CS342 - SOCIAL NETWORK ANALYSIS

MINI_PROJECT

Submitted By:

Name: Eswar Pesala,Kumudini Gholap

Roll No: U22CS133,U22CS134

Branch: CSE

Semester: 6TH Sem

Division: B

Department of Computer Science and Engineering

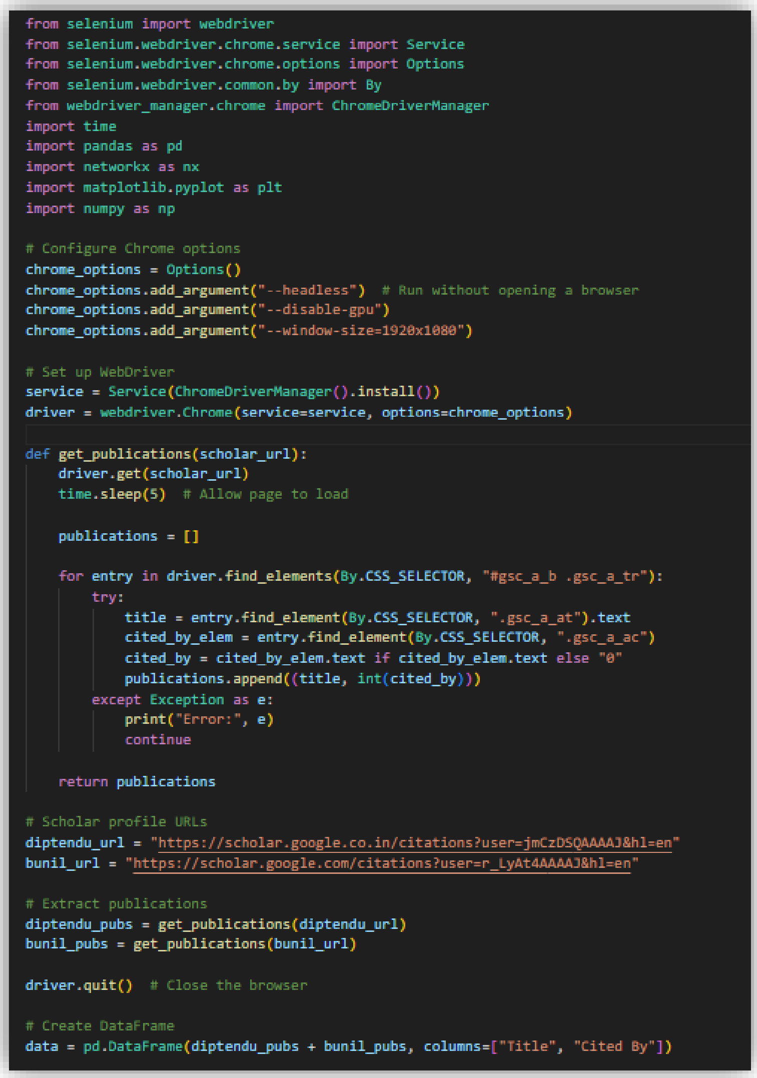
SV NATIONAL INSTITUTE OF TECHNOLOGY

SURAT

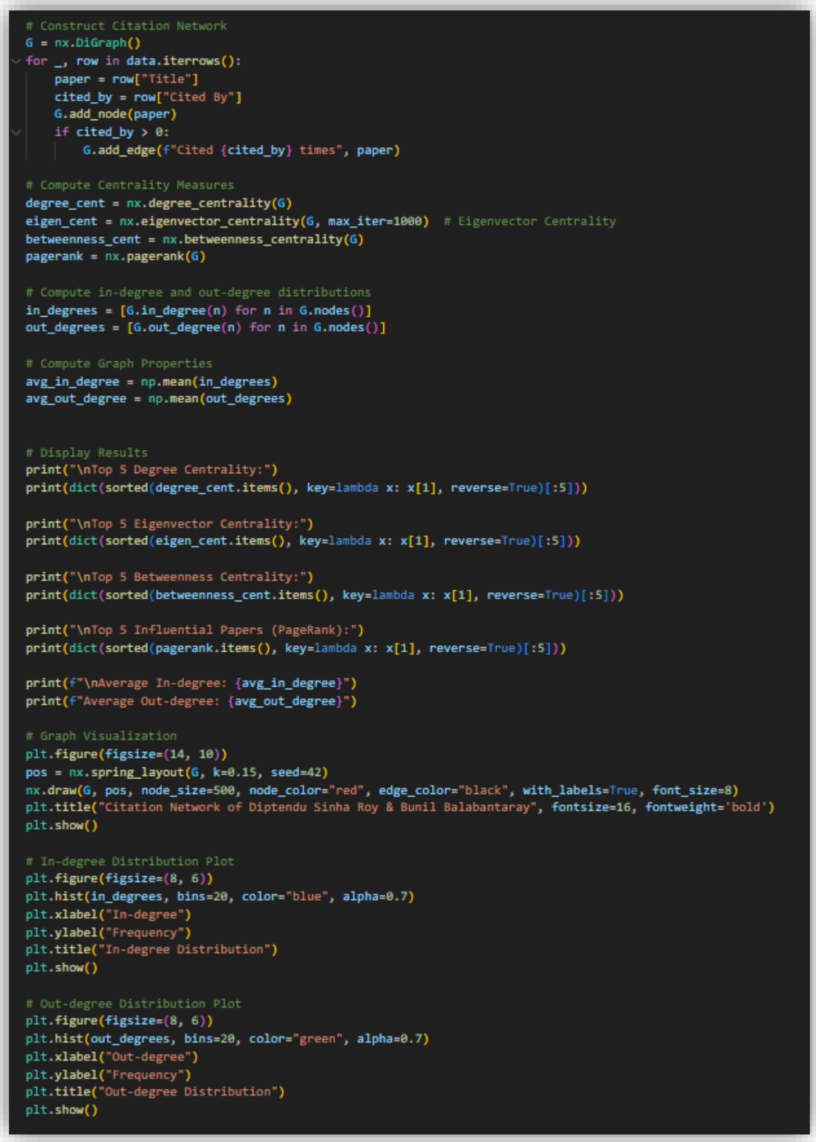
Citation_Network_Project

- **The problems statement is regarding citation network institute wise. For case study, consider one department, one or two professor with last two -three years publications. Using this data construct the citation network with different graph based statistical measure, degree of centrality, influence, degree of connectivity. Demonstrate your finding with appropriate data plot.**

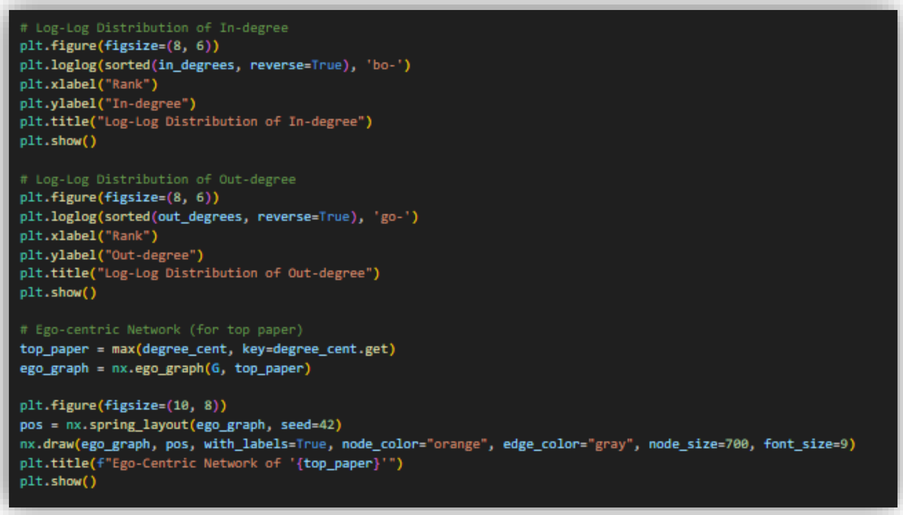
➢ *Code for NIT_MEGHALAYA:-*



```python
from selenium import webdriver
from selenium.webdriver.chrome.service import Service
from selenium.webdriver.chrome.options import Options
from selenium.webdriver.common.by import By
from webdriver_manager.chrome import ChromeDriverManager
import time
import pandas as pd
import networkx as nx
import matplotlib.pyplot as plt
import numpy as np

# Configure Chrome options
chrome_options = Options()
chrome_options.add_argument("--headless")  # Run without opening a browser
chrome_options.add_argument("--disable-gpu")
chrome_options.add_argument("--window-size=1920x1080")

# Set up WebDriver
service = Service(ChromeDriverManager().install())
driver = webdriver.Chrome(service=service, options=chrome_options)

def get_publications(scholar_url):
    driver.get(scholar_url)
    time.sleep(5)  # Allow page to load

    publications = []

    for entry in driver.find_elements(By.CSS_SELECTOR, "#gsc_a_b .gsc_a_tr"):
        try:
            title = entry.find_element(By.CSS_SELECTOR, ".gsc_a_at").text
            cited_by_elem = entry.find_element(By.CSS_SELECTOR, ".gsc_a_ac")
            cited_by = cited_by_elem.text if cited_by_elem.text else "0"
            publications.append((title, int(cited_by)))
        except Exception as e:
            print("Error:", e)
            continue

    return publications

# Scholar profile URLs
diptendu_url = "https://scholar.google.co.in/citations?user=jmCzDSQAAAAJ&hl=en"
bunil_url = "https://scholar.google.com/citations?user=r_LyAt4AAAAJ&hl=en"

# Extract publications
diptendu_pubs = get_publications(diptendu_url)
bunil_pubs = get_publications(bunil_url)

driver.quit()  # Close the browser

# Create DataFrame
data = pd.DataFrame(diptendu_pubs + bunil_pubs, columns=["Title", "Cited By"])
```

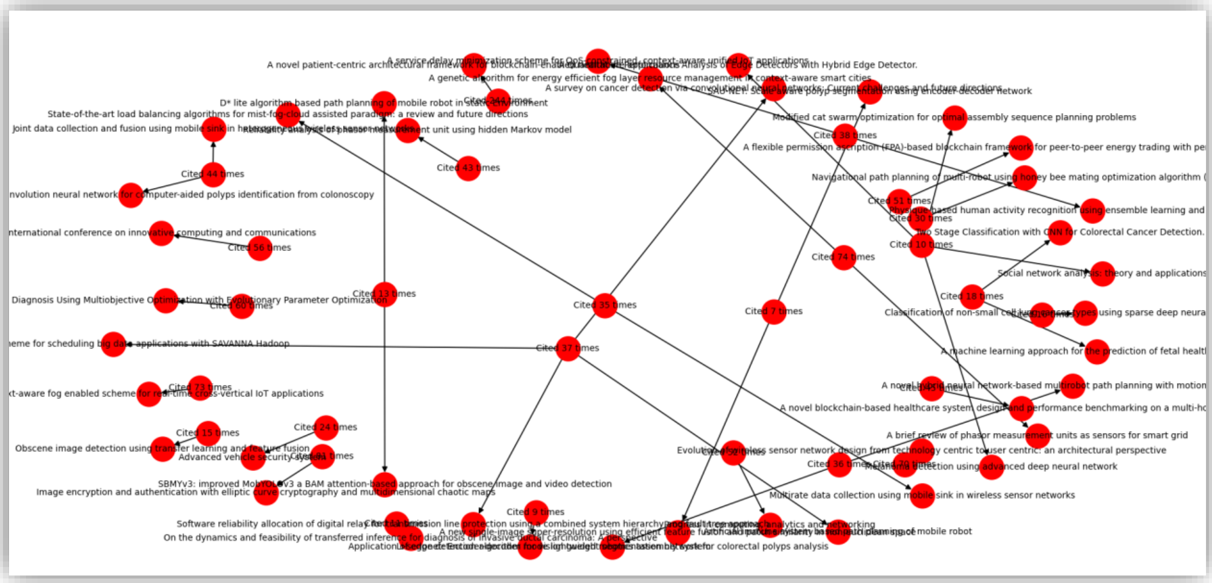


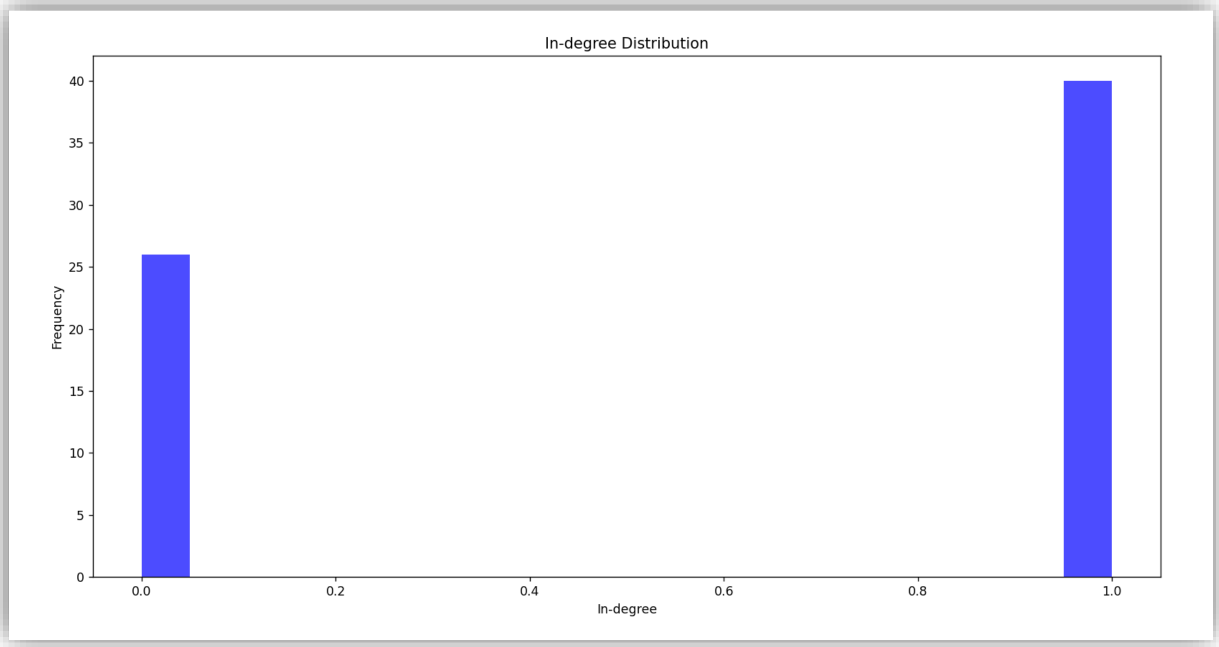
```python
# Construct Citation Network
G = nx.DiGraph()
for _, row in data.iterrows():
    paper = row["Title"]
    cited_by = row["Cited By"]
    G.add_node(paper)
    if cited_by > 0:
        G.add_edge(f"Cited {cited_by} times", paper)

# Compute Centrality Measures
degree_cent = nx.degree_centrality(G)
eigen_cent = nx.eigenvector_centrality(G, max_iter=1000)  # Eigenvector Centrality
betweenness_cent = nx.betweenness_centrality(G)
pagerank = nx.pagerank(G)

# Compute in-degree and out-degree distributions
in_degrees = [G.in_degree(n) for n in G.nodes()]
out_degrees = [G.out_degree(n) for n in G.nodes()]

# Compute Graph Properties
avg_in_degree = np.mean(in_degrees)
avg_out_degree = np.mean(out_degrees)


# Display Results
print("\nTop 5 Degree Centrality:")
print(dict(sorted(degree_cent.items(), key=lambda x: x[1], reverse=True)[:5]))

print("\nTop 5 Eigenvector Centrality:")
print(dict(sorted(eigen_cent.items(), key=lambda x: x[1], reverse=True)[:5]))

print("\nTop 5 Betweenness Centrality:")
print(dict(sorted(betweenness_cent.items(), key=lambda x: x[1], reverse=True)[:5]))

print("\nTop 5 Influential Papers (PageRank):")
print(dict(sorted(pagerank.items(), key=lambda x: x[1], reverse=True)[:5]))

print(f"\nAverage In-degree: {avg_in_degree}")
print(f"Average Out-degree: {avg_out_degree}")

# Graph Visualization
plt.figure(figsize=(14, 10))
pos = nx.spring_layout(G, k=0.15, seed=42)
nx.draw(G, pos, node_size=500, node_color="red", edge_color="black", with_labels=True, font_size=8)
plt.title("Citation Network of Diptendu Sinha Roy & Bunil Balabantaray", fontsize=16, fontweight='bold')
plt.show()

# In-degree Distribution Plot
plt.figure(figsize=(8, 6))
plt.hist(in_degrees, bins=20, color="blue", alpha=0.7)
plt.xlabel("In-degree")
plt.ylabel("Frequency")
plt.title("In-degree Distribution")
plt.show()

# Out-degree Distribution Plot
plt.figure(figsize=(8, 6))
plt.hist(out_degrees, bins=20, color="green", alpha=0.7)
plt.xlabel("Out-degree")
plt.ylabel("Frequency")
plt.title("Out-degree Distribution")
plt.show()
```

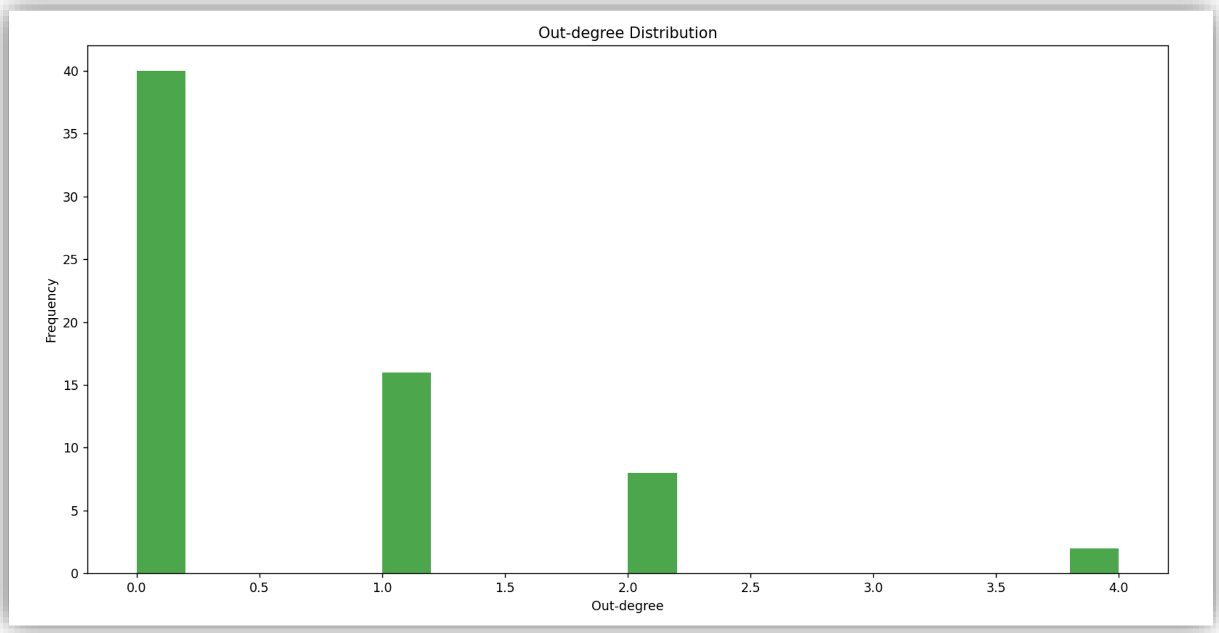


```python
# Log-Log Distribution of In-degree
plt.figure(figsize=(8, 6))
plt.loglog(sorted(in_degrees, reverse=True), 'bo-')
plt.xlabel("Rank")
plt.ylabel("In-degree")
plt.title("Log-Log Distribution of In-degree")
plt.show()

# Log-Log Distribution of Out-degree
plt.figure(figsize=(8, 6))
plt.loglog(sorted(out_degrees, reverse=True), 'go-')
plt.xlabel("Rank")
plt.ylabel("Out-degree")
plt.title("Log-Log Distribution of Out-degree")
plt.show()

# Ego-centric Network (for top paper)
top_paper = max(degree_cent, key=degree_cent.get)
ego_graph = nx.ego_graph(G, top_paper)

plt.figure(figsize=(10, 8))
pos = nx.spring_layout(ego_graph, seed=42)
nx.draw(ego_graph, pos, with_labels=True, node_color="orange", edge_color="gray", node_size=700, font_size=9)
plt.title(f"Ego-Centric Network of '{top_paper}'")
plt.show()
```

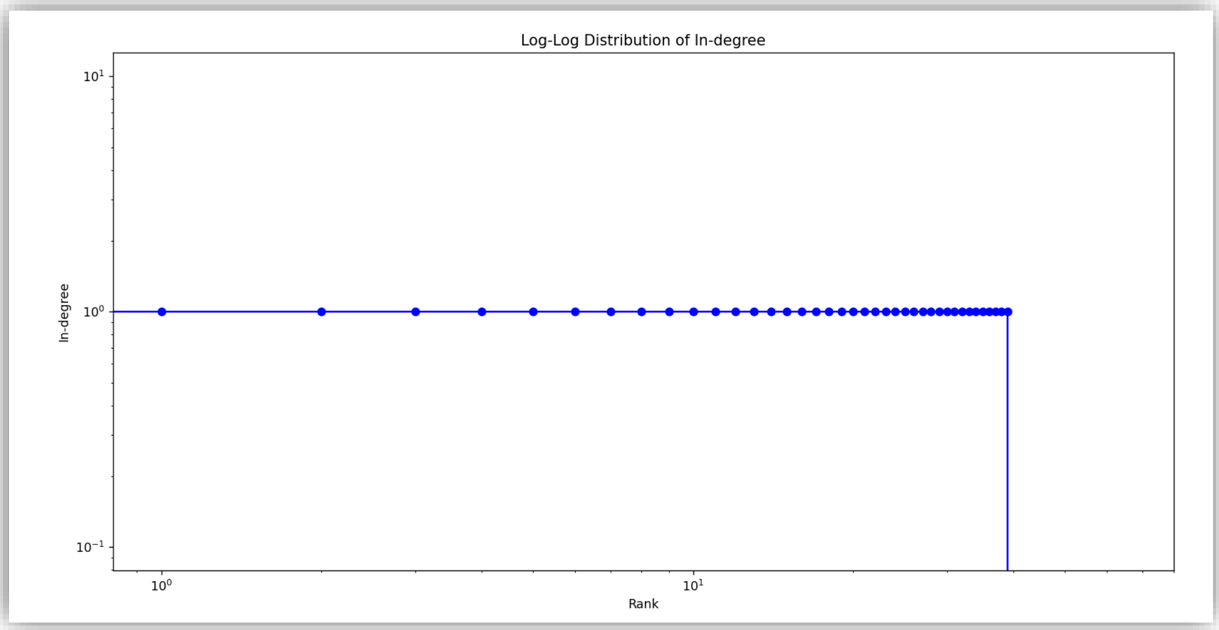
➢ Outputs_for NIT_MEGHALAYA:-


A novel patient-centric architectural framework for blockchain-enabled healthcare applications
A service delay minimization scheme for QoS-constrained, context-aware unified IoT applications
Analysis of Edge Detectors with Hybrid Edge Detector.
A genetic algorithm for energy efficient fog layer resource management in context-aware smart cities
A survey on cancer detection via convolutional neural networks: Current challenges and future directions
SAU-NET: Scale aware polyp segmentation using encoder-decoder network
D* lite algorithm based path planning of mobile robot in static Environment
Cited 244 times
State-of-the-art load balancing algorithms for mist-fog-cloud assisted paradigm: a review and future directions
Joint data collection and fusion using mobile sink in heterogeneous wireless sensor networks
Reliability analysis of phasor measurement unit using hidden Markov model
Modified cat swarm optimization for optimal assembly sequence planning problems
Cited 38 times
A flexible permission ascription (FPA)-based blockchain framework for peer-to-peer energy trading with pe
Cited 43 times
Cited 44 times
Navigational path planning of multi-robot using honey bee mating optimization algorithm (
nvolution neural network for computer-aided polyps identification from colonoscopy
Cited 51 times
Physique-based human activity recognition using ensemble learning and
Cited 30 times
Two Stage Classification with CNN for Colorectal Cancer Detection.
nternational conference on innovative computing and communications
Cited 56 times
Cited 10 times
Cited 74 times
Social network analysis: theory and applications
Cited 18 times
Diagnosis Using Multiobjective Optimization with Evolutionary Parameter Optimization
Cited 13 times
Cited 60 times
Cited 35 times
Cited 7 times
Classification of non-small cell lung cancer types using sparse deep neura
Cited 16 times
heme for scheduling big data applications with SAVANNA Hadoop
Cited 37 times
A machine learning approach for the prediction of fetal healt
Cited 73 times
xt-aware fog enabled scheme for real-time cross-vertical IoT applications
A novel hybrid neural network-based multirobot path planning with motion
Cited 45 times
A novel blockchain-based healthcare system design and performance benchmarking on a multi-ho
Cited 15 times
Cited 24 times
A brief review of phasor measurement units as sensors for smart grid
Obscene image detection using transfer learning and feature fusion
Advanced vehicle security system
Cited 91 times
Evolution of wireless sensor network design from technology centric to user centric: an architectural perspective
Cited 52 times
Cited 36 times
Cited 70 times
Melanoma detection using advanced deep neural network
SBMYv3: improved MobYOLOv3 a BAM attention-based approach for obscene image and video detection
Image encryption and authentication with elliptic curve cryptography and multidimensional chaotic maps
Multirate data collection using mobile sink in wireless sensor networks
Cited 9 times
Software reliability allocation of digital relay for transmission line protection using a combined system hierarchy and fault tree approach
Cited 11 times
A new single-image super-resolution using efficient feature fusion and patch similarity in non-euclidean space
On the dynamics and feasibility of transferred inference for diagnosis of invasive ductal carcinoma: A perspective
Progress in computing, analytics and networking
Artificial immune system based path planning of mobile robot
Application of edge detection algorithm for design of lightweight robotic assembly system
for colorectal polyps analysis

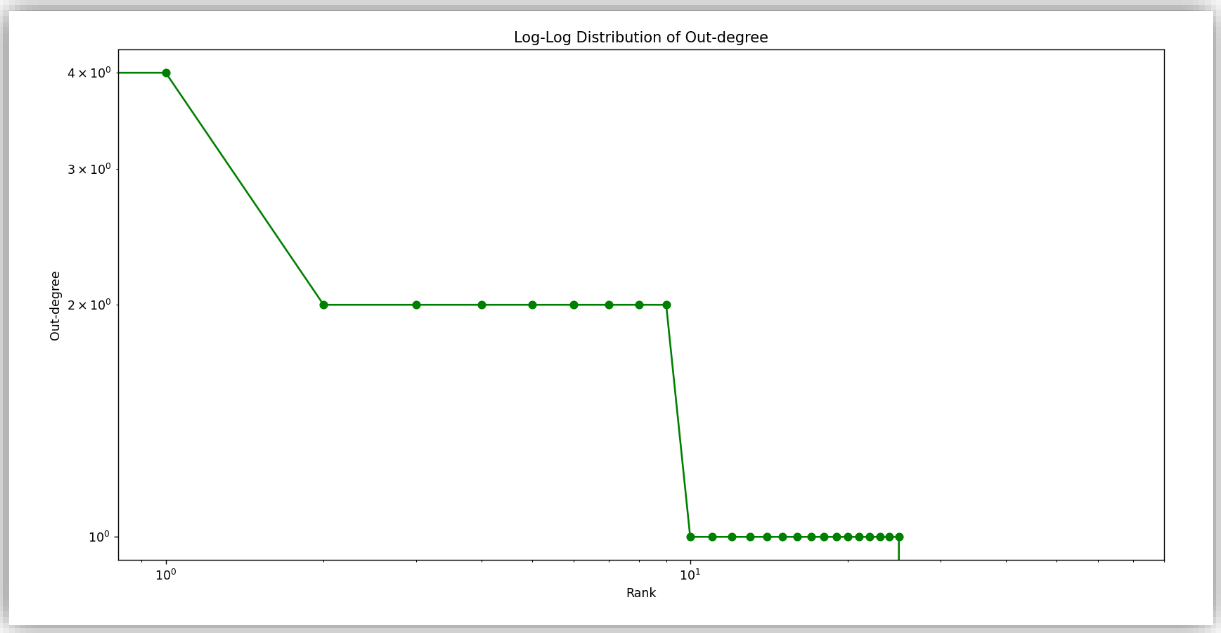


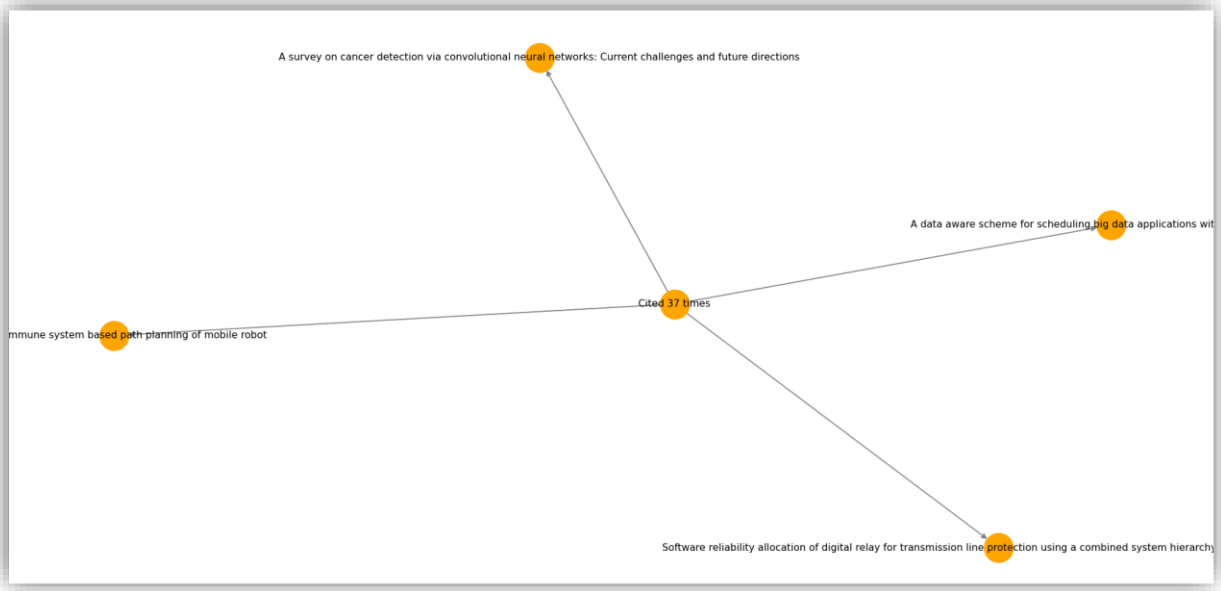


Out-degree Distribution
Frequency
40
35
30
25
20
15
10
5
0
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
Out-degree

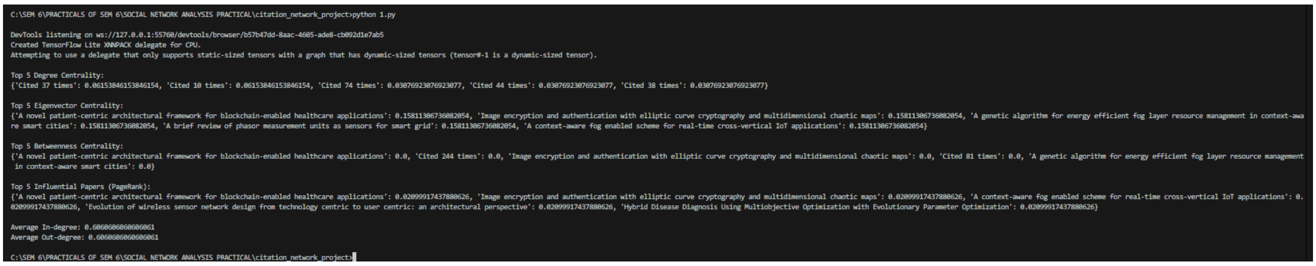


Log-Log Distribution of In-degree
In-degree
10^1
10^0
10^{-1}
10^0
10^1
Rank



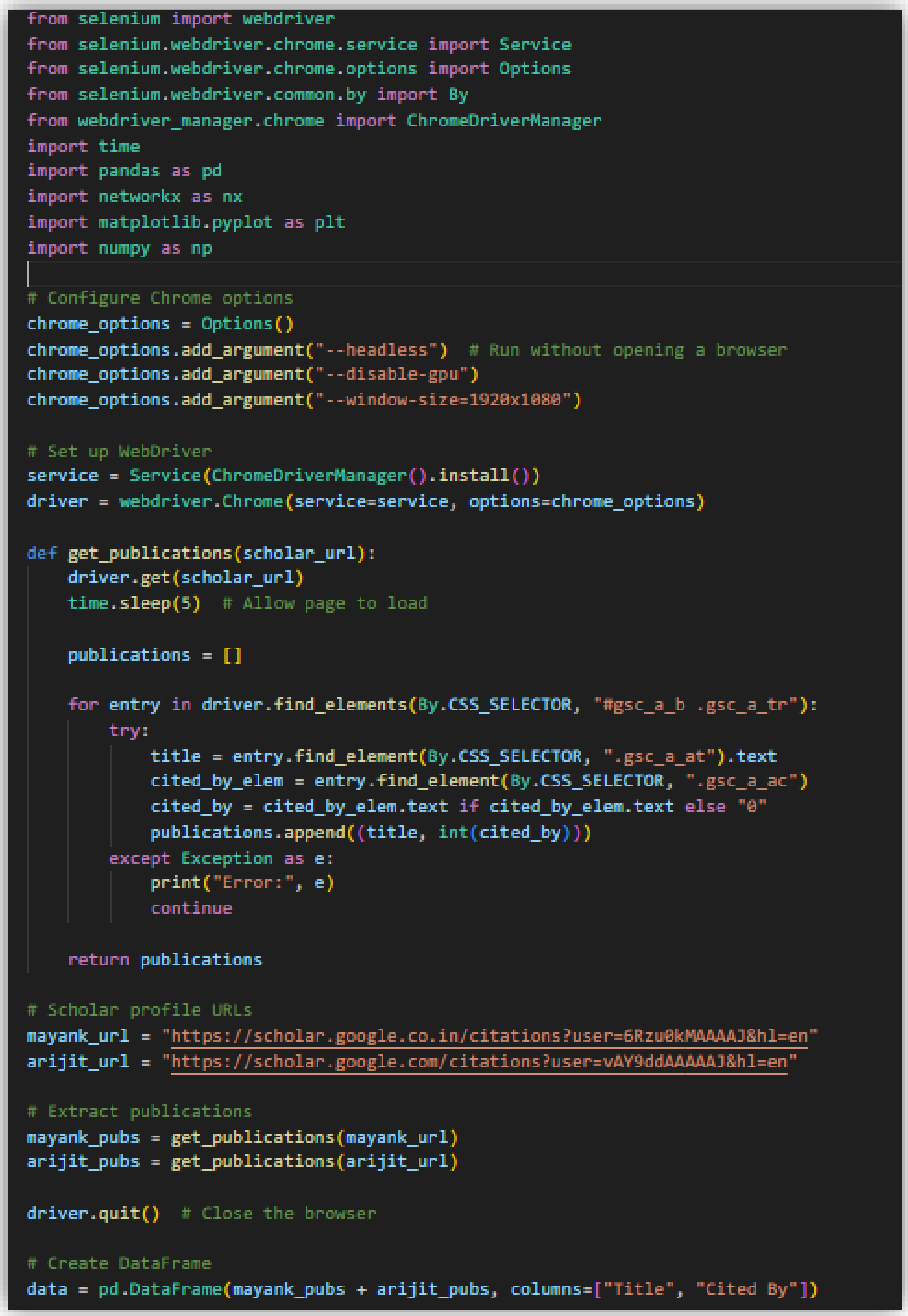


A survey on cancer detection via convolutional neural networks: Current challenges and future directions
A data aware scheme for scheduling big data applications wit
Cited 37 times
mmune system based path planning of mobile robot
Software reliability allocation of digital relay for transmission line protection using a combined system hierarchy

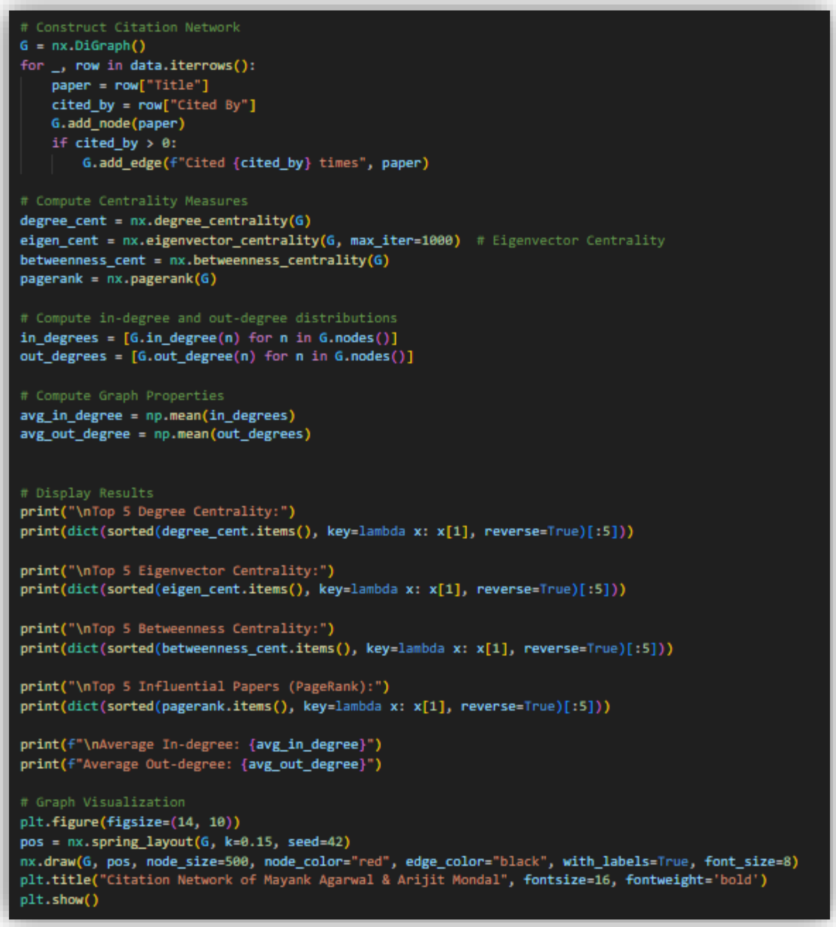


```
C:\SEM 6\PRACTICALS OF SEM 6\SOCIAL NETWORK ANALYSIS PRACTICAL\citation_network_project>python 1.py

DevTools listening on ws://127.0.0.1:55760/devtools/browser/b57b47dd-8aac-4605-ade8-cb092d1e7ab5
Created TensorFlow Lite XNNPACK delegate for CPU.
Attempting to use a delegate that only supports static-sized tensors with a graph that has dynamic-sized tensors (tensor#-1 is a dynamic-sized tensor).

Top 5 Degree Centrality:
{'Cited 37 times': 0.06153846153846154, 'Cited 10 times': 0.06153846153846154, 'Cited 74 times': 0.03076923076923077, 'Cited 44 times': 0.03076923076923077, 'Cited 38 times': 0.03076923076923077}

Top 5 Eigenvector Centrality:
{'A novel patient-centric architectural framework for blockchain-enabled healthcare applications': 0.15811306736082054, 'Image encryption and authentication with elliptic curve cryptography and multidimensional chaotic maps': 0.15811306736082054, 'A genetic algorithm for energy efficient fog layer resource management in context-awa
re smart cities': 0.15811306736082054, 'A brief review of phasor measurement units as sensors for smart grid': 0.15811306736082054, 'A context-aware fog enabled scheme for real-time cross-vertical IoT applications': 0.15811306736082054}

Top 5 Betweenness Centrality:
{'A novel patient-centric architectural framework for blockchain-enabled healthcare applications': 0.0, 'Cited 244 times': 0.0, 'Image encryption and authentication with elliptic curve cryptography and multidimensional chaotic maps': 0.0, 'Cited 81 times': 0.0, 'A genetic algorithm for energy efficient fog layer resource management
 in context-aware smart cities': 0.0}

Top 5 Influential Papers (PageRank):
{'A novel patient-centric architectural framework for blockchain-enabled healthcare applications': 0.02099917437880626, 'Image encryption and authentication with elliptic curve cryptography and multidimensional chaotic maps': 0.02099917437880626, 'A context-aware fog enabled scheme for real-time cross-vertical IoT applications': 0.
02099917437880626, 'Evolution of wireless sensor network design from technology centric to user centric: an architectural perspective': 0.02099917437880626, 'Hybrid Disease Diagnosis Using Multiobjective Optimization with Evolutionary Parameter Optimization': 0.02099917437880626}

Average In-degree: 0.6060606060606061
Average Out-degree: 0.6060606060606061

C:\SEM 6\PRACTICALS OF SEM 6\SOCIAL NETWORK ANALYSIS PRACTICAL\citation_network_project>
```

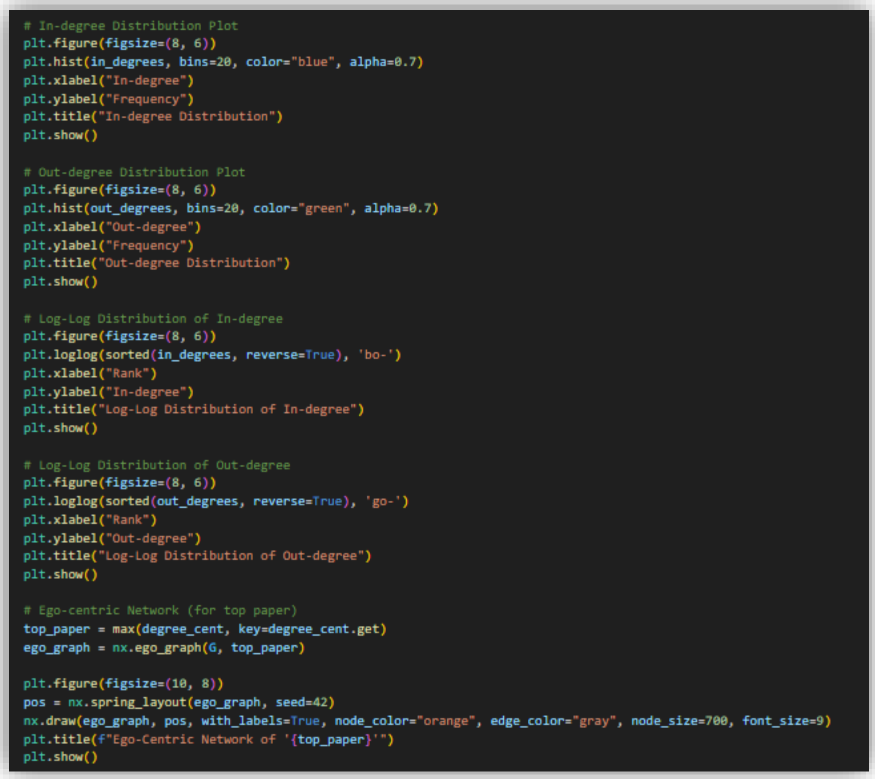
➢ *Code for IIT_PATNA:-*



```python
from selenium import webdriver
from selenium.webdriver.chrome.service import Service
from selenium.webdriver.chrome.options import Options
from selenium.webdriver.common.by import By
from webdriver_manager.chrome import ChromeDriverManager
import time
import pandas as pd
import networkx as nx
import matplotlib.pyplot as plt
import numpy as np

# Configure Chrome options
chrome_options = Options()
chrome_options.add_argument("--headless")  # Run without opening a browser
chrome_options.add_argument("--disable-gpu")
chrome_options.add_argument("--window-size=1920x1080")

# Set up WebDriver
service = Service(ChromeDriverManager().install())
driver = webdriver.Chrome(service=service, options=chrome_options)

def get_publications(scholar_url):
    driver.get(scholar_url)
    time.sleep(5)  # Allow page to load

    publications = []

    for entry in driver.find_elements(By.CSS_SELECTOR, "#gsc_a_b .gsc_a_tr"):
        try:
            title = entry.find_element(By.CSS_SELECTOR, ".gsc_a_at").text
            cited_by_elem = entry.find_element(By.CSS_SELECTOR, ".gsc_a_ac")
            cited_by = cited_by_elem.text if cited_by_elem.text else "0"
            publications.append((title, int(cited_by)))
        except Exception as e:
            print("Error:", e)
            continue

    return publications

# Scholar profile URLs
mayank_url = "https://scholar.google.co.in/citations?user=6Rzu0kMAAAAJ&hl=en"
arijit_url = "https://scholar.google.com/citations?user=vAY9ddAAAAAJ&hl=en"

# Extract publications
mayank_pubs = get_publications(mayank_url)
arijit_pubs = get_publications(arijit_url)

driver.quit()  # Close the browser

# Create DataFrame
data = pd.DataFrame(mayank_pubs + arijit_pubs, columns=["Title", "Cited By"])
```

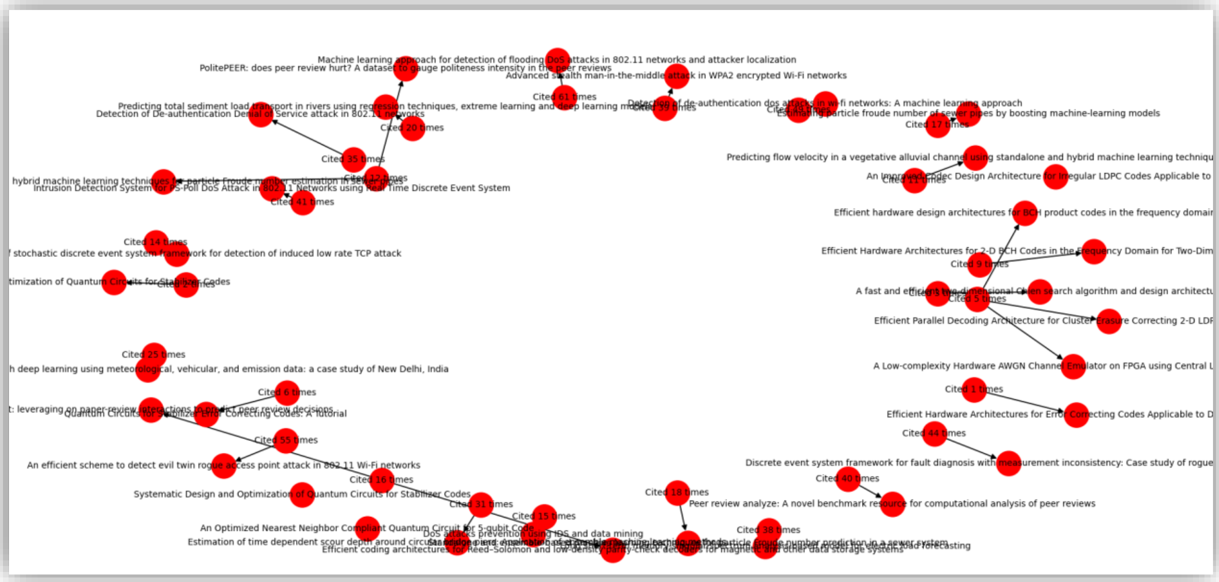


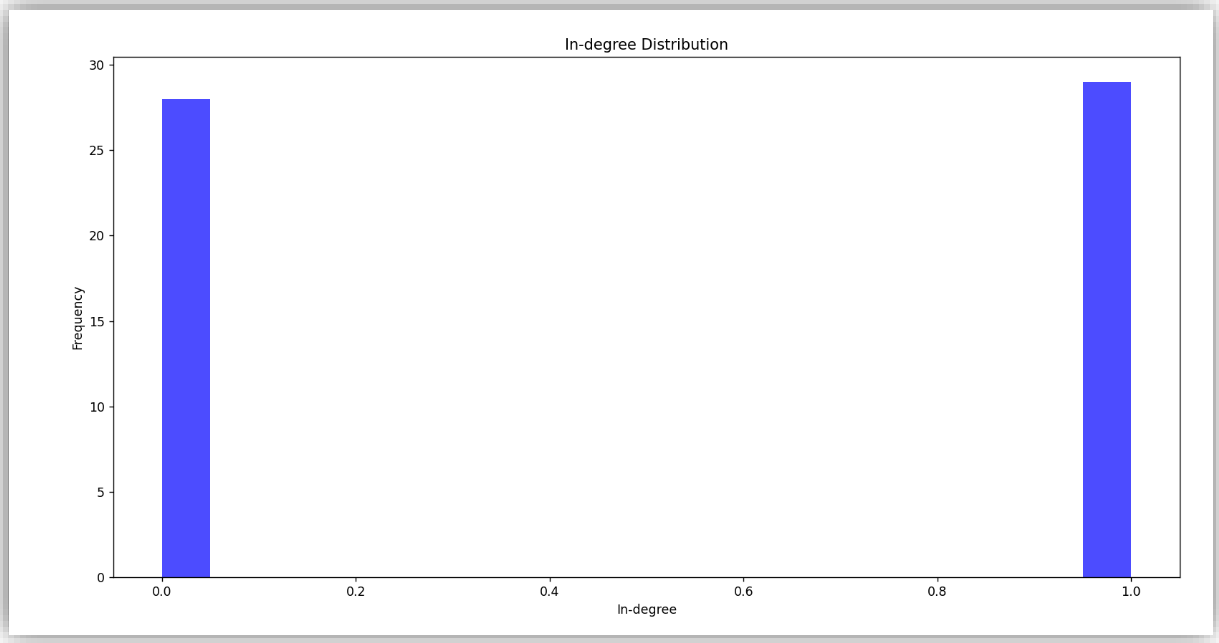
```python
# Construct Citation Network
G = nx.DiGraph()
for _, row in data.iterrows():
    paper = row["Title"]
    cited_by = row["Cited By"]
    G.add_node(paper)
    if cited_by > 0:
        G.add_edge(f"Cited {cited_by} times", paper)

# Compute Centrality Measures
degree_cent = nx.degree_centrality(G)
eigen_cent = nx.eigenvector_centrality(G, max_iter=1000)  # Eigenvector Centrality
betweenness_cent = nx.betweenness_centrality(G)
pagerank = nx.pagerank(G)

# Compute in-degree and out-degree distributions
in_degrees = [G.in_degree(n) for n in G.nodes()]
out_degrees = [G.out_degree(n) for n in G.nodes()]

# Compute Graph Properties
avg_in_degree = np.mean(in_degrees)
avg_out_degree = np.mean(out_degrees)


# Display Results
print("\nTop 5 Degree Centrality:")
print(dict(sorted(degree_cent.items(), key=lambda x: x[1], reverse=True)[:5]))

print("\nTop 5 Eigenvector Centrality:")
print(dict(sorted(eigen_cent.items(), key=lambda x: x[1], reverse=True)[:5]))

print("\nTop 5 Betweenness Centrality:")
print(dict(sorted(betweenness_cent.items(), key=lambda x: x[1], reverse=True)[:5]))

print("\nTop 5 Influential Papers (PageRank):")
print(dict(sorted(pagerank.items(), key=lambda x: x[1], reverse=True)[:5]))

print(f"\nAverage In-degree: {avg_in_degree}")
print(f"Average Out-degree: {avg_out_degree}")

# Graph Visualization
plt.figure(figsize=(14, 10))
pos = nx.spring_layout(G, k=0.15, seed=42)
nx.draw(G, pos, node_size=500, node_color="red", edge_color="black", with_labels=True, font_size=8)
plt.title("Citation Network of Mayank Agarwal & Arijit Mondal", fontsize=16, fontweight='bold')
plt.show()
```

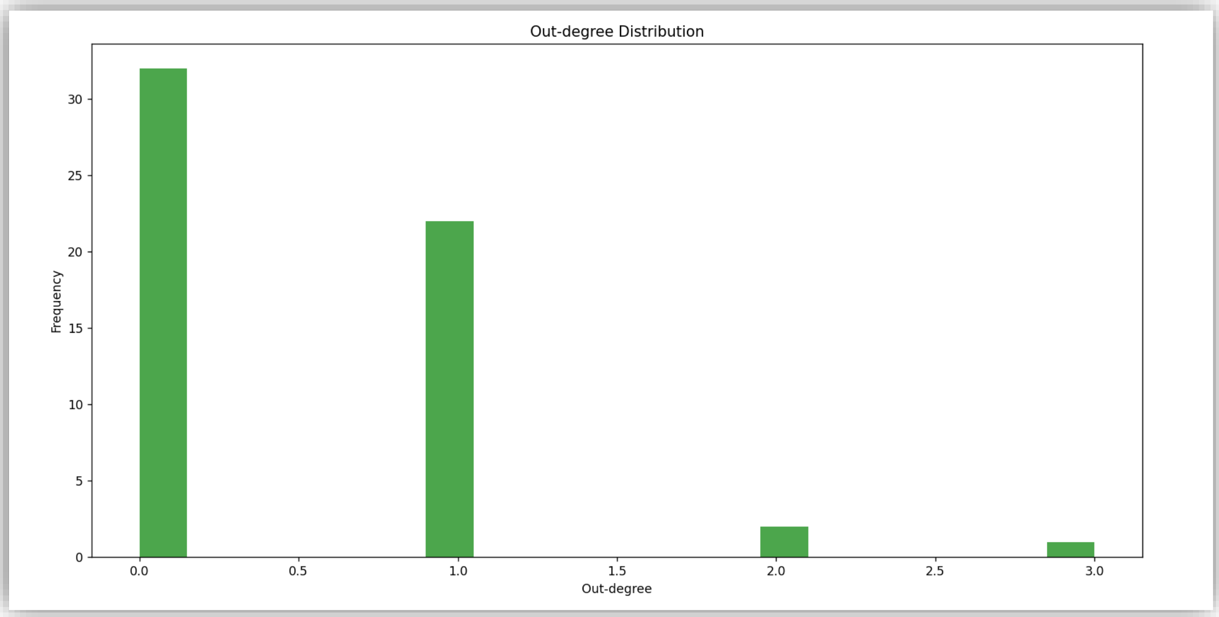


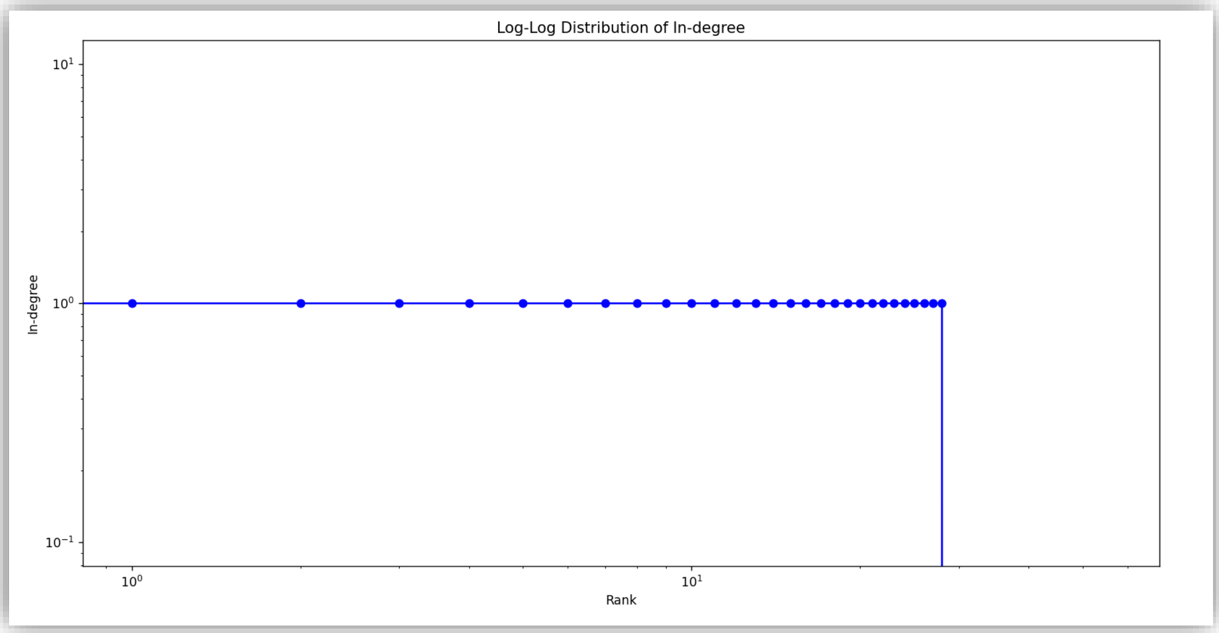
```python
# In-degree Distribution Plot
plt.figure(figsize=(8, 6))
plt.hist(in_degrees, bins=20, color="blue", alpha=0.7)
plt.xlabel("In-degree")
plt.ylabel("Frequency")
plt.title("In-degree Distribution")
plt.show()

# Out-degree Distribution Plot
plt.figure(figsize=(8, 6))
plt.hist(out_degrees, bins=20, color="green", alpha=0.7)
plt.xlabel("Out-degree")
plt.ylabel("Frequency")
plt.title("Out-degree Distribution")
plt.show()

# Log-Log Distribution of In-degree
plt.figure(figsize=(8, 6))
plt.loglog(sorted(in_degrees, reverse=True), 'bo-')
plt.xlabel("Rank")
plt.ylabel("In-degree")
plt.title("Log-Log Distribution of In-degree")
plt.show()

# Log-Log Distribution of Out-degree
plt.figure(figsize=(8, 6))
plt.loglog(sorted(out_degrees, reverse=True), 'go-')
plt.xlabel("Rank")
plt.ylabel("Out-degree")
plt.title("Log-Log Distribution of Out-degree")
plt.show()

# Ego-centric Network (for top paper)
top_paper = max(degree_cent, key=degree_cent.get)
ego_graph = nx.ego_graph(G, top_paper)

plt.figure(figsize=(10, 8))
pos = nx.spring_layout(ego_graph, seed=42)
nx.draw(ego_graph, pos, with_labels=True, node_color="orange", edge_color="gray", node_size=700, font_size=9)
plt.title(f"Ego-Centric Network of '{top_paper}'")
plt.show()
```

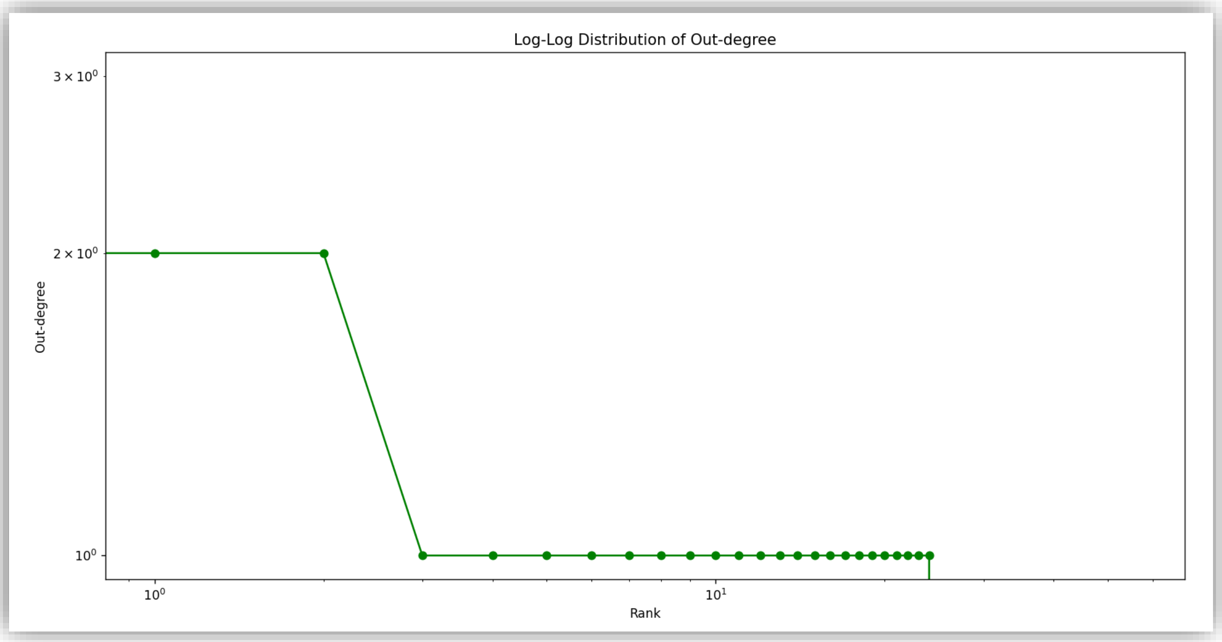
➢ ***Outputs for IIT_PATNA:-***

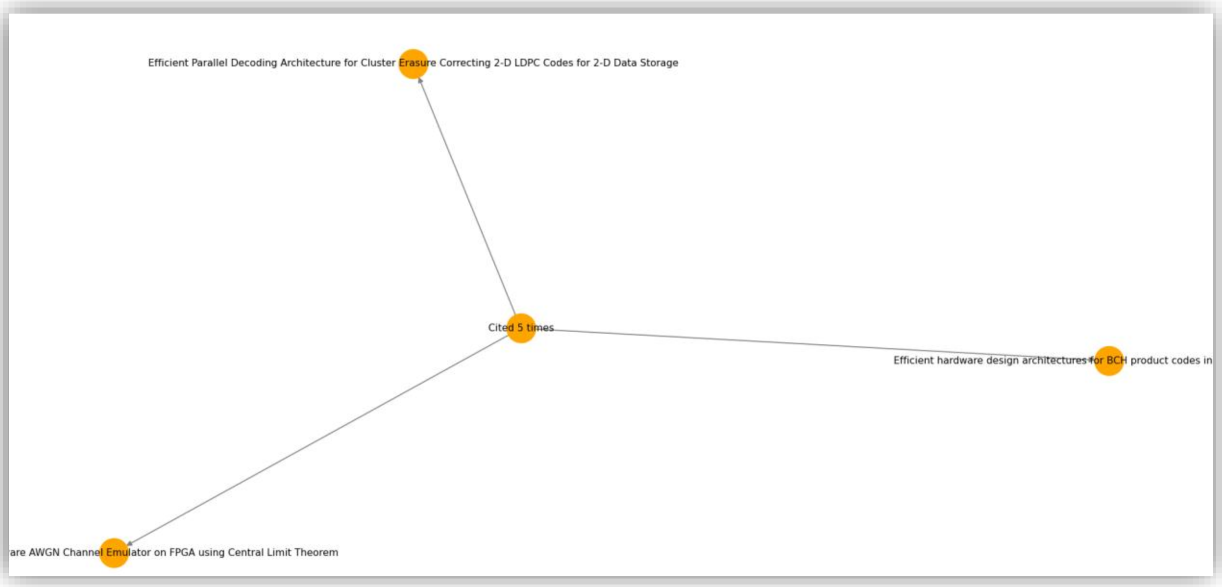

Machine learning approach for detection of flooding DoS attacks in 802.11 networks and attacker localization
PolitePEER: does peer review hurt? A dataset to gauge politeness intensity in the peer reviews
Advanced stealth man-in-the-middle attack in WPA2 encrypted Wi-Fi networks
Cited 61 times
Predicting total sediment load transport in rivers using regression techniques, extreme learning and deep learning models
Detection of De-authentication Denial of Service attack in 802.11 networks
Cited 20 times
Cited 39 times
Detection of de-authentication dos attacks in wi-fi networks: A machine learning approach
Cited 49 times
Estimating particle froude number of sewer pipes by boosting machine-learning models
Cited 17 times
Cited 35 times
Cited 12 times
hybrid machine learning techniques for particle Froude number estimation in sewer pipes
Intrusion Detection System for PS-Poll DoS Attack in 802.11 Networks using Real Time Discrete Event System
Cited 41 times
Predicting flow velocity in a vegetative alluvial channel using standalone and hybrid machine learning techniqu
An Improved Codec Design Architecture for Irregular LDPC Codes Applicable to
Cited 11 times
Efficient hardware design architectures for BCH product codes in the frequency domain
Cited 14 times
f stochastic discrete event system framework for detection of induced low rate TCP attack
Efficient Hardware Architectures for 2-D BCH Codes in the Frequency Domain for Two-Dim
Cited 9 times
timization of Quantum Circuits for Stabilizer Codes
Cited 2 times
A fast and efficient two-dimensional Chien search algorithm and design architectu
Cited 3 times
Cited 5 times
Efficient Parallel Decoding Architecture for Cluster Erasure Correcting 2-D LDP
Cited 25 times
h deep learning using meteorological, vehicular, and emission data: a case study of New Delhi, India
A Low-complexity Hardware AWGN Channel Emulator on FPGA using Central L
Cited 6 times
Cited 1 times
t: leveraging on paper-review interactions to predict peer review decisions
Quantum Circuits for Stabilizer Error Correcting Codes: A Tutorial
Efficient Hardware Architectures for Error Correcting Codes Applicable to D
Cited 55 times
Cited 44 times
An efficient scheme to detect evil twin rogue access point attack in 802.11 Wi-Fi networks
Discrete event system framework for fault diagnosis with measurement inconsistency: Case study of rogue
Cited 16 times
Cited 40 times
Systematic Design and Optimization of Quantum Circuits for Stabilizer Codes
Cited 18 times
Cited 31 times
Peer review analyze: A novel benchmark resource for computational analysis of peer reviews
Cited 15 times
An Optimized Nearest Neighbor Compliant Quantum Circuit for 5-qubit Code
Cited 38 times
DoS attacks prevention using IDS and data mining
Estimation of time dependent scour depth around circular bridge piers: Application of ensemble machine learning methods
particle Froude number prediction in a sewer system
forecasting
Efficient coding architectures for Reed–Solomon and low-density parity-check decoders for magnetic and other data storage systems

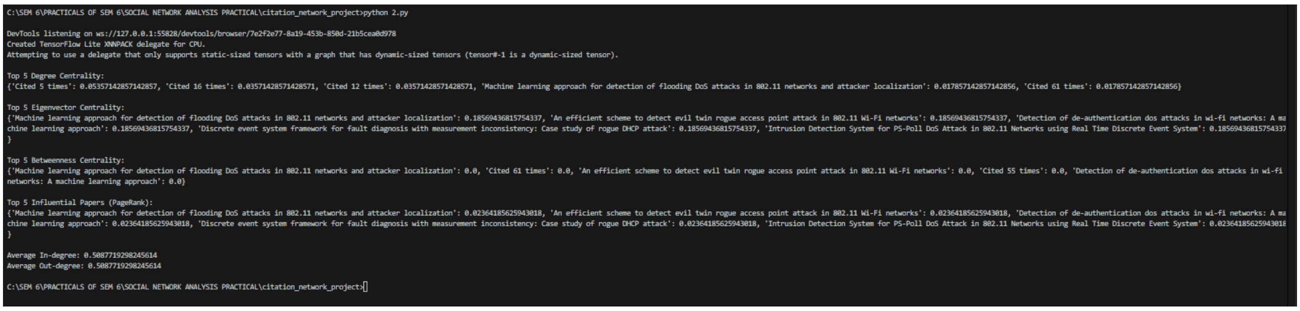



Out-degree Distribution
Frequency
0
5
10
15
20
25
30
0.0
0.5
1.0
1.5
2.0
2.5
3.0
Out-degree


Log-Log Distribution of In-degree
In-degree
10^{1}
10^{0}
10^{-1}
10^{0}
10^{1}
Rank




Efficient Parallel Decoding Architecture for Cluster Erasure Correcting 2-D LDPC Codes for 2-D Data Storage
Cited 5 times
Efficient hardware design architectures for BCH product codes in
are AWGN Channel Emulator on FPGA using Central Limit Theorem



```
C:\SEM 6\PRACTICALS OF SEM 6\SOCIAL NETWORK ANALYSIS PRACTICAL\citation_network_project>python 2.py

DevTools listening on ws://127.0.0.1:55828/devtools/browser/7e2f2e77-8a19-453b-850d-21b5cea0d978
Created TensorFlow Lite XNNPACK delegate for CPU.
Attempting to use a delegate that only supports static-sized tensors with a graph that has dynamic-sized tensors (tensor#-1 is a dynamic-sized tensor).

Top 5 Degree Centrality:
{'Cited 5 times': 0.05357142857142857, 'Cited 16 times': 0.03571428571428571, 'Cited 12 times': 0.03571428571428571, 'Machine learning approach for detection of flooding DoS attacks in 802.11 networks and attacker localization': 0.017857142857142856, 'Cited 61 times': 0.017857142857142856}

Top 5 Eigenvector Centrality:
{'Machine learning approach for detection of flooding DoS attacks in 802.11 networks and attacker localization': 0.18569436815754337, 'An efficient scheme to detect evil twin rogue access point attack in 802.11 Wi-Fi networks': 0.18569436815754337, 'Detection of de-authentication dos attacks in wi-fi networks: A ma
chine learning approach': 0.18569436815754337, 'Discrete event system framework for fault diagnosis with measurement inconsistency: Case study of rogue DHCP attack': 0.18569436815754337, 'Intrusion Detection System for PS-Poll DoS Attack in 802.11 Networks using Real Time Discrete Event System': 0.18569436815754337
}

Top 5 Betweenness Centrality:
{'Machine learning approach for detection of flooding DoS attacks in 802.11 networks and attacker localization': 0.0, 'Cited 61 times': 0.0, 'An efficient scheme to detect evil twin rogue access point attack in 802.11 Wi-Fi networks': 0.0, 'Cited 55 times': 0.0, 'Detection of de-authentication dos attacks in wi-fi
networks: A machine learning approach': 0.0}

Top 5 Influential Papers (PageRank):
{'Machine learning approach for detection of flooding DoS attacks in 802.11 networks and attacker localization': 0.02364185625943018, 'An efficient scheme to detect evil twin rogue access point attack in 802.11 Wi-Fi networks': 0.02364185625943018, 'Detection of de-authentication dos attacks in wi-fi networks: A ma
chine learning approach': 0.02364185625943018, 'Discrete event system framework for fault diagnosis with measurement inconsistency: Case study of rogue DHCP attack': 0.02364185625943018, 'Intrusion Detection System for PS-Poll DoS Attack in 802.11 Networks using Real Time Discrete Event System': 0.02364185625943018
}

Average In-degree: 0.5087719298245614
Average Out-degree: 0.5087719298245614

C:\SEM 6\PRACTICALS OF SEM 6\SOCIAL NETWORK ANALYSIS PRACTICAL\citation_network_project>
```

Citation_Network_SNA

The citation network analysis of **IIT Patna** and **NIT Meghalaya** focuses on understanding the academic influence of faculty members based on their publications and citation patterns. Using Selenium, we extracted publication data from Google Scholar for four professors: **Arjit Mondal** and **Mayank Agarwal** (IIT Patna), and **Diptendu Sinha Roy** and **Bunil Kumar Balabantaray** (NIT Meghalaya). The extracted data includes publication titles and the number of times each paper has been cited.

We constructed a directed citation network using **NetworkX**, where nodes represent research papers, and directed edges indicate citations. Several network measures were computed to evaluate the influence of different papers:

- ***Degree Centrality:*** Measures the importance of papers based on the number of connections (citations).
- ***Eigenvector Centrality:*** Identifies influential papers that are cited by other influential papers.
- ***Betweenness Centrality:*** Highlights key papers that act as bridges in the citation network.
- ***PageRank:*** Ranks papers based on their overall influence, considering both direct and indirect citations.
- ***In-degree and Out-degree Distribution:*** Analyzed how citations are distributed among papers.
- ***Ego-Centric Networks:*** Visualized networks centered around the most influential papers.

Through visualization techniques such as **degree distribution plots, log-log plots, and network graphs**, we identified highly cited papers and explored the structural properties of the network. The analysis provides insights into research impact and collaboration patterns within these institutions.