Programming Project 2018

CITS2200 – Data Structures and Algorithms

Bruce How (22242664), Haolin Wu (21706137)

i. Introduction

Centrality is an important measure of the global influence of a vertex in a graph. This measure is used extensively in large social network graphs, often for information diffusion and marketing. Many different measures of centrality exist. The four explored in our project are Degree, Closeness, Betweenness and Katz.

Figure 1: UML diagram of overall project

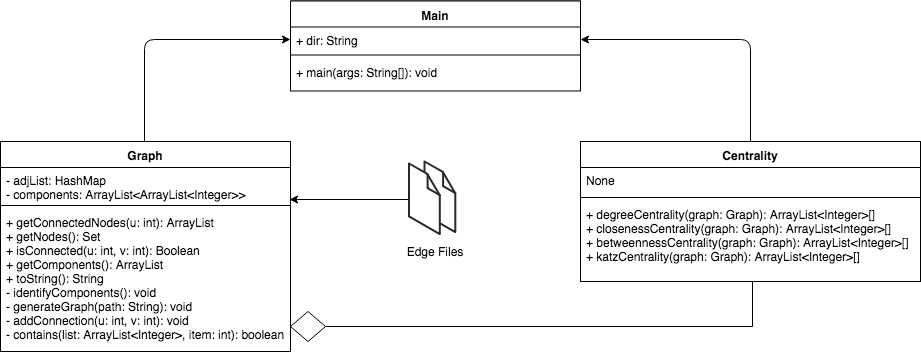


Figure 1 depicts a general UML diagram of our entire process. main classes are usually not shown, however for the purpose of clarity, it is represented as a class in the diagram. The entire project is based off three classes, graph, centrality and main.

­­­

ii. Algorithms

Breadth First Search

In our project, the Breadth First Search (BFS) algorithm is used heavily in both classes. It is used in the identifyComponents() method in the Graph class, and the Closeness and Katz centralities in the Centrality class.

A BFS begins at an arbitrary node in a graph, and explores neighbor nodes first, before moving to the next level. BFS uses a queue, as queue is First In First Out (FIFO) and checks whether a vertex has been visited, and if it hasn’t, it enqueues it to the queue.

The reason why BFS is the primary algorithm chosen is due to the unweighted and undirected nature of the Graphs presented by the project.

Brandes Algorithm

For node , Brandes Algorithm requires the shortest path from s to every other node . These paths are stored for each pair s,t and is achieved by performing a Breadth First Search.

In Brandes Algorithm, for a certain node v, the ratio of shortest paths between s and t that go through v and the total number of shortest paths between s and t is called the pair-wise dependency:

Therefore:

Brandes Algorithm for a non-tree case uses an algorithm dubbed ‘Ultimate MAGIC’. When there is alternative shortest paths that bypass v, the situation becomes more complex. A proportion of these shortest paths to nodes go through v, but a proportion doesn’t. Ultimate MAGIC determines this ratio using a mathematical algorithm that is further explained in Appendix 1. In terms of the code, the dependency of each source node of the BFS is different, and the betweenness centrality of the node is calculated from the summation of all dependency values, also known as the dependency accumulation. The dependency is described as:

Centrality Implementation:

(Project Deliverable 2)

Complexity:

Execution:

(Project Deliverable 1 – main())

Conclusion: