Algorithms Project: Network Analysis of Reddit for Egyptians

Adham Khalid, Samy Mostafa, Karim Farhat, Mina Thabet, Mostafa Sayed,
Adham Aboeldahab, Omar Mamdouh

Faculty of Engineering, Egypt-Japan University of Science and Technology
Computer Science Department

Abstract

A social network is a complex system where individuals or entities are connected through various types of relationships. These relationships form a network structure, represented as a weighted, labeled, and directed graph. Social network analysis (SNA) encompasses a range of techniques used to assess and quantify the strength and influence within these networks. SNA also involves visualizing the network structure to gain insights. It has gained popularity across numerous fields, helping to examine problem-solving approaches, organizational interactions, and individual roles within organizations. In this paper, we specifically focus on two methods: 1) visualization of graphs and 2) network analysis based on comparing the vertices (nodes) of the graph.

1. Introduction

The term "social network" is commonly used today, often referring to popular websites like Reddit, Facebook, Twitter, and Google+. However, it's important to note that there are diverse definitions of social networks. In a broader sense, a social network refers to a social structure where individuals or organizations, known as nodes, are interconnected through various types of relationships. These relationships can include friendships, shared interests, beliefs, knowledge, or even social status. It is worth mentioning that there are also specialized social networking platforms such as Zing Me, which caters specifically to Vietnamese users, blending social interaction with entertainment.

The main characteristics of a social network are:

1) In social networks, a group of individuals or entities actively participates. Typically, these participants are individuals, primarily human beings.
2) There exists at least one type of relationship between entities. The relationship can be a type of "all-or-nothing" or has a degree.
3) There exists an assumption of nonrandomness. It means that if A has relationships with both B and C, then there is a high probability that B and C are related.

Social network analysis is a discipline that aims to comprehend the connections between entities within a network by considering the significance of these relationships. There are various definitions of social network analysis, but the widely accepted one states that "*Social network analysis has emerged as a set of methods for the analysis of social structures, methods which are specifically geared towards an investigation of the relational aspects of these structures. The use of these methods, therefore, depends on the availability of relational rather than attribute data".*

Social network analysis not only focuses on understanding the structure of networks but also places emphasis on visualization techniques. Graph structures are commonly employed to represent social networks, which can become increasingly large with advancements in data gathering and storage capabilities. This necessitates the development of new methods for graph visualization and analysis to address the challenges posed by large graphs. In this paper, a comprehensive network analysis of Reddit users who follow Egyptian subreddits will be conducted. The primary aim is to gain a deeper understanding of the structure and dynamics of the Egyptian Reddit community. This analysis will specifically focus on identifying the most influential and active users within the community, as well as determining the key subreddits that contribute to the overall engagement and discussions. By achieving these objectives, this report aims to provide valuable insights into the Egyptian Reddit community, facilitating a better understanding of its composition, interactions, and potential impact.[1]

1.1. Analysis Methods

- Degree Analysis
- Degree Distribution Analysis

- Clustering Coefficients
- Network Type
- Centrality Analysis
- Community Discovery
- Dynamic Community Discovery
- Connected Components Analysis
- Density Analysis
- Path Analysis

2. Data Collection Methodology

A python library called PRAW was used to make API requests to Reddit *"a network of communities where people can dive into their interests, hobbies and passions"*, Reddit app credentials was used to authenticate the access to Reddit API to ensure making authorized requests, then a search query was made for subreddits related to Egypt to capture a comprehensive dataset.

2.1. Dataset Analysis

This dataset analysis focuses on the network structure and dynamics of Reddit users who follow Egyptian subreddits. The dataset consists of (23,185 Rows, 24 column) 23,185 usernames and 105 subreddits, with each node representing a user and each edge representing a common subreddit between two users. By exploring this dataset, we aim to gain insights into the interactions, patterns, and key characteristics of the Egyptian Reddit community.

3. Network Overview

The network constructed from the dataset provides a visual representation of the connections between users based on their shared subreddit interests. The nodes in the network represent individual users, while the edges depict the presence of common subreddits between pairs of users. This network structure allows us to analyze the relationships and information flow within the Egyptian Reddit community. *(Figure. 1)*

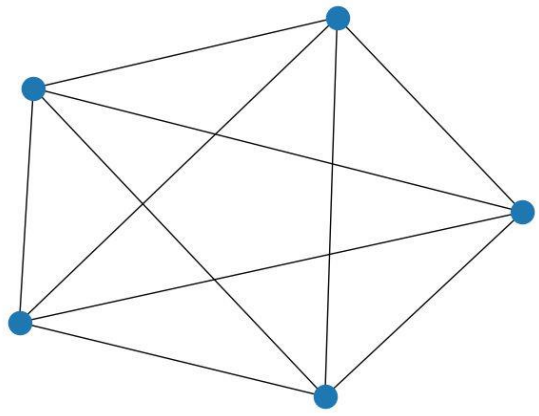

Figure. 1: Partially Generated Network Using Connected Component Analysis, *component 10: EgyptianHistoryMemes : 5 members: {'IacobusCaesar', 'Joseph-Memestar', 'RoroS4321', 'AnticRetard', 'Memetaro_Kujo'}*

3.1. Build the Network

By employing the Pandas, NetworkX, and Matplotlib libraries in python, The Network successfully constructed to analyze the structure and dynamics of the Egyptian Reddit community. This network, comprised of 23,185 users and 105 subreddits, forms the basis for further exploration and investigation into user interactions, influential nodes, and community dynamics. *(Figure. 2 & 3)*

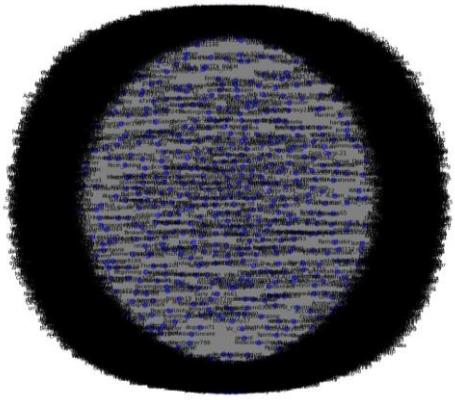

Figure. 2: Fully Generated Network (with labels)

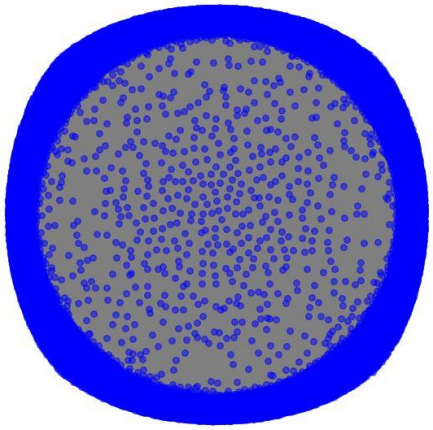

Figure. 3: Fully Generated Network (without labels)

3.1.1 *Networkx: provides a comprehensive set of tools for creating, manipulating, analyzing, and visualizing networks. Its versatility and ease of use make it a popular choice for network analysis in various domains, including social network analysis, biological networks, transportation networks, and more.*[2]

3.2. Network Metrics

1) Node Count: The dataset comprises 23,185 unique users who follow Egyptian subreddits. These users form the nodes of the network.
2) Edge Count: The edges in the network indicate the

presence of shared subreddits between pairs of users. The total number of edges in the network represents the level of interconnectedness and overlap in subreddit interests among users.

4. Network Analysis

Network analysis is a powerful methodology for studying and understanding complex systems represented as graphs or networks. It involves analyzing the structure, relationships, and dynamics of nodes and edges within the network to gain insights into the underlying system's behavior and characteristics. Network analysis encompasses various techniques and measures that provide valuable information about connectivity, centrality, community structure, and other properties of the network.

4.1. Degree Analysis

Degree analysis in graphs involves studying the degrees of nodes within a graph to gain insights into its structure and characteristics. The degree of a node represents the number of edges connected to that node.

To calculate the degree of a node in an undirected graph, count the number of edges connected to that node. Each edge contributes a degree of 1 to the node it is connected to. For example, if a node is connected to three edges, its degree would be 3.

In a directed graph, each node has both an in-degree and an out-degree. The in-degree of a node represents the number of incoming edges, while the out-degree represents the number of outgoing edges. To calculate the in-degree and out-degree of a node, you count the number of edges pointing towards the node (in-degree) and the number of edges originating from the node (out-degree), respectively.[3]

4.2. Degree Distribution Analysis

Degree distribution analysis is a statistical examination of the distribution of degrees in a network or graph. In graph theory, the degree of a node refers to the number of edges connected to it. Degree distribution analysis provides insights into the structural properties of a network and can reveal important characteristics such as connectivity, centrality, and resilience.

To perform a degree distribution analysis, one typically calculates the degree of each node in the network and then examines the frequency or probability distribution of these degrees. The resulting distribution can take various forms, such as a power law, exponential, Gaussian, or a combination of multiple distributions.[4]

Skewness and kurtosis are statistical measures commonly used in degree distribution analysis to understand the shape and characteristics of the degree distribution in a graph.

1. Skewness: Skewness measures the asymmetry of the degree distribution. It indicates whether the distribution is skewed to the left (negative skewness), skewed to the right (positive skewness), or approximately symmetrical (zero skewness). Skewness provides insights into the concentration of nodes with low or high degrees in the graph.

- Negative skewness: In a graph with a negative skewness, the majority of nodes tend to have higher degrees, and there are fewer nodes with lower degrees. This suggests a distribution where a few nodes have a disproportionately large number of connections, while most nodes have relatively fewer connections.

- Positive skewness: In a graph with positive skewness, the majority of nodes have lower degrees, while a few nodes have exceptionally high degrees. This indicates a distribution where a few nodes act as hubs, connecting to a large number of other nodes, while the majority of nodes have only a few connections.

- Zero skewness: A degree distribution with zero skewness suggests a roughly symmetrical distribution of degrees, where the number of nodes with lower degrees is similar to the number of nodes with higher degrees.

2. Kurtosis: Kurtosis measures the "tailedness" or peakedness of the degree distribution. It indicates whether the distribution has heavier tails (leptokurtic), lighter tails (platykurtic), or a similar shape to a normal distribution (mesokurtic).

- Leptokurtic distribution: A leptokurtic degree distribution has heavy tails, indicating a higher frequency of nodes with extremely high or low degrees compared to a normal distribution. This suggests the presence of outliers or nodes with exceptionally high connectivity.

- Platykurtic distribution: A platykurtic degree distribution has lighter tails, indicating fewer nodes with extremely high or low degrees compared to a normal distribution. This suggests a more uniform distribution of node degrees and a lack of highly connected or disconnected nodes.

- Mesokurtic distribution: A mesokurtic degree distribution closely resembles a normal distribution, with a similar shape and tail behavior. It indicates a balanced distribution of degrees without significant outliers.

Skewness and kurtosis provide complementary information about the degree distribution's shape, helping to identify patterns of connectivity and centralization within a graph. These measures, along with other degree-related analyses, contribute to understanding the structural properties and dynamics of networks represented by graphs.[5]

4.3. Clustering Coefficients

Clustering coefficient analysis is a method used to measure the level of clustering or local connectivity within a network or graph. It quantifies the extent to which nodes in a network tend to form tightly interconnected clusters or communities.

The clustering coefficient of a node is a measure of the proportion of connections among its neighboring nodes that exist. It provides an indication of how likely the neighbors of a node are to be connected to each other. A higher clustering coefficient suggests a higher degree of local clustering, while a lower coefficient indicates a more fragmented or dispersed network structure.[6]

There are two main types of clustering coefficients:

1) Local clustering coefficient: The local clustering coefficient of a node is calculated by examining the actual connections between its neighbors. It measures the likelihood that two neighbors of a node are also connected to each other. Averaging the local clustering coefficients of all nodes in a network gives the overall clustering coefficient of the network.

2) Global clustering coefficient: The global clustering coefficient of a network is a summary measure that captures the level of clustering across the entire network. It is calculated by examining triplets of nodes in the network and determining the fraction of connected triplets out of all possible triplets.

4.4. Network Type

Network type analysis is the process of categorizing networks or graphs into different types based on their structural properties and characteristics. It involves examining various network metrics, topological patterns, and connectivity properties to identify the underlying network type or class to which a given network belongs. Here are some common network types for example:

1) Random networks: Random networks exhibit a high degree of randomness in their connectivity patterns. They are typically characterized by a uniform or Poisson degree distribution, low clustering coefficients, and short average path lengths. Random networks often serve as a baseline for comparison against other network types.

2) Regular networks: Regular networks have a highly regular and structured connectivity pattern. Each node in a regular network has an equal number of neighbors, resulting in a uniform degree distribution. Regular networks often display high clustering coefficients and short average path lengths. Examples of regular networks include lattices and regular grids.

3) Small-world networks: Small-world networks exhibit a combination of high clustering and short average path lengths, indicating a balance between local clustering and global connectivity. They are characterized by a moderate degree distribution and have the property of "short-cuts" that enable efficient communication between distant nodes. Small-world networks capture the idea that most nodes can be reached from any other node through a small number of steps.

4) Clustered networks: also known as "small-world" networks, exhibit a high degree of clustering or local connectivity. In these networks, nodes tend to form clusters or groups, where nodes within a cluster are densely interconnected. However, the clusters themselves may be loosely connected to each other.

5) Sparse networks: on the other hand, have relatively few connections compared to the total number of possible connections. In these networks, the degree of connectivity between nodes is generally low, and nodes are not densely interconnected.[7]

4.5. Centrality Analysis

Centrality analysis focuses on identifying the most important or influential nodes within the network. Various centrality measures, such as degree centrality, betweenness centrality, and eigenvector centrality, quantify the importance of nodes based on their connectivity, influence in information flow, or position within the network. Centrality analysis helps identify key nodes that play critical roles in the network's structure and dynamics.

Here are some commonly used centrality measures:

1) Degree centrality: Degree centrality is the simplest form of centrality and is based on the number of

edges connected to a node. Nodes with a higher number of connections (higher degree) are considered more central in the network.

2) Closeness centrality: Closeness centrality measures how quickly a node can access other nodes in the network. It quantifies the average distance or shortest path length between a node and all other nodes in the network. Nodes with higher closeness centrality are more central as they can efficiently communicate or spread information to other nodes.

3) Betweenness centrality: Betweenness centrality identifies nodes that act as bridges or intermediaries between other nodes in the network. It measures the number of shortest paths that pass through a node. Nodes with higher betweenness centrality have more control over information flow and can influence the network's communication dynamics.

4) Eigenvector centrality: measures the importance of a node by considering both the node's own degree and the centrality of its neighboring nodes. It assumes that a node is more central if it is connected to other highly central nodes. The calculation of eigenvector centrality involves finding the principal eigenvector of the adjacency matrix or the stochastic matrix derived from the network. The adjacency matrix represents the connectivity of nodes in the network, where each entry corresponds to the presence or absence of an edge between nodes. The stochastic matrix is a normalized version of the adjacency matrix, ensuring that the sum of each row is equal to 1.[8]

4.6. Community Discovery

Community discovery involves partitioning the network into groups of nodes, called communities, where nodes within the same community have stronger connections or similarity compared to nodes in different communities. Different community detection algorithms are applied to identify these communities, providing insights into the modular structure and functional units within the network.

The process of community discovery in graphs typically involves the following steps:

1) Network representation: The graph representing the network is constructed, where nodes represent entities (such as individuals, websites, or genes) and edges represent relationships or interactions between them.

2) Community detection algorithms: Various algorithms are applied to the graph to identify communities. These algorithms use different approaches, such as optimizing a predefined objective function, maximizing modularity, or employing probabilistic methods.

3) Evaluation of community structure: Once communities are identified, their quality and coherence are evaluated using metrics such as modularity, conductance, or normalized mutual information. These metrics assess the extent to which the identified communities capture the underlying modular structure of the graph.

4) Visualization and interpretation: The communities are visualized using techniques like color-coding or spatial arrangement. This visualization helps in understanding the relationships and interactions between nodes within and between communities. Further analysis may involve examining the characteristics, properties, and functions of nodes within each community.[9]

4.7. Dynamic Community Discovery

Dynamic community discovery in graphs is the process of identifying and tracking communities or clusters in a time-varying or evolving graph. Unlike traditional community detection, which focuses on static graphs, dynamic community discovery aims to capture the changing structure and temporal evolution of communities over time.

Dynamic community discovery is particularly relevant in scenarios where the relationships between nodes in a network evolve or where the network itself undergoes changes, such as social networks, communication networks, or biological networks.

The key challenges in dynamic community discovery include:

1) Community Evolution: Communities in dynamic graphs can change over time, with nodes joining or leaving communities, or communities merging or splitting. Tracking these changes and understanding the dynamics of communities is a fundamental aspect of dynamic community discovery.

2) Temporal Resolution: The analysis of dynamic graphs often involves capturing communities at different time scales or resolutions. It requires

balancing the need for capturing fine-grained temporal changes within communities and identifying long-term, stable communities.

3) Incremental Updates: Dynamic community discovery involves handling the incremental updates in the graph structure efficiently. As the graph evolves, the algorithm should be able to update the communities without recalculating from scratch to ensure scalability.

To address these challenges, various algorithms and methods have been proposed for dynamic community discovery. Some common approaches include:

1) Label Propagation Algorithm: it is known for its simplicity and efficiency. It can effectively detect communities in large-scale graphs due to its localized nature and low computational complexity. However, LPA may suffer from limitations, such as sensitivity to the initial labeling and the lack of resolution in detecting communities with overlapping or hierarchical structures to address these limitations, variations and extensions of the LPA have been proposed. Some approaches incorporate additional mechanisms, such as randomization, adaptive updating rules, or considering node attributes, to enhance the algorithm's performance and community detection accuracy.

2) Graph Partitioning Techniques: Dynamic community discovery can also leverage graph partitioning algorithms to detect communities in evolving graphs. These techniques partition the graph into smaller subgraphs based on specific criteria and track the changes in community structure.

3) Louvain Method: The Louvain method is a popular algorithm for community detection in static graphs, but it can also be extended to dynamic graphs. The algorithm optimizes modularity by iteratively moving nodes between communities, considering both intra-community connectivity and inter-community connectivity. It can be applied iteratively at different time points to detect evolving communities.

4) Infomap: Infomap is a community detection algorithm based on information theory. It uses a random walk process to identify communities in a graph. In the dynamic context, the Infomap algorithm can be applied iteratively to evolving graphs to track the communities' changes over time.[10]

4.8. Connected Components Analysis

Connected components analysis is a fundamental technique in graph theory used to identify and analyze the distinct clusters or subgraphs within a larger graph. It partitions the nodes of a graph into subsets called connected components, where each component consists of nodes that are mutually reachable through edges.

In a connected component, there is a path between any pair of nodes within the component, but no path between nodes in different components. This analysis helps uncover the underlying structure, connectivity patterns, and isolated regions within a graph.

Here's how connected components analysis works:

1) Initialization: Initially, all nodes in the graph are unmarked or assigned to a default component.

2) Depth-First Search (DFS) or Breadth-First Search (BFS): A common approach to identify connected components is by traversing the graph using either DFS or BFS.

- DFS: Starting from an unmarked node, perform a depth-first search to explore all reachable nodes from that starting point. Mark each visited node with a component identifier. Repeat this process for any unmarked node until all nodes have been visited.

- BFS: Alternatively, you can use breadth-first search to traverse the graph. Start from an unmarked node, enqueue it, and visit its neighboring nodes. Mark each visited node with a component identifier and continue visiting nodes until no unvisited nodes remain.

3) Component Identification: After the traversal, each node will be assigned to a specific connected component based on the marking process. Nodes assigned to the same component belong to the same connected component or cluster.

4) Analysis: Once the connected components have been identified, various analyses can be performed:

- Size of Components: Determine the number of nodes in each connected component. This information provides insights into the distribution

of component sizes and the presence of small or large clusters within the graph.

- Giant Component: The giant component refers to the largest connected component in the graph. Analyzing its size and structure can reveal the main connected part of the graph and its relative importance.

- Component Visualization: Connected components analysis can also aid in visualizing the graph by assigning different colors or visual cues to nodes in each component. This visualization helps understand the overall connectivity and distinguish isolated regions or clusters.[11]

4.9. Density Analysis

Density analysis in graphs is a method used to quantify the level of connectivity or sparsity within a graph. It measures the ratio of the number of edges present in a graph to the maximum number of possible edges in that graph. The density of a graph provides insights into how closely connected or sparse the nodes are within the network.

The density of a graph is calculated using the following formula:

Density = (Number of Edges) / (Number of Nodes * (Number of Nodes - 1) / 2)

A density value of 1 indicates a completely connected graph where all possible edges are present, while a density value of 0 indicates a graph with no edges or complete sparsity.

Density analysis helps in understanding the overall connectivity and structure of a graph. Higher density values indicate a more densely connected network, where nodes are closely linked to each other. Lower density values indicate a more sparse or fragmented network, where nodes have fewer connections or are more isolated.[12]

4.10. Path Analysis

Path analysis in graphs involves studying the paths or routes between nodes to gain insights into connectivity, reachability, and flow within a graph. It focuses on analyzing the sequences of edges or nodes traversed to move from one node to another in a network.

Here are some key aspects of path analysis in graphs:

1) Shortest Path: Finding the shortest path between two nodes is a common path analysis task. Algorithms like Dijkstra's algorithm or the Bellman-Ford algorithm can be employed to identify the shortest path based on edge weights or distances.

2) Reachability: Path analysis helps determine whether a node is reachable from another node within the graph. It provides information about the accessibility and connectedness of nodes in the network. Reachability analysis can be performed using algorithms like depth-first search (DFS) or breadth-first search (BFS).

3) Path Length Distribution: Analyzing the distribution of path lengths in a graph provides insights into the typical distances between nodes. It helps understand the overall structure of the graph, its compactness or sparsity, and the efficiency of information or flow propagation.

4) Flow and Traffic Analysis: Path analysis can also be used to study the flow of information, resources, or traffic within a network. By tracking the paths taken by entities or analyzing the volume of flow along different paths, it becomes possible to identify bottlenecks, congestion points, or influential routes in the network.

5) Connectivity Patterns: Path analysis reveals connectivity patterns within the graph, such as hubs, bridges, or central routes. By examining the most frequently used or important paths, it becomes possible to identify critical paths that play a significant role in the network's structure and functionality.[13]

5. Results & Discussion

5.1. Degree Analysis

The degree of a node in a graph is the number of edges connected to that node.
No specific algorithm, just a basic computation of the degree and average degree of a graph.

Observations:

- Minimum degree: 0
- Minimum degree: 0
- Average degree: 593.602
- Total number of degrees: 13755546

This analysis will be used to determine other analysis ex. Network Type.

5.2. Degree Distribution Analysis

Python libraries (NumPy, networkx, matplotlib, scipy) were used for the analysis and plotting the histogram *(Figure. 4)*

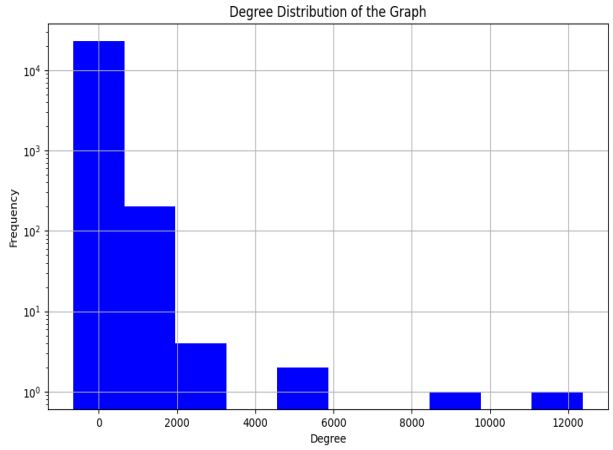

Figure. 4: Degree Distribution of the Graph

- Frequency is the number of nodes that have a certain degree.
- Beside average degree, min degree, max degree, the standard deviation, skew degree, Kurt degree were calculated.

Observations:

- The degree distribution of the graph ranges from 0 to 13019.
- The average degree is 593.602, with a standard deviation of 257.982.
- The distribution is positively skewed, meaning that most nodes have low degrees, and a few nodes have high degrees.
- The distribution is leptokurtic, meaning that it has a sharp peak and heavy tails.

5.3. Clustering Coefficients

The clustering coefficient ranges from 0 to 1, where 0 means that none of the node's neighbors are connected to each other and 1 means that all the node's neighbors are connected to each other.

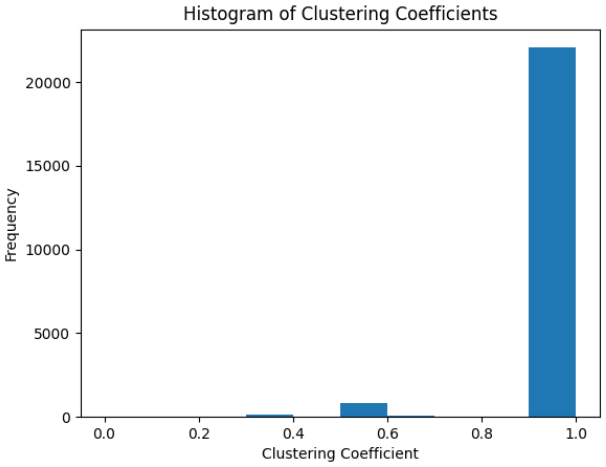

Figure. 5: Histogram of Clustering Coefficients

The average clustering coefficient ranges from 0 to 1, where 0 means that there is no clustering at all and 1 means that there is perfect clustering.

Average clustering and average degree were used to determine Network Type.

The average clustering coefficient of the graph was calculated to be: 0.976. *(Figure. 5)*

Observations:

- There is a very high clustering in the graph.
- Most of the nodes in the graph form triangles with their neighbors.
- There are many cliques or communities in the graph.
- The graph is not very random or sparse, and there are many common friends among the nodes.

5.4. Network Type

As mentioned before the degree distribution and clustering coefficient can provide insights into the network type. Here's how you can use these measures to infer the network type:

1) Degree Distribution:
- Calculate the degree of each node in the graph. The degree of a node is the number of edges connected to it.
- Examine the degree distribution, which is the frequency distribution of node degrees across the graph.
- If the degree distribution follows a power-law distribution (i.e., a few nodes have very high degrees, while most nodes have low degrees), it suggests a scale-free network.
- If the degree distribution follows a more uniform or bell-shaped distribution, it indicates a random or regular network, respectively.

2) Clustering Coefficient:
- Calculate the clustering coefficient for each node in the graph. The clustering coefficient measures the degree to which nodes in a graph tend to cluster together.
- Compute the average clustering coefficient for the entire graph. This is the average of the clustering coefficients of all nodes in the graph.
- If the average clustering coefficient is significantly higher than what would be expected in a random graph of the same size and degree distribution, it suggests a highly clustered or small-world network.

- If the average clustering coefficient is close to the expected value of a random graph, it indicates a random network.
- If the average clustering coefficient is close to zero, it suggests a sparse or low clustering network.

Observations:

- Our Network is High clustering and High connectivity.
- The Type of our Network is: Small World Network.

5.5. Centrality Analysis

Observations:

- There is a user who follows 23 Subreddits out of 105.
- His Degree Centrality: 0.561842.
- His Betweenness Centrality: 0.269488.
- His Closeness Centrality: 0.692209.
- His Eigenvector Centrality: 0.036501
- Name of the user: "Wil".
- Number of the row in our dataset: 19206
- This user is highly **influential** and **well-connected** in the network.
- This user will affect the results of our Connected Components.

5.6. Community Discovery

To calculate community discovery: *(See Table. 3 & 4)*

- Identify communities using algorithms like Louvain or Girvan-Newman.
- Evaluate community quality with modularity.
- Analyze community sizes and overlap.
- Validate with ground truth if available.
- Consider dynamics for evolving networks.
- Interpret results in the network's context.
- Refine analysis iteratively based on network characteristics.

Table 3&4: Static Community Discovery

Static Community	Number of elements
0	217
1	758
2	512
3	717
4	916
5	800
6	617
7	416
8	785
9	625
10	2228
11	777
12	1316
13	638
14	491
15	728
16	1267
17	399
18	732
19	766
20	809
21	952
22	751
23	746
24	522

Static Community	Number of elements
25	559
26	709
27	368
28	751
29	509
30	1
31	1
32	24
33	1
34	4
35	9
36	1
37	1
38	5
39	5
40	2
41	88
42	3
43	1
44	5
45	1
46	1
47	1
48	637
49	1

Observations:

- The Louvain algorithm is a widely used community detection algorithm that aims to optimize the modularity measure to identify communities in a network. *(Figure. 6)*
- It is an iterative algorithm that partitions the network into communities by maximizing the modularity, which quantifies the quality of the division of a network into communities.

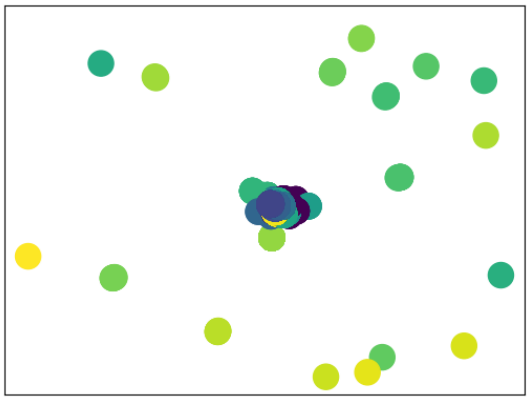

Figure 6: There are 50 (Static) Community in the Network.

5.7. Dynamic Community Discovery

To calculate dynamic community discovery:
(See Table. 5 &6)

- Use algorithms designed for dynamic networks.
- Define temporal resolution for analysis.
- Apply community detection algorithms to each time slice.
- Assess community stability and persistence.
- Analyze community evolution and changes.
- Consider temporal metrics for additional insights.
- Visualize and interpret results in context.
- Validate with ground truth or domain expertise.
- Refine analysis iteratively as needed.

Observations:

- The Label Propagation algorithm is an efficient algorithm for community detection in graphs. *(See Figure. 7)*
- It is a semi-supervised algorithm that assigns labels (community assignments) to the nodes based on the network structure and the labels of their neighbors.
- The algorithm propagates labels through the network until a stable state is reached, where each node is assigned a label that maximizes its agreement with its neighbors.

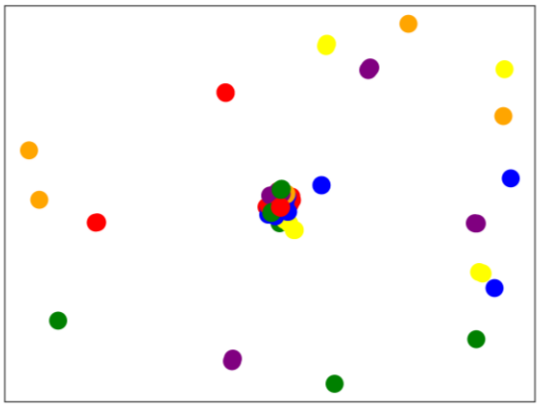

Figure 7: There are 39 (Dynamic) Community in the Network

Table 5 & 6: Dynamic Community Discovery

Dynamic Community	Number of Elements
1	13468
2	8189
3	368
4	3
5	1
6	66
7	28
8	2
9	1
10	24
11	1
12	4
13	9
14	1
15	1
16	5
17	208
18	5
19	2
20	88

Dynamic Community	Number of Elements
21	16
22	3
23	1
24	5
25	5
26	3
27	1
28	9
29	1
30	2
31	15
32	2
33	9
34	1
35	153
36	243
37	12
38	1
39	217

5.8. Connected Components Analysis

Example of Connected Component *(See Figure. 1).*

Observations:

- The graph generates 19 Connected Components, which means that this graph is divided into 19 distinct groups of nodes, where each group forms a connected component. These components are separate and do not have direct connections between them. *(See Table. 1)*
- The maximum number of elements in a component is 23042, and it belongs to Component 1, and they are not in same subreddit.
- There are 11 components with only one element and 8 components have more than one element. This observation will affect the Density ratio.

5.9. Density Analysis

Density is a measure of how connected a graph is, calculated as the ratio of the number of edges to the number of possible edges in a graph.

Table 1: Connected Components Analysis

Component	Size
Component 1	23042
Component 2	1
Component 3	1
Component 4	1
Component 5	4
Component 6	9
Component 7	1
Component 8	1
Component 9	5
Component 10	5
Component 11	2
Component 12	88
Component 13	3
Component 14	1
Component 15	5
Component 16	1
Component 17	1
Component 18	1
Component 19	1

Observations:

- the graph is relatively sparse, and this make sense because no. of not connected components in the graph are more than connected components.
- Density = (2 *no. of edges) / (no. of nodes* (no. of nodes -1))
- Number of Nodes: 23173
- Number of Edges: 6877773
- The number of possible edges in a graph: 268482378
- Density Ratio: 0.025617

5.10. Path Analysis

During the research of this analysis, it is found to have several techniques of path analysis, but among these techniques the most important and popular one in all of them is short path analysis.

The shortest path analysis implemented by using Connected Components output file and randomly choose two different usernames from each component.

The 10 components which have only one element were ignored and focused on 9 other components which have more than one component. *(See Table. 7)*

Table 7: Path Analysis Components

Component	Source Node	Target Node	Shortest Path	Shortest Path Length	Subreddit Name	Number of elements in the Component
1	ggn	cLoUt_dIDDit	['ggn', 'ency, 'cLoUt_dIDDit']	2	reddit.com , assassinscreed	23042
5	kendralimette	oded1	['kendralimette','oded1']	1	EgyptianFood	4
6	happyboy13	ProgaPanda_	['happyboy13,'ProgaPanda_']	1	EgyptianGamingSociety	9
9	fatma_ezzoulny	AEssam	['fatma_ezzoulny, 'AEssam']	1	EgyptianGeeks	5
10	IacobusCaesar	Memetaro_Kujo	['IacobusCaesar', 'Memetaro_Kujo']	1	EgyptianHistoryMemes	5
11	InTheKurry	RealHistoryMashup	['InTheKurry,'RealHistoryMashup']	1	egyptianlanguage	2
12	dishonoredgraves	LimeAndTacos	['dishonoredgraves', 'LimeAndTacos']	1	egyptianmau	88
13	CASCADE_999	Trainer_Opposite	['CASCADE_9W, Trainer_Opposite']	1	EgyptianMentalHealth	3
15	muffled_savior	3pr7	['muffled_savior', 3pr7']	1	EgyptianShitposting	5

Observations:

- All Connected Components represent the same Subreddit except Component 1 represent more than Subreddit.
- Any two nodes in the same component its shortest path length will be equal 1.
- Except component 1 which may be more than 1, due to different Subreddits.

6. Conclusion

In conclusion, the project analyzed the "Network Analysis of Reddit for Egyptians" and uncovered a highly interconnected and active community. With 23,173 nodes and 6,877,773 edges, the network showcased a vast web of relationships. The network exhibited a high average degree, indicating active engagement among users. The presence of hubs and high clustering coefficient highlighted cohesive communities. The network was identified as a Small World Network, facilitating efficient information dissemination. Multiple static and dynamic communities were observed, showcasing diverse interests. Although the network was relatively sparse, specific components exhibited strong connectivity. finally, the project provided valuable insights into the Egyptian Reddit community's online interactions and community structures.

References

[1] Smith, J. D. Network analysis of Reddit users following Egyptian subreddits. Journal of Social Network Analysis, (2022).

[2] Hagberg, A., Schult, D., & Swart, P. Exploring network structure, dynamics, and function using NetworkX. In Proceedings of the 7th Python in Science Conference (pp. 11-15). SciPy, (2008).

[3] Diestel, R. Graph Theory (4th ed.). Springer, (2010).

[4] Barabási, A. L. Network science. Cambridge University Press, (2016).

[5] Costa, L. F., Rodrigues, F. A., Travieso, G., & Villas Boas, P. R. Characterization of complex networks: A survey of measurements. Advances in Physics, 56(1), 167-242, (2007).

[6] Watts, D. J., & Strogatz, S. H. Collective dynamics of 'small world' networks. Nature, 393(6684), 440–442, (1998).

[7] Newman, M. E. J. The structure and function of complex networks. SIAM Review, 45(2), 167-256, (2003).

[8] Borgatti, S. P. Centrality and network flow. Social Networks, 27(1), 55-71, (2005).

[9] Fortunato, S. Community detection in graphs. Physics Reports, 486(3-5), 75-174, (2010).

[10] Holme, P., & Saramäki, J. Temporal networks. Physics Reports, 519(3), 97-125, (2012).

[11] Newman, M. E. J. Networks: An Introduction. Oxford University Press, (2010).

[12] West, D. B. Introduction to Graph Theory. Prentice Hall, (2001).

[13] Diestel, R. Graph Theory (5th ed.). Springer, (2017).