Problem Statement and Goals
Centrality In Graphs

Atiyeh Sayadi

January 20, 2024

Table 1: Revision History

Date	Developer(s)	Change
January 20, 2024	Atiyeh Sayadi	Initial Draft
Date2	Name(s)	Description of changes
...	...	...

1 Problem Statement

Calculating how important nodes are in graphs is crucial for understanding how networks work. Metrics like degree centrality, closeness centrality, and betweenness centrality help us find key nodes that are vital for information flow, connections, and influence in a network "Newman. (2013). Networks: An Introduction,.". This project aims to use the concept of centrality to calculate the main and most important nodes in the given graph.

1.1 Problem

In this project, we want to study a given graph using centrality measures, specifically degree centrality and closeness centrality. Centrality in a graph is like a way to figure out which nodes are really important. Degree centrality looks at how many connections each node has, and if a node has a lot of connections, it's considered more important. The formula for calculating Degree Centrality for a node i in a directed graph is as follows "Wasserman, S. and Faust, K. (1994). Social Network Analysis: Methods and Applications.":

C(i) = Degree of Node i / (Total number of nodes in the graph - 1);

C(i) : Degree Centrality of Node i,
Degree of Node i: The number of edges connected to Node i,

Total number of nodes in the graph: The overall count of nodes in the graph.

The result of Degree Centrality for each node lies between 0 and 1, where 1 indicates complete centrality (all edges are connected to the node) and 0 indicates no centrality (no edges are connected to the node). On the other hand, closeness centrality checks how close a node is to all the other nodes in the graph. It focuses on nodes that are central in terms of reaching other nodes efficiently. The equation for computing the Closeness Centrality of a node i in a directed graph is given by "Wasserman, S. and Faust, K. (1994). Social Network Analysis: Methods and Applications.":

C(i) = 1 / Average distance from node i to all other nodes;

C(i) : Closeness Centrality of node i,
Average distance from node i to all other nodes: The average distance between node i and all other nodes in the graph.

The result of Closeness Centrality for each node can range between 0 and 1, where 1 indicates complete closeness to all other nodes (minimum average distance), and 0 indicates a lack of closeness to other nodes.

Libraries are available that calculate closeness and degree centrality in graphs, incuding: NetworkX, igraph, Networkit, Gephi and Cytoscape. By visiting the documentation of each of these libraries, you will be able to obtain detailed information on how to use these libraries to compute closeness and degree centrality.
For this project, we start with a graph, and by using algorithms for degree centrality and closeness centrality, we find out how central each node is. Degree centrality highlights nodes with many connections, showing their importance in terms of direct connections. Closeness centrality identifies nodes that are central in terms of efficient communication with other nodes. The project's output provides these centrality measures for each node, helping users understand the importance and connections of each node in the given graph. The main goal is to give valuable information about the structure of the network and pinpoint the most influential nodes.

1.2   Inputs and Outputs

Input:

A given graph Centrality metrics, specifically degree centrality and closeness centrality algorithms

Output:

Centrality measures for each node in the given graph.

For degree centrality: Highlight nodes with a large number of connections, indicating their significance in terms of direct connections.

For closeness centrality: Identify nodes that are central in terms of efficient communication with other nodes.

The output presents these centrality metrics for each node, enabling users to understand the importance and connectivity of individual nodes within the given graph. The primary focus is on providing valuable insights into the network structure and identifying the most influential nodes.

1.3   Environment

This project will be developed using Python.

2   Goals

The project has several goals. The main focus is on implementing centrality measures, specifically degree centrality and closeness centrality algorithms, to calculate the importance of each node in a given graph. Degree centrality highlights nodes with many connections, indicating their importance, while closeness centrality identifies nodes crucial for efficient communication. The project output presents these measures for each node, helping users understand the significance and connections of individual nodes in the graph. The primary aim is to provide insights into the network structure and identify influential nodes.

3   Stretch Goals

Implementation of Broader Algorithms: Adding additional centrality algorithms beyond degree and closeness, such as betweenness centrality or eigenvector centrality.

Enhanced User Interface: Developing a more efficient and feature-rich user interface for graph analysis.

Support for Larger Graphs: Improving program performance to process and analyze larger graphs.

Advancement in Pattern Recognition: Implementing algorithms for identifying specific patterns within the graph.

Support for Diverse Graphs: Adding support for directed or multipartite graphs.

Enhanced Reporting Capabilities: Increasing reporting capabilities for analyses and project results.