network. On the other hand, a node with a high hub score is considered to be a node that is connected to many important authorities.
The HITS centrality scores are computed iteratively in an alternating fashion, using the adjacency matrix A of the network. The update equations are:

$$\begin{aligned} h^{k+1} &= Aa^k \\ &= \frac{AA^T h^k}{||A^T h^k||} \end{aligned} \qquad \begin{aligned} a^{k+1} &= \frac{A^T h^{k+1}}{||A^T h^{k+1}||} \\ &= \frac{A^T A a^k}{||A^T A a^k||} \end{aligned} \tag{8}$$

where $k \in \mathbf{N}$, h and a the vectors of *hub* and *authority* scores respectively.
It turns out that in our case hub and authority scores are virtually equal, which is why we consider only authority rankings in the graphs below.
Unexpectedly, HITS behaves very differently from degree and pagerank, assigning very low scores to almost all nodes except for a small group revolving around National Football League teams. Moreover, the lowest ranked node in this group has an authority score 10 times higher than the highest authority score outside the group.
We suspect that this may be related to the fact that this group of nodes has a very high average clustering coefficient and average degree.
In the following images, the size of each node is directly proportional to its authority score.

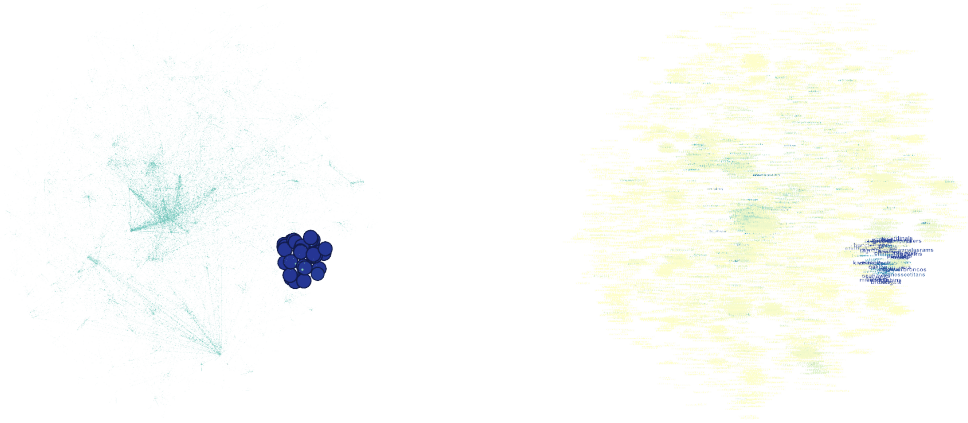


Figure 7: sizes and colors proportional to HITS Authority scores

5 Community Detection

Community detection is the process of identifying groups of nodes in a network that have a high degree of connectivity within the group and relatively low connectivity between groups. The goal of community detection is to uncover the underlying organizational structure of a network, which can reveal important patterns, relationships, and functionalities within the network.
There are various strategies for community detection, in the following we will focus on two algorithms based on Modularity Optimization: Louvain and Authority Shifting.
Recall that modularity is defined as:

$$\mathcal{Q} = \frac{1}{2m} \sum_{i,j} \left[A_{ij} - \frac{k_i k_j}{2m} \right] \delta(c_i, c_j) \tag{9}$$

where m is the number of edges in the network, $\delta(c_i, c_j)$ is the Kronecker delta function, which takes the value 1 if nodes v_i and v_j are in the same community ($c_i = c_j$) and 0 otherwise. Modularity quantifies how much the observed densities of groups of nodes deviate from the expected density after random rewiring of the graph edges. In this sense, high modularity indicates that the densities of the groups are not random, thus identifying possible community structures.
Apart from modularity optimization, we also experimented with Spectral Clustering, where we used the Random Walk Laplacian L_{RW} as a dimensionality reduction tool and subsequent clustering with the *X-means* algorithm (recall that $L_{RW} = D^{-1}L = D^{-1}(D - A)$). However, we decided not to report the final results because they were rather poor: the algorithm kept identifying single isolated nodes as communities, despite the number of eigenvectors chosen.
In the next page we start the discussion about Louvain.

5.1 Louvain Modularity Maximization

The Louvain Modularity Maximization algorithm is a heuristic algorithm that iteratively improves the division of the network into communities by moving nodes so as to increase modularity with each iteration. In details, the algorithm starts by considering each node as a separate community and iteratively merges the communities based on the increase in modularity that would result from the merge. The algorithm continues to iterate until no further improvement in modularity can be achieved.
In our case, Louvain's algorithm is able to distinguish 57 communities by obtaining a modularity of $\mathcal{Q} \approx 0.84$. We checked some of the communities and these seem to be quite consistent, in the sense that the nodes within a community are highly correlated with each other. There are still some misplacements, but overall the communities obtained are quite satisfactory.
The largest community consists of 856 nodes and, not surprisingly, is the one related to *the_donald*.
Note, however, that this community, while containing many political subreddits (both pro- and anti-Trump), also seems to contain many of the most popular subreddits on Reddit in general, not necessarily related to politics.
Some of the topics (communities) identified include: politics, vaping, hockey, soccer, baseball, football, programming languages, smartphones, tech support, cryptocurrency, anime, drugs, gaming and so on.
Note that although Louvain identifies a "gaming" community, this does not include all game-related subreddits, e.g., Nintendo has its own community, dominated by Pokemon-related subreddits, just like League of Legends, Hearthstone, Overwatch, and many other popular games with a strong fan base and community support.
This also happens with other communities, such as the "sports" community. There is a general sports community, but specific sports tend to have their own separate communities composed of subreddits related to various teams and so on.

Figure 8: communities found by Louvain Modularity Optimization

5.2 Authority Shifting via Personalized-Pagerank

Authority shifting via Personalized-Pagerank can be seen as a hybrid of Pagerank and modularity maximization, in that it uses Pagerank scores to guide community search while optimizing network modularity. In detail, for each node v we use Personalized-Pagerank scores to identify the most influential node u and then merge the communities associated with these two nodes only if the merge results in increased modularity.
In our case Authority Shifting identifies 422 communities, resulting in a modularity of $\mathcal{Q} \approx 0.79$. The largest community consists of 840 nodes and again contains *the_donald*, with an overlapping of 646 nodes w.r.t. the largest community identified by Louvain.
Interestingly, a difference of about 5% in modularity resulted in over 350 additional communities.
As before, we checked some communities, and these seem to be fairly consistent. The big difference, which also explains their large number, is that communities now tend to be extremely topic-centric, and even highly related topics produce different communities.
For example, Pokemon and Nintendo are now divided into two different communities, the first containing only Pokemon-related content and the second containing all of Nintendo's non-Pokemon-related content.
Remember, however, that modularity is only a heuristic idea and there is no one method better than the other. The different algorithms allow us to observe the network at different levels of granularity, so they are both useful and powerful in their own ways.

Figure 9: communities found by Authority Shifting

6 Network Robustness

Robustness refers to the ability of a network to maintain its structural and functional integrity in the face of failure or attack. The study of robustness is critical for many real-world networks, as it helps us understand the vulnerability of complex systems and design strategies to improve system resilience.
Robustness can be measured in several ways, including average shortest path length, largest connected component size, and resilience to targeted attacks on specific nodes.
In the following we will focus on the latter two aspects, since the large network structure makes it computationally too challenging to compute distances.
In particular, we analyze the robustness of the network under conditions of

- random node removal: random selection and removal of a node;
- targeted attacks: at each iteration the highest degree node is removed.

In both cases we use the relative size of the Giant Component to assess the resilience of the network. The network breaks down when the fraction of nodes removed f_c is such that the Giant Component disappears.
It turns out that we can compute exactly this threshold using the *Molloy-Reed* criterion, which states that a Giant Component exists if $\mathcal{K} = \frac{<k^2>}{<k>} > 2$, where $\mathcal{K}$ is the *inhomogeneity ratio*.
The previous criterion is very powerful because it holds regardless of the underlying degree distribution. Assuming a power-law distribution for k we can then calculate the theoretical values of both f_c^{random} and $f_c^{targeted}$, which are given by, respectively:

$$f_c = 1 - \frac{1}{\mathcal{K} - 1} \qquad\qquad f_c^{\frac{2-\gamma}{1-\gamma}} = 2 + \frac{2-\gamma}{3-\gamma} k_{min} (f_c^{\frac{3-\gamma}{1-\gamma}} - 1) \tag{10}$$

In the following table we report both theoretical and empirically computed thresholds.

	theoretical	empirical
f_c^{random}	0.97	0.996
$f_c^{targeted}$	0.20	0.28

The first thing we notice is that the theoretical thresholds are quite similar to the empirical ones. The second thing is that in the case of targeted attacks, the network breaks down much more easily.
To get a better idea of what these thresholds actually mean, let us also look at the graph below, which shows the relative size of Giant Component against the fraction of nodes removed.

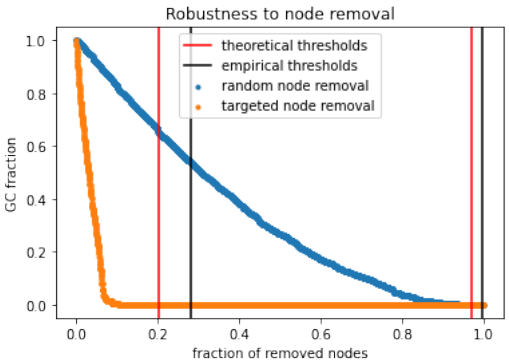


Figure 10: network robustness

As can be seen, the network is particularly robust to random node removal, which requires the removal of about 99% of the nodes to make the network collapse. This is not the case with targeted node removal, which instead collapses the network fairly quickly, requiring the removal of only 28% of nodes.
These results are not at all surprising; in fact, power-law distributions are known to suffer node-targeted attacks.

7 Link Prediction

Finally, as a last thing, let's also take a look at Link Prediction.
Link prediction is the task of predicting the formation of new links or relationships in a network. It is based on the idea that the formation of links is determined by certain patterns and regularities. The goal of link prediction is to identify these patterns and use them to predict the formation of new links.
There are several approaches to link prediction, including statistical methods, machine learning techniques, and network-based methods. In the following we will focus on the latter.
In our case, we implemented both Common Neighbours type approaches and Path-based techniques. Specifically, we tested: Jaccard Common Neighbours (JCN), Resource Allocation (RA), Adamic Adar (AA), Katz Local Path (LP), and Random Walk with Restart (RWR). In the following we recall very quickly how such techniques predict a link between two nodes v_i and v_j ($N(i)$ is the set of neighbours of v_i):

$$S_{JCN}(i,j) = \frac{|N(i) \cap N(j)|}{|N(i) \cup N(j)|} \qquad S_{RA}(i,j) = \sum_{k \in N(i) \cap N(j)} \frac{1}{|N(k)|} \tag{11}$$

$$S_{AA}(i,j) = \sum_{k \in N(i) \cap N(j)} \frac{1}{\log |N(k)|} \qquad S_{LP}(i,j) = (A^2 + \beta A^3)_{ij} \tag{12}$$

And finally recall also Random Walk with Restart:

$$S_{RWR}(i,j) = p_j^i + p_i^j \tag{13}$$

where p_j^i is the j-th component of the Personalized-Pagerank vector related to v_i.
However, calculating the above values is not enough: now we need to set a threshold above which the links will actually be activated. To determine the optimal value for the threshold we use Cross-Validation. Specifically, we proceed in the following way:

- split E into $E_{Train-Val}$ and E_{Test} (80% and 20% of edges respectively);
- split $E_{Train-Val}$ into E_{Train} and E_{Val} (80% and 20% of edges respectively).

We then use

- $G_{Train} = (V, E_{Train})$ to compute the pairwise similarity coefficients;
- $G_{Val} = (V, E_{Val})$ to perform a grid-search over some set of thresholds. The best activation thresholds is the one minimizing $FPR^2 + (1 - TPR)^2$ on G_{Val};
- $G_{Test} = (V, E_{Test})$ to test the model on never before seen data.

Unfortunately, all models get rather poor results except for AA and RA. The latter is the best model in terms of TPR and FPR, obtaining on the Test set scores of about $TPR \approx 71\%$, $FPR \approx 19\%$.
A common problem with all the algorithms is the extremely low *precision* rate, probably due to the sparsity of the graph. Indeed, as one would expect, prediction of ties is more difficult in sparse graphs, where most ties are in fact nonexistent.
In our case, the situation is even more problematic because degree 1 nodes make up 54% of the entire network. Degree 1 nodes are problematic in cross-validation because if the related edge is not in E_{Train}, it will be impossible to predict links for the node since it disappears from the training graph.
To take this into account, during partitioning we left all degree 1 nodes in G_{Train}. This obviously contributed to worse overall performances.

8 Conclusions

This project delved into the interactions between subreddits during 2016-2017, with an analysis of network structure and node importance also related to the 2016 U.S. presidential election.
The results of our analysis revealed that the degree distribution of the network follows a power law, with the subreddit *the_donald* having the highest degree, indicating its central role in connecting the different subreddits. The network is relatively sparse and random, as shown by the clustering and assortativity coefficients, which are relatively low. The Average Degree of Neighbours function, before and after random edge rewiring, confirmed the randomness of the links.
Measures of centrality were used to determine the relative importance of different nodes in the network. Degree centrality and Pagerank yielded similar results, while HITS produced unusual patterns, assigning low scores to all nodes except for a tightly clustered group revolving around the NFL.

Community detection algorithms such as Louvain and Authority Shifting were used to analyze network structure. These algorithms were able to identify consistent and coherent communities, allowing us to observe the behavior of the network at different levels of granularity.
We also tested the robustness of the network, finding significant vulnerability to node-targeted attacks, consistent with the underlying power-law distribution.
Finally, we attempted to predict the formation of new links using network-based link prediction techniques. The Resource Allocation method performed the best, but all methods had extremely low *precision*, probably due to the sparsity of the graph and the abundance of nodes with only one link.
Overall, our analysis indicates that the 2016 U.S. presidential election certainly had an impact on interactions among subreddits, with conspiracy theories playing a prominent role.
For potential future developments, it might also be interesting to explore overlapping community detection algorithms, statistical methods of link prediction, and perhaps even somehow incorporate the original attributes of each link, information that we discarded to achieve a simpler graph.
In conclusion, our analysis represents only a starting point, and there is still much room for further ideas and future developments.

References

[1] Albert-László Barabási et al. *Network science*, Cambridge university press, 2016.