CS 579: Online Social Network Analysis

**Project 1 – Social Media Data Analysis**

**"Mapping the Corporate Landscape: NetworkX Visualization of LinkedIn Job Market Connectivity"**

**INTRODUCTION:**

Our project will undertake a complete market dynamics and job availability trend analysis for companies in different locations by gathering data from social media platforms with LinkedIn as the primary source. To this end, we use a structured way to build a Company-Insight network model. In the beginning, we introduce a Bipartite structure that consists of companies and their locations. Subsequently, the bipartite structure of the network is transformed into an unimodal one, thus simplifying the representation for further analysis. Our approach includes data extraction from LinkedIn to collect information on company operations and employment opportunities in various geographical locations. Such mapping of links between companies and their respective locations gives us the opportunity to analyze the expansion strategies of businesses and their job demands in different regions of the country. Also, our study goes further to discover the occurrence of common places of business expansion and the corresponding number of jobs available in that area. Furthermore, we focus on the networks of relationships that are established among companies, especially when they share the same space, which may indicate possible partnerships or investment. What the project offers is a comprehensive data processing, modelling, and analysis to deliver actionable intelligence on market dynamics and job availability trends, which will guide the stakeholders to make informed decisions and strategically move within the corporate environment.

**DATA COLLECTION:**

How you crawled your chosen platform to collect the data?

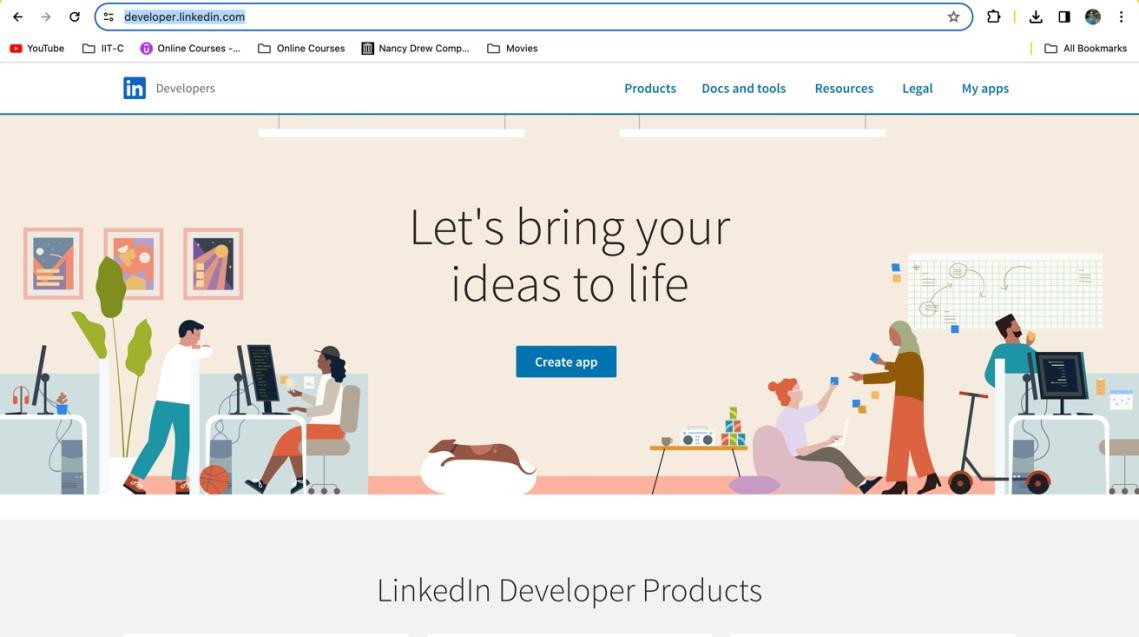
We have collected around 200-300 dataset with company name, location of the company and the job titles using LinkedIn API call with selenium and BS4 web scrape libraries. We have registered an account in developer tool with the use case for the collection of data.

Created an app with default authentication for API to access LinkedIn portal using ClientID and secret code. And using selenium and Beautiful Soup library we collected the required data viz job titles, company, location, weblink. We provide the URL of search page on LinkedIn, with Software engineer as job title and location as united states for the sample. The URL get authenticated using API for the given client id and selenium driver helps to navigate for more pages. Using BS4 library we collected the data under search tags of html. Finally, the data was collected using Pandas data frame and exported as excel.

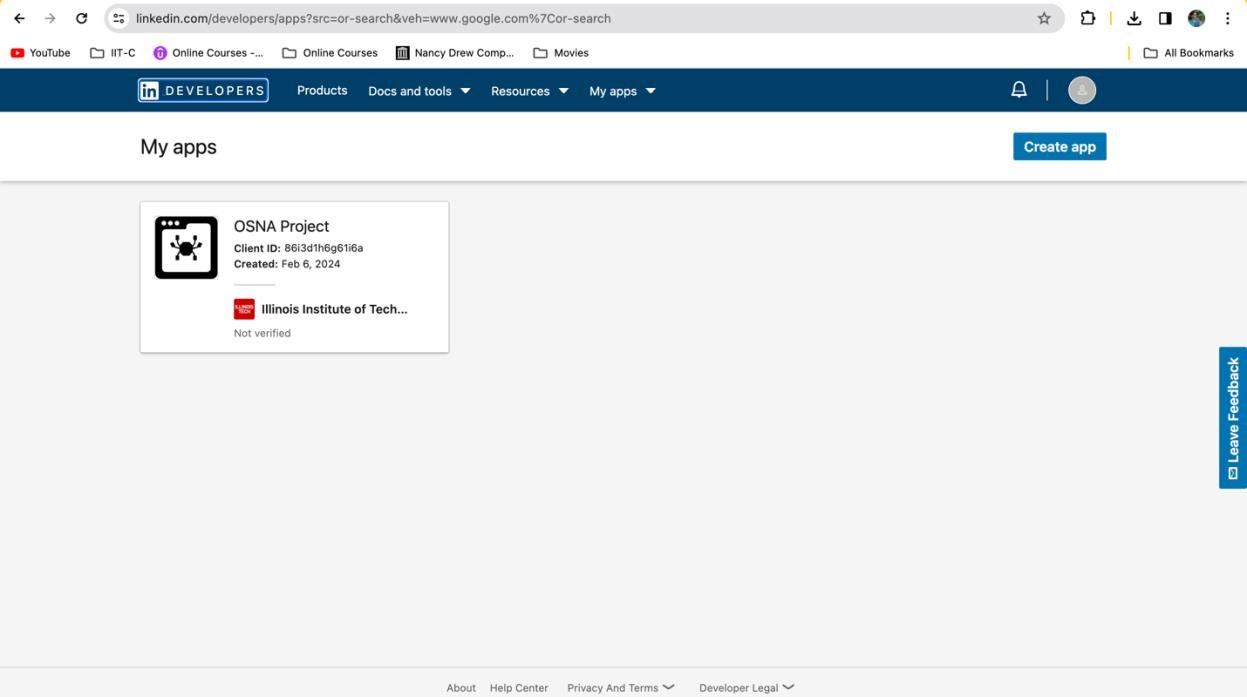
**LinkedIn Developer Portal details:**

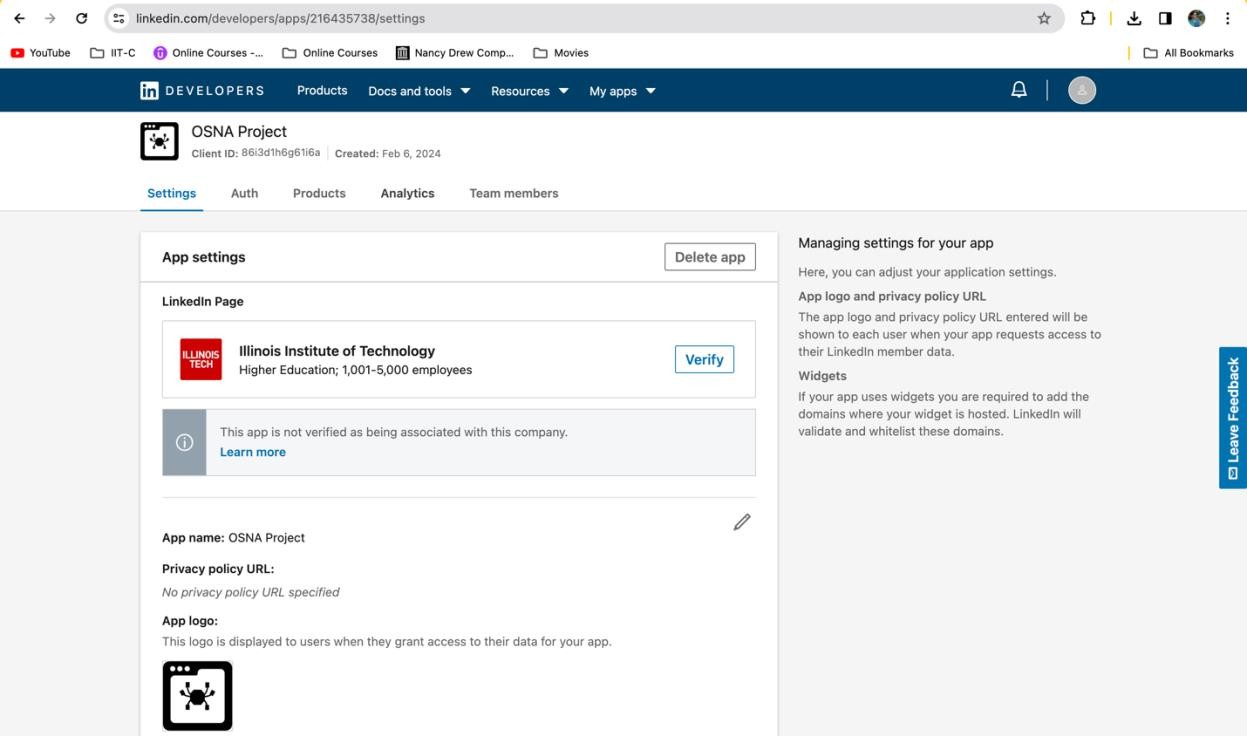
page url :-- <https://developer.linkedin.com/> [https://www.linkedin.com/company/100508559/admin/feed/posts/?feedType=follo](https://www.linkedin.com/company/100508559/admin/feed/posts/?feedType=following) [wing](https://www.linkedin.com/company/100508559/admin/feed/posts/?feedType=following)

**LinkedIn Developer portal Screenshots:**

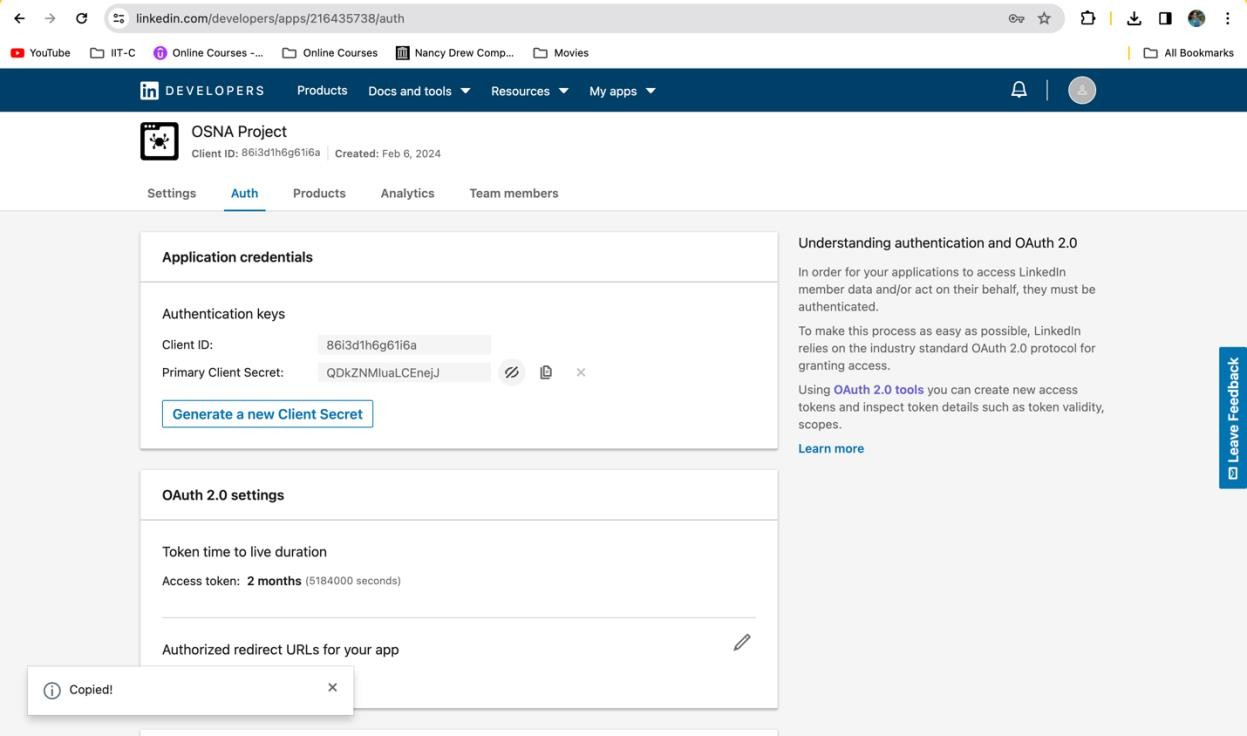


**Developer app created for the Project:**

****



**Client ID and Details:**

****

Clientid='86i3d1h6g61i6a' Secretcode-'QDkZNMIuaLCEnejJ'

**DATA COLLECTION CODE:**

This script is meant for automating the process of collecting job listings information from LinkedIn using Python. It does this by way of Selenium package, that controls a Chrome web browser, which goes to a specific LinkedIn job search URL. The script is authenticated using self-serving API ids. The script goes through the page many times loading all the content as jobs on the landing page are dynamic. Then, it loads the necessary content and uses BeautifulSoup to extract the text on the page and then parse the HTML, which gives it details about each job listing, like the job title, company name, job location, and a direct link to the job post. After this, the data is structured into a panda DataFrame that is a tabular data structure to make it easy to manipulate and analyze. Then, the DataFrame is saved as 'job\_listings1.xlsx' in user's Desktop which gives an easy-to-use summary of job listings scraped from LinkedIn. Additionally, the script generates an approving message after a successful export of the data. This automation is of special importance to job seekers or researchers trying to investigate the job market shifts, because it makes the data collection quicker.

**Challenges Faced:**

The data collection process was initially limited to scraping data from only 25 nodes for one single page, hence data for graph analysis was constrained significantly. Realizing this deficiency, we looked at the possibility of using the web Chrome driver technique to extract data from multiple pages on LinkedIn. This adjustment increased our data set by an order of magnitude and allowed us to collect data for between 200-300 nodes. On the other hand, the augmented data penetration necessitated magnified time commitments. However, the extended effort remains critical in achieving our aim which is to create a broader range of samples for our graph study.

**API Data usage policy:**

We have used Self-Serve API Program per API user policy to develop Applications using APIs that are available upon registration for an API key on the Developer Site- https://[www.linkedin.com/legal/l/api-terms-of-use.](http://www.linkedin.com/legal/l/api-terms-of-use)

Privacy Policy on the data we collect: <https://www.linkedin.com/legal/privacy-policy>

# DATA VISUALIZATION:

To visualize our network plot, we have chosen the NetworkX library, a powerful and multifunctional Python package for the creation, manipulation, and research of networks with nodes as vertices and edges as entities between them. The network analysis done using NetworkX was selected over the other available packages like SNAP, Gephi, NodeXL, and graph-tool, which were influenced by several important factors. Firstly, Python programming environment is one of the components that makes NetworkX easily integrate with the programming language that makes data scientists and researchers familiar with Python to access the network seamlessly. Furthermore, NetworkX comes with a great deal of support for different types of networks, from the simplest undirected graphs to the most complex, multi- relational networks, which makes it perfect in terms of our project requirements.

**Reasoning for Software Choice**:

NetworkX distinctly stands out due to its excellent documentation and active community support thus making it relatively easier for learners to use. Additionally, it provides the ability of directly modifying the graph properties and attributes using Python that in turn allows for a more hands-on and interactive approach to graph analysis and visualization. This capacity is especially beneficial for our project, which is a very complex one, that needs to study the firms-locations networks. Picking NetworkX, we connect to the modularity and scalability strategy of our project that can be effortlessly extended or modified in the future.

**Graph Visualization:**

The data for the project used in NetworkX graph visualization was structured as a series of edges connecting nodes, where each node can represent a firm or a location, and each edge shows the existence of a company at a given location. This format is especially suited as it provides flexibility in the creation of complex network structures without the need for conversion to adjacency matrix or list. The data processing happens by our Python script that iterates each row from our dataset, extracting the desired company and location information to prepare the nodes and link them in the graph. The combination of NetworkX and matplotlib can visualize the network structure leading to the creation of interesting visualizations. Our first visualization delivers a general outline, clearly indicating the bipartite structure of the network by using different node colors for companies and locations and by showing the edges leading to affiliations. The refinement of the visualization has been improved by adjusting the node sizes according to degrees to emphasize the highly connected nodes and by using the edge transparency to show the density of connections, so the graph has more readability and analytical value. The analysis is deepened by calculating the modularity of the network, which aids in identifying community structure of the network. A high modularity score is a clear hint that the network is subdivided into modules or communities, a feature that was consistent in our re-rendered diagrams. These communities may be markers or aggregates of closely related firms associated with specific geographic areas, bringing knowledge on the regional economic landscapes or the industry-specific hubs.

**Conversion of Bipartite Graph into Unimodal Graph:**

We further analyzed the affiliation network by projecting it onto a unimodal network, focusing solely on companies and their shared locations. This projection allowed us to identify clusters of companies sharing common locations. To study the network's structure deeper, the bipartite graph was converted into a mono-modal network that concentrates only on the links between the companies that have the same location. These projections enabled us to single out the central nodes (companies) given the connectivity degree within the network and indicating the potential key players in the industry or region.

Modularity of this unimodal network has been refined by removing outliers and recalculation of modularity resulted in the clear view of the central structure and exposed the most interconnected companies and the possible revealing of strategic partnerships or market dominance patterns. The tool NetworkX has been a real asset for us, providing clear insights into the complexity of the relationships between companies and locations when used as a visualizer and analyzer of our network graph. Using strategic data visualization methods as well as careful analysis, we have seen structure and pattern in the network which may direct future research, determine business strategy, or influence the economic policy. The flexibility and power offered by NetworkX, as well as the compatibility between it and Python, have turned it into an invaluable tool in our project.

The graph visualization provides a comprehensive overview of the dataset, depicting two distinct node types: jobs and places. Edges connect each company node to its corresponding location, and thereby represent the spatial presence of these companies. Node size is related to the node degree, marking those nodes with higher connectivity. Using a force-directed layout exposes communities and groups among the graph that are then reinforced by a high modularity score, implying strong community structure. Besides this, a unimodal projection makes the graph clearer and more compact by isolating companies instead of noise nodes. Though unimodal network has a slightly lower modularity, it is still essential in revealing information about economic activity, regions of business hubs, and market dynamics. This code fragment employs the NetworkX library to convert a bipartite graph on the companies and their locations into a unimodal network with one type of nodes. It starts with importing relevant libraries like pandas, networkx, and matplotlib.pyplot The code thereafter initializes the bipartite graph by looping through a DataFrame and creating vertices for companies and locations while establishing edges between them. Then, it builds a set of company nodes and

projects the bipartite graph to a unimodal one by removing the connections between the companies that do not have the same location. Visualizing the final network requires setting up the layout of the graph including node positioning and appearance personalizing, and then displaying it as an image. In general, the code enables the investigation of correlations, patterns, and clusters in the network which can be informative regarding corporate penetration, market competition, and regional dynamics of industries.

# NETWORK MEASURES:

**Degree Distribution**:

The degrees of a node within a network are the number of its ties to other nodes. The Degree Distribution is one of the most crucial measures in network analysis that allows for the study of the connections at a macroscopic level. It is a moment occurred when the auxiliary information is presented of connections distribution among nodes of the network. NetworkX was used to compute the degree of each node and it was featured in a histogram which indicates the nodes with the corresponding frequencies of the degrees. The histogram of Degree Distribution (i.e. it's the line chart with degrees on the X-axis and a degree count on the Y-axis) is a right skew suggesting that most nodes have a low degree, while the remainder has a high degree. This points to scale-free network, which is typical for real-world networks in which there exist a few highly complex hubs.

**Closeness Centrality:**

Centrality of the closeness indicates how far the node is from all the other nodes in the network. It is, in other words, a reverse of the average length of the shortest paths between the node and all other nodes. Closeness centrality of the nodes of a network that are well connected will result in a better dissemination of the information in the network.

Visualization of Closeness Centrality on our network, which is in the form of a bimodal distribution, is captured. This indicates the presence of two distinct groups within the network: the first group lying somewhere around the core of the network, possibly around the central node with markedly higher closeness centrality, and another group at a lower closeness centrality, likely on the periphery of the network.

**Betweenness Centrality:**

Betweenness Centrality is used to measure the number of times a node is a bridge along the shortest paths between two other nodes. It signifies a node with brokering or intermediation capacity giving an insight into the locations of information flow control points in the network.

There is a very pronounced right-skewed distribution of Betweenness Centrality among nodes that consist of the large number of low values and a peak very close to zero. Such an observation implies that just a few nodes are frequently visited on the shortest paths and that the rest of the nodes have less betweenness centrality that highlights the role of critical connectors or bottlenecks within the network.

**Result:**

A multi-faceted view of the network structure through the analysis of Degree Distribution, Closeness Centrality, and Betweenness Centrality is given. The Degree Distribution which is scale-free implies that our network may be resilient to random failures however, prone to intentional attacks on the high degree nodes. The bimodal Closeness Centrality signifies a possible splitting within the network between the influential nodes and those nodes less central. It is clear from the skewness of Betweenness Centrality that this is a network in which most nodes have no control on the flow of information and only a few controls the whole data transmission. The outcomes raised a few questions which require further inquiry. Which factors are responsible for high scores of the centrality of some nodes? Do they reflect organizational hierarchies or communities that reside in a network? Can the nodes with the high betweenness centrality be vital in the disseminating of information or diseases depending on the network's context?

The next step after that is the running of a community detection analysis to find the tightly knit groups or clusters that are present in the network. Furthermore, the traits of the high-centrality nodes may give more information about their function and character. More thorough simulation, which might include targeted removal of nodes, could produce more insights about the network's resilience.

# CONCLUSION:

The application of NetworkX for the extraction and visualization of the key network measures has shed light on the complex network architecture revealing its strengths and weaknesses. The data acquired through this analysis are not only a

basis for our foundational knowledge of the network's structure but also, they pave the way for the focused research that uses this structure in practical applications or academic research. More layers will be peeled off as we move further into the network and uncover the substrata of connectivity and influence, which will drive knowledge-based decision-making and stimulate new research in the field. The advantages of the visualization and analysis of an affiliation network using a tool such as NetworkX are numerous. undefined

1. **Strategic Insights:** Finding out the connections between the companies and their locations may provide strategic business insights. For example, companies can find places for expansion, concentrations of industries, or high-competition regions.
2. **Market Analysis:** The project would give a microcosm view of the job market, displaying the spatial distribution of job opportunities and hence could be useful to workforce development agencies, job seekers and policymakers.
3. **Data-Driven Decisions:** Visualization of complex networks assists in taking decisions based on empirical data. Stakeholders can identify key influencers, central hubs, and whole network structure which can direct marking, investment, and operational strategies.
4. **Community Detection:** Through modularity analysis, the project will find community structures inside the network. It can be especially useful in identifying groups of firms that have the same interests or are anticipated to cooperate, thereby creating business ecosystems.
5. **Resource Optimization:** Companies can optimize their resources by focusing on specific locations that act as the center of their industry's network. This results in the savings of marketing and logistics resources.
6. **Network Resilience Analysis:** The project can test the strength of the network through the identification of the critical nodes whose elimination can cause network disruption. This is very useful for risk assessment as well as planning in supply chain management.
7. **Academic Contributions:** The project is an applied contribution to the field of network science where theoretical measures are applied to real-world data, shedding new light on how networks of affiliation function.
8. **Skill Development:** The project becomes a platform for people to learn and develop skills in data science. The skills people will learn include data collection and cleaning, network analysis, algorithm implementation, and statistical interpretation.
9. **Technological Proficiency:** Having mastery with tools such as NetworkX, pandas, and Matplotlib is very advantageous since they are the most popular tools in the industry for almost every type of data analysis task.
10. **Predictive Analytics:** Insights from the current network structure can be used

to forecast the future dynamics of job markets, e.g., booming regions of industry or areas undergoing decline.

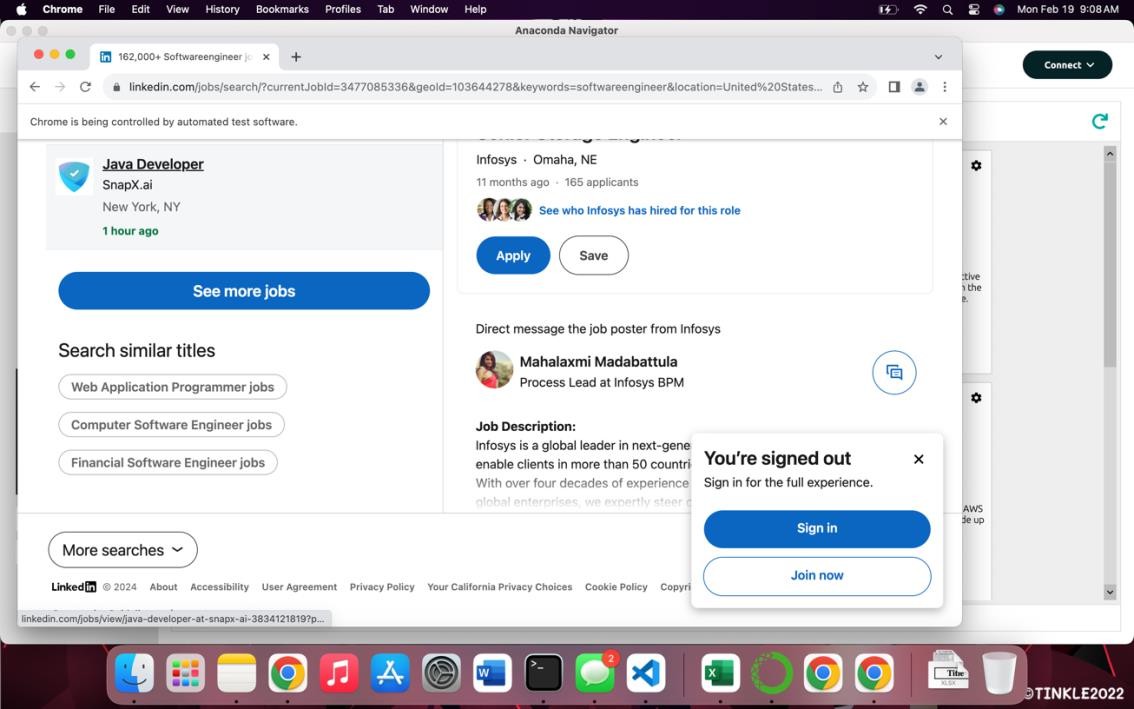
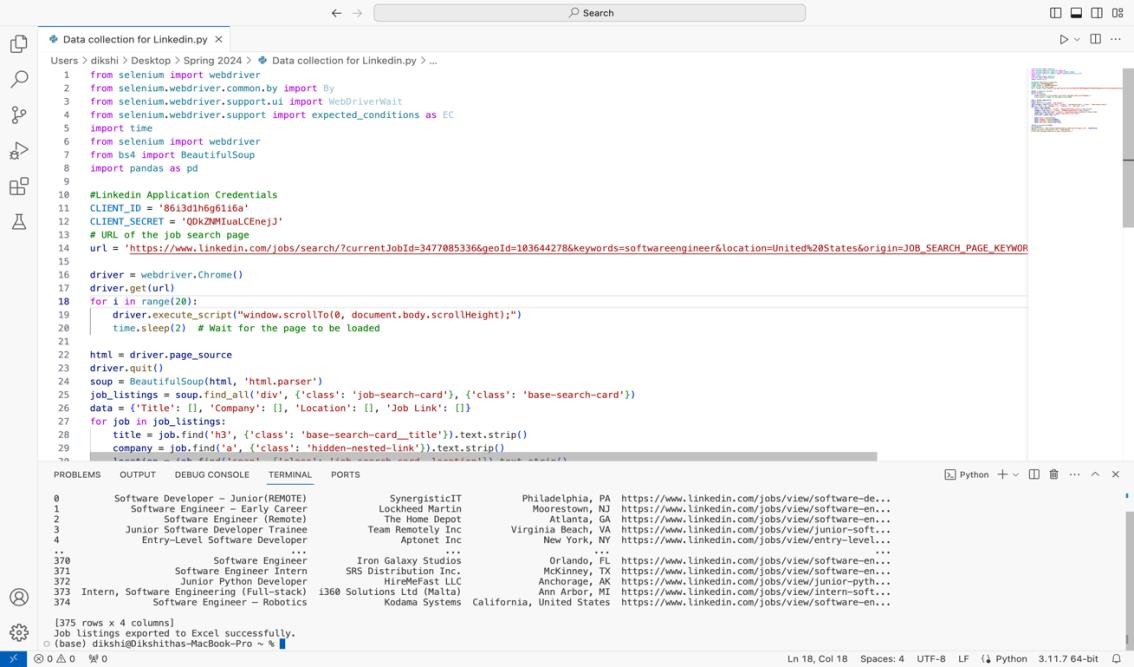
This project not only ensures that the stakeholders understand the concept of the interconnectedness of the corporate world but also provides them strategic initiatives by developed their analytical skills and also contributes to the body of knowledge on network theory and its applications.

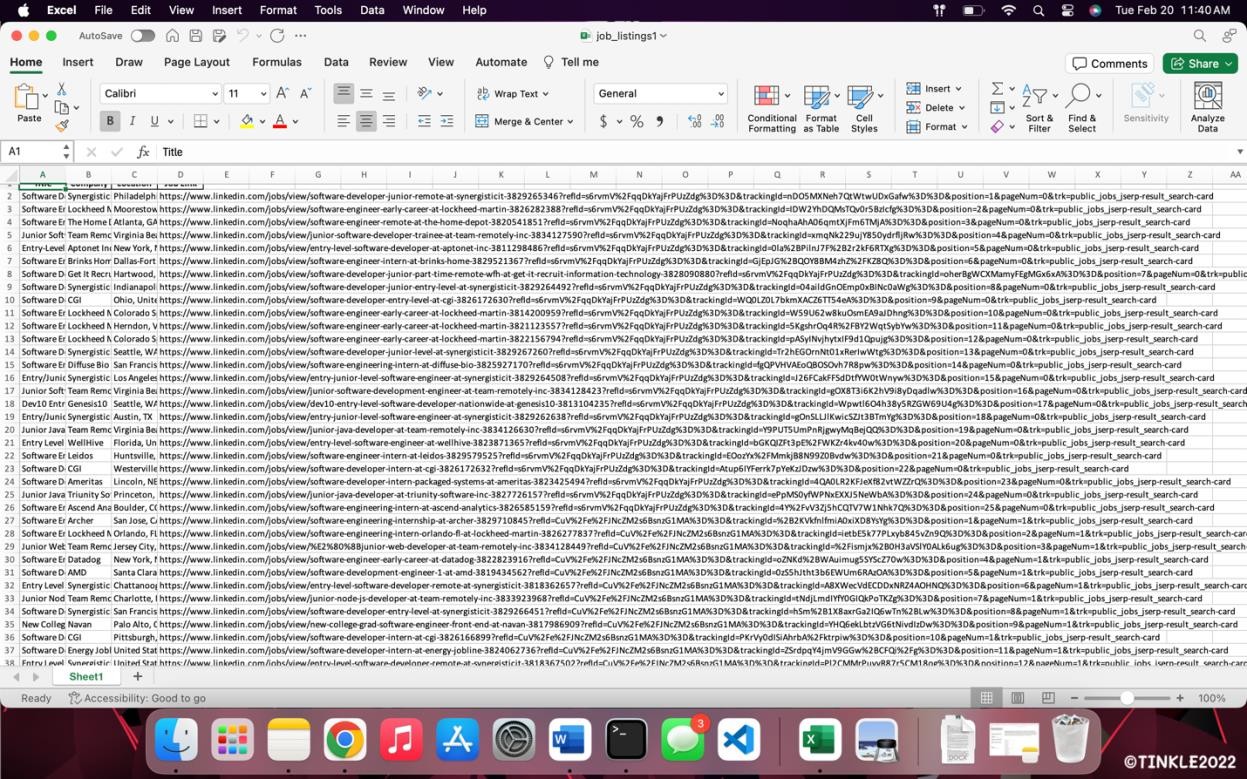
# REFERENCES:

* 1. https://[www.youtube.com/watch?v=3vwKBO9YkBI](http://www.youtube.com/watch?v=3vwKBO9YkBI)
  2. https://github.com/vyasadi/LinkedIn-Job-Scraping/blob/main/linkedinjobspython.ipynb
  3. https://sites.google.com/chromium.org/driver/getting-started - chrome driver installations
  4. https://chat.openai.com/c/7333fc99-21cb-48cb-a2bf-82121cc57519 - chat gpt
  5. https://anaconda.org/conda-forge/selenium - selenium installation
  6. https://api-university.com/blog/api-usage/how-to-use-the-linkedin-api-and-oauth
  7. https://[www.youtube.com/watch?v=jYflkIo1R4A](http://www.youtube.com/watch?v=jYflkIo1R4A)
  8. https://networkx.org/documentation/networkx-1.9.1/\_downloads/networkx\_tutorial.pdf
  9. https://[www.mathworks.com/help/matlab/getting-started-with-matlab.html](http://www.mathworks.com/help/matlab/getting-started-with-matlab.html)
  10. https://python-louvain.readthedocs.io/en/latest/
  11. https://reference.wolfram.com/language/guide/GraphMeasures.html

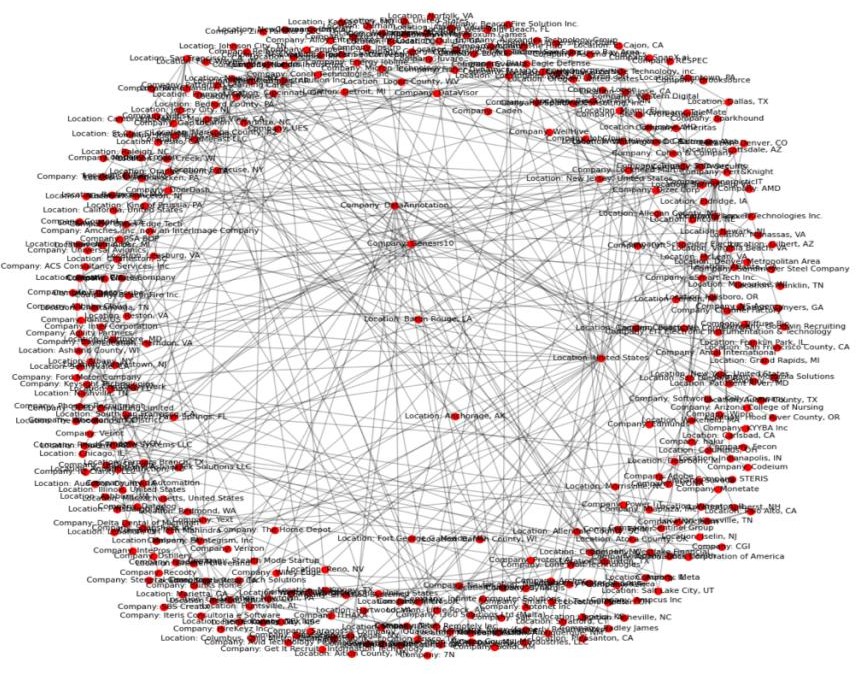
# OUTPUT SNIPPETS:

**DATA COLLECTION:**

****

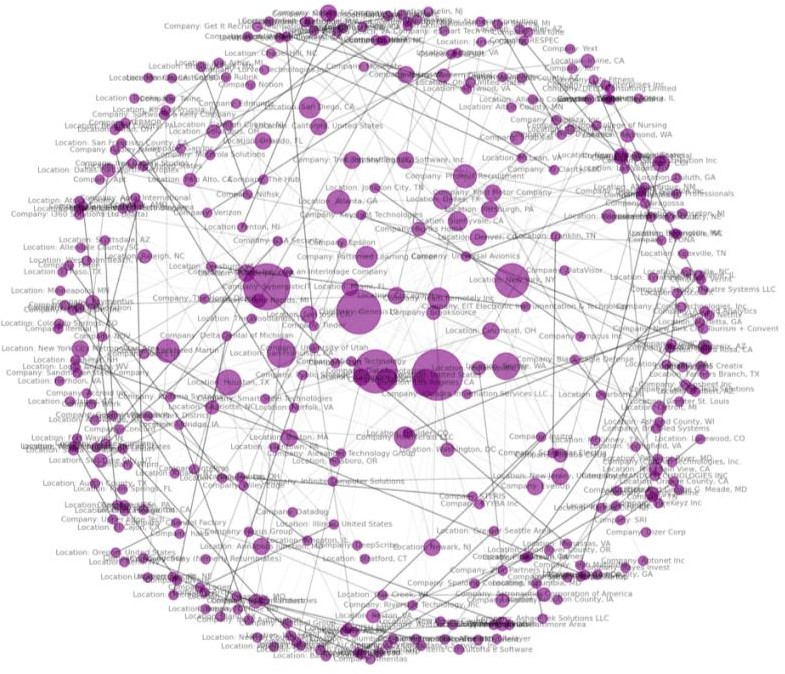


**AFFLIATION NETWORK:**

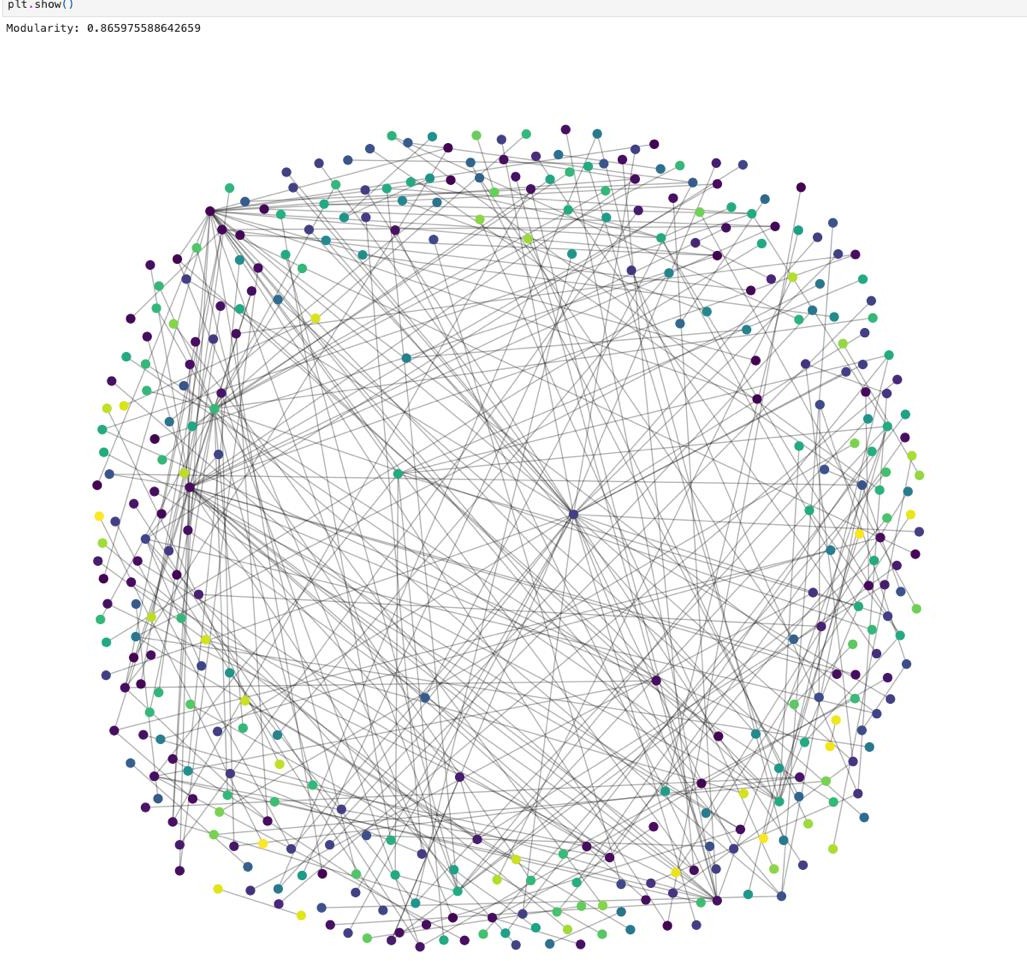
****

**REFINING THE NETWORK:**

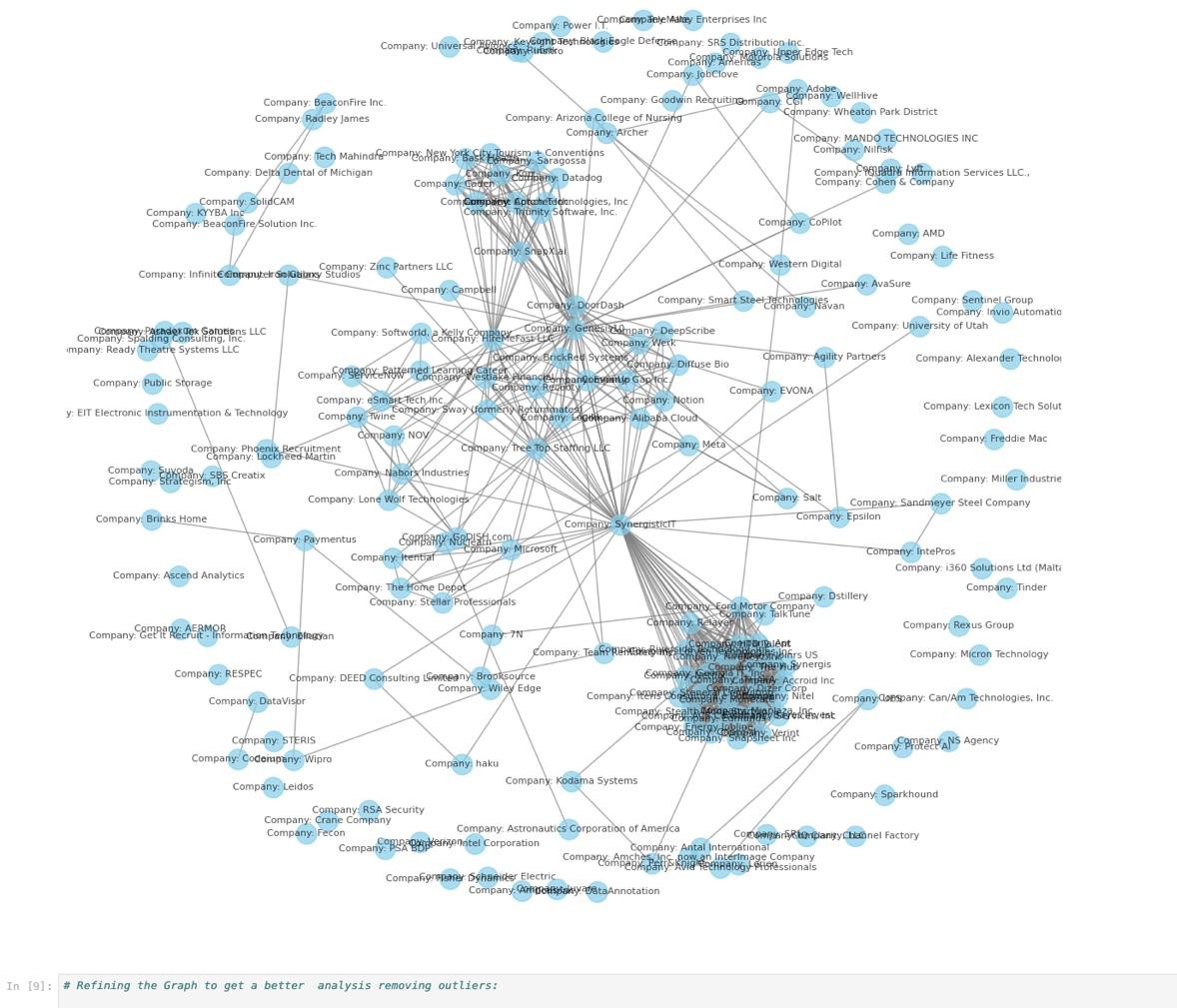
The size of the nodes in the refined graph is determined by the degree of the nodes (the number of edges connected to a node). This means larger nodes have more connections, indicating they are either companies with multiple locations or locations with many companies.



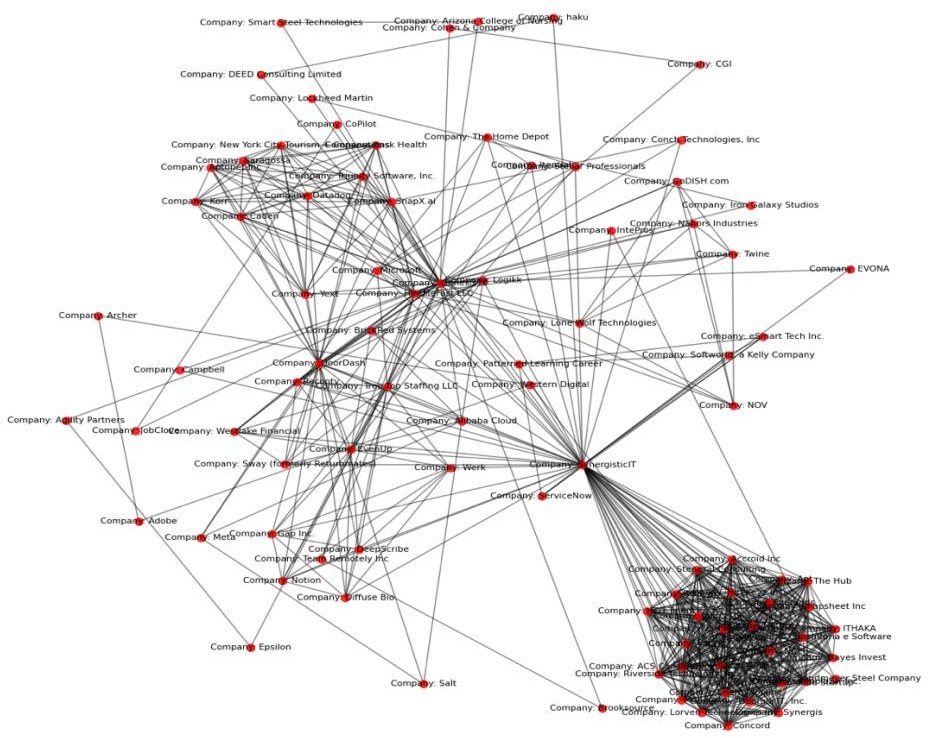
**MODULARITY CALCULATION:**

****

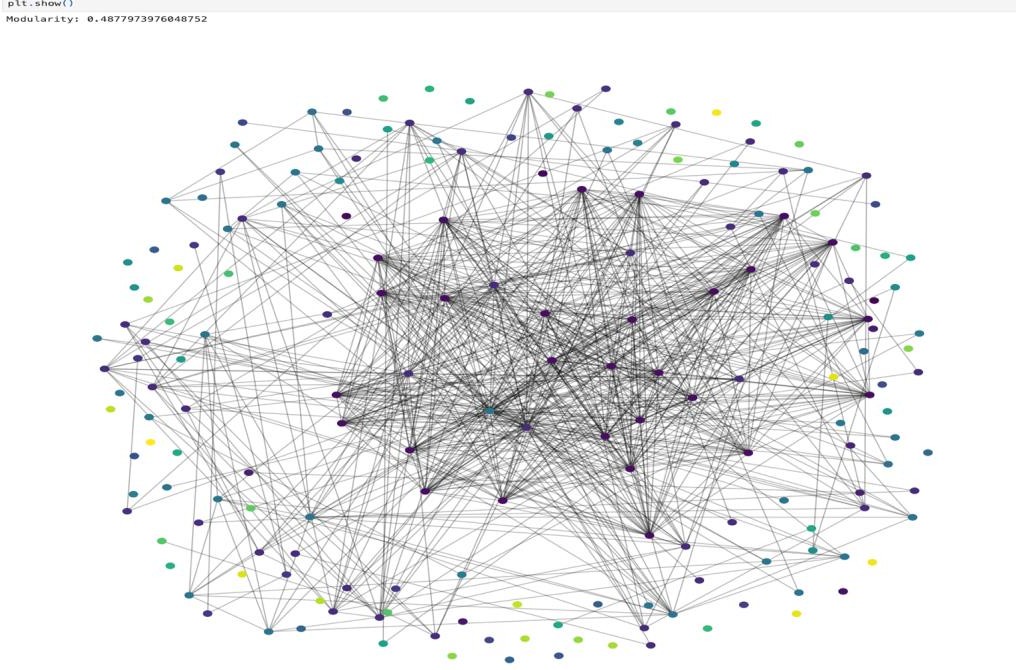
**CONVERTING THE AFFILIATION NETWORK TO UNIMODEL NETWORK:**

****

**REFINING THE NETWORK BY REMOVING THE OUTLIERS:**

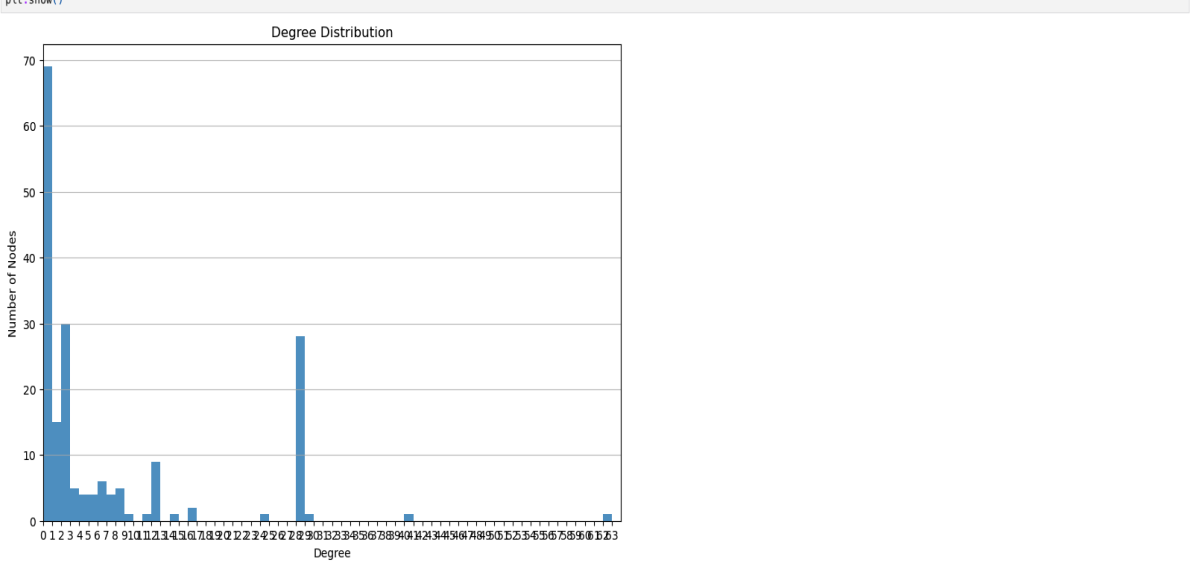
****

**MODULARITY CALCULATION:**

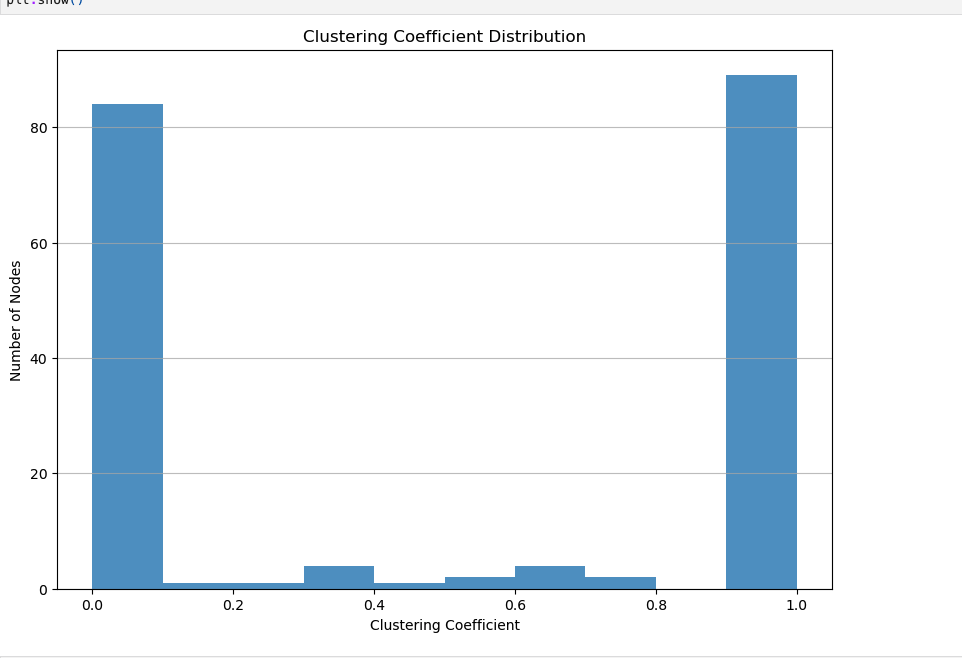
****

**NETWORK MEASURES:**

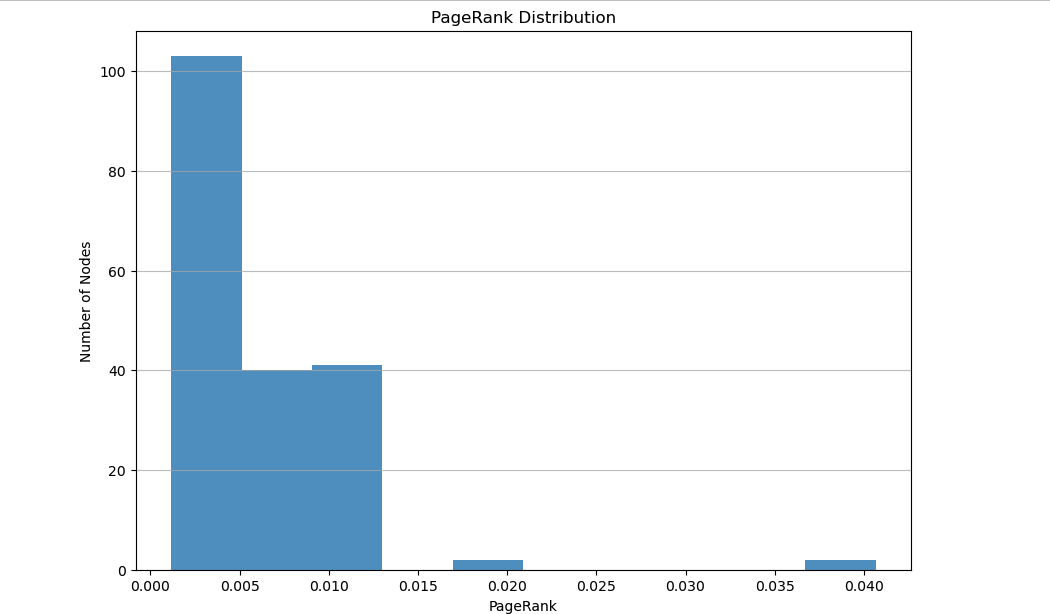
**DEGREE DISTRIBUTION:**

****

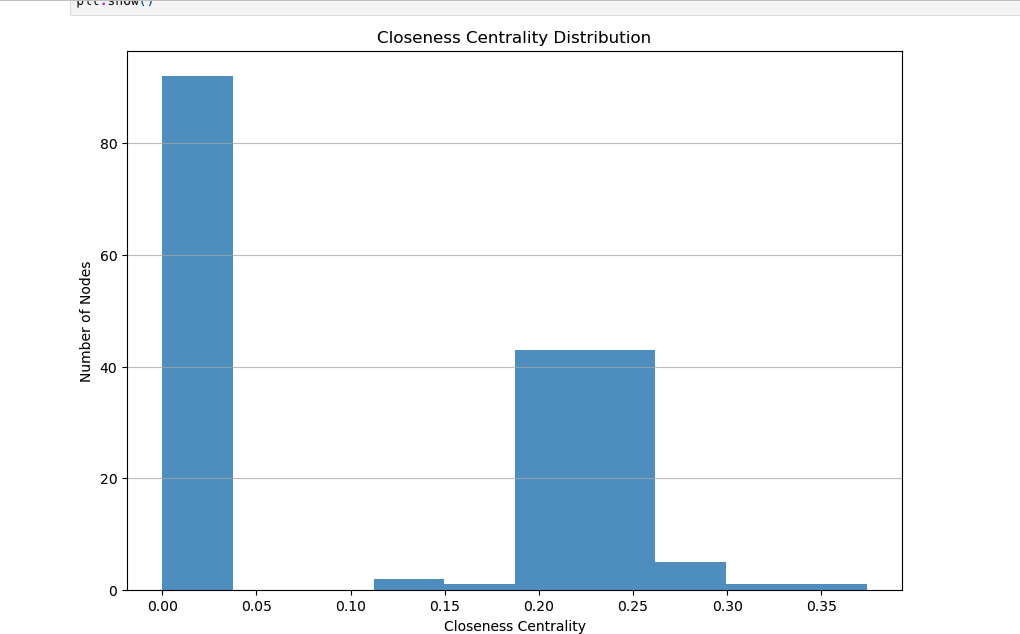
**CLUSTERING COEFFICIENT:**

****

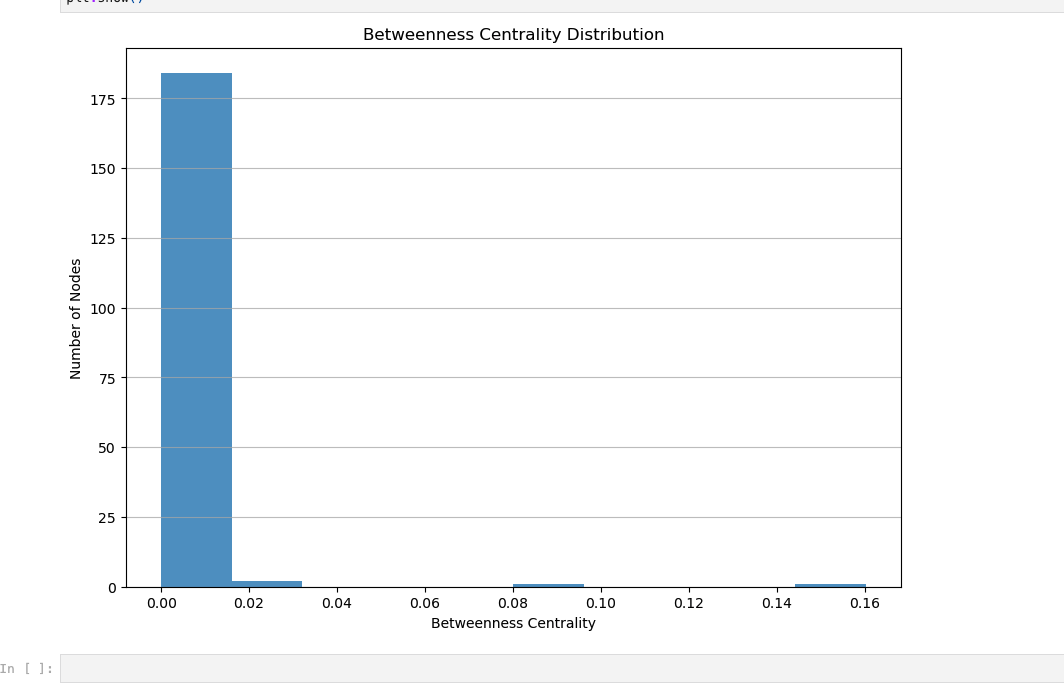
**PAGE RANK DISTRIBUTION:**

****

**CLOSENESS CENTRALITY DISTRIBUTION:**

****

**BETWEENESS CENTRALITY DISTRIBUTION:**

****