

CS225 Final Project Results

Madhav Aggarwal , Charles Xu , Elia Chudov , and Matt Geimer

Outcome

We implemented the following things in various folders/files:

- Graph
 - Edge Class: The edge class contains data regarding the vertices it connects, as well as the weight of the edge.
 - Graph Class: The graph class is our implementation of a graph in C++. It has all the major functions we needed for our algorithms, as well as some other creature-comfort functions.
 - Parsing: The parsing namespace is a set of functions that allows us to parse the data from a text file into a useful graph
- Traversals
 - GraphTraversal: Abstract parent class for graph traversals. Used to implement an iterator for child traversal classes
 - BFS: Breadth first search that implements GraphTraversal.
 - DFS: Depth first search that implements GraphTraversal.
- Algorithms
 - FloydWarshall: Finds the shortest distance (and therefore shortest path) from every vertex to every other vertex.
 - Dijkstra: Finds the shortest path between 2 vertices
 - Betweenness Centrality: Returns the most central vertex in the given graph. That is, it returns the vertex in the graph that is most commonly found on the shortest path between every vertex and every other vertex.

Discoveries

We used the betweenness centrality algorithm to determine that Zurich International Airport was the most central airport in the openflights dataset we used.