Graphs

Alex Sverdlov
alex@theparticle.com

1 Introduction

Many problems can be modeled as graphs. A graph is a tuple, (V, E), where V are vertices (nodes), and E are edges (links between nodes). Graphs can be directed (E has a direction) or not. Some graphs may also have weights (some metric) associated with each edge. There might also be a type for node, or an edge.

1.1 Customers & Products

Suppose we are modeling a retail store. The objects we are concerned with are customers, products, and purchases. In this scenario, customers and products may be vertices, and purchases are edges between customer vertices and product vertices. The weight of the edge may be the number of times a customer purchases a particular product.

1.2 Social Graph

Similarly if we are modeling a social network, vertices may be individuals, and edges may represent some individual-to-individual relationship. Follows relationship could be directed, while friends relationship bidirectional. Individuals could also be different type, such as corporations, or real people, etc.

1.3 World Wide Web

The World Wide Web already implies a graph in its name. Vertices are generally pages, and links are directed edges.

2 Common Things

There are a few common things folks do with graphs. Below is just a summary.

2.1 Paths

Is there a path between two vertices? Path finding is often accomplished by breadth first search from the starting vertex, until the destination vertex is found.

2.2 Shortest Path

Of all the paths, which one has the lowest weight?

One way to solve this is to do a breadth-first search to find all the paths, and then pick the minimum.

Instead of find all paths and then eliminating all the long ones, Dijkstra's algorithm maintains the shortest distance from start to every explored node—eliminating paths that are longer than the shortest already known.

2.3 Spanning Trees

Spanning Tree starts at a vertex, and builds a path to every other vertex, without cycles. It's essentially a graph traversal. The most common ones are breadth first and depth first.

The traversal method starts at a vertex, and selectively explores edges leading from the selected vertex.

The result of a traversal is a spanning tree. It's a tree (no cycles) that includes all vertices, with root being the starting vertex.

If the edges have a weight, then a minimum spanning tree is the smallest total-weight spanning tree.

2.4 Connected Components

A spanning tree is a connected component. If the entire graph is covered by a spanning tree, then it's a connected graph.

If we run a spanning tree algorithm on a graph, and there are left-over vertices, then we have more than one connected component.

Finding all connected components is often accomplished by running the spanning tree algorithm repeatedly on left-over vertices until none is left.

2.5 Bridges & Articulation Points

In a connected graph, a bridge is an edge, that if removed, creates a disconnected component.

Articulation points are vertices, that if removed, create a disconnected component.

Finding bridges & articulation points is critical to avoiding single-point-of-failure in a lot of systems.

2.6 Clique & Maximal Clique

A clique is a graph in which every vertex is connected to every other vertex via an edge (each vertex is adjacent to every other vertex).

Often we wish to find clique sub-graphs in a bigger graph.

A maximal clique is the biggest sub-graph that is a clique (cannot be made into a bigger clique by adding an adjacent vertex).

2.7 Density

Graph density is a number between 0 and 1 measuring the number of edges, to the total possible edges. A graph (or subgraph) with density 1 is a clique.

3 Recommendation Engines

One way to think of recommendations is by measuring similarity. Similar customers will wish to purchase similar products. If customers and products are vertices, and purchases are edges, then one way to articulate similarity is by comparing customer to customer based on the products they've purchased.

4 Page Rank

The basic idea is random surfer who clicks on links from page to page. The pages that the random surfer visits most often are obviously more "important": that importance score is the "Page Rank" of that page. Page rank of a page is the sum of page ranks of all pages that link to it.

Additions to make this simple idea work: If a page has many outgoing links, then its page rank should be divided equally between each outgoing links.

Some pages may not have any outgoing links. For example, image files. Once the random surfer goes to one of these, they have no way if clicking out. At this point, the random surfer has the option of jumping to any page randomly.

At some point, the random surfer may get tired of following links, and just enter a new address (essentially jumping to another random page).

5 TODO: more in class