Graph Data Model

Created	@February 19, 2025 9:23 AM
Class	DS 4300
Reviewed	☐

Graph Database

- Data model based on graph data structure
- Composed of nodes and edges
  - Each node and edge is uniquely identifiable
- Queries are based on graph-oriented operations
  - Not similar to SQL

Graph Theory

- Labeled Property Graph
  - Composed of set of node (vertex) and relationship (edge)
  - Labels are used to mark a node as part of a group
  - Properties are attributes (think KV pairs) and can exist on nodes and edges
  - Nodes with no relationships are OK, edges with no nodes are not ok

Paths

- a path is an ordered sequence of nodes connected by edges in which no nodes or edges are repeated
- There are n choose 2 pairs of nodes

Types of Graphs

- Connected vs. Disconnected: split into Components or SCC
- Weighted vs. Unweighted: edges have weight
- Directed vs. Undirected: edges have directions
- Acyclic vs. Cyclic: graph contains cycles
- Sparse vs. Dense vs. Complete:
  - sparse: one relationship between one node and another
  - dense: multiple relationships between one node and others
  - complete (clique): every node connected to every other

Trees

- Rooted Tree: root node, acyclic
- Binary Tree: up to 2 child nodes and no cycles
- Spanning Tree: subgraph of all nodes but not all relationships and no cycles

Graph Algorithms

Pathfinding

- finding shortest path between two nodes, if one exists, is probably the most common operation
- Shortest means fewest edges or lowest weight
- Average shortest path can be used to monitor efficiency and resiliency of networks
- MST, cycle detection, max/min flow
- Breadth First Search (BFS): visits nearest neighbor first
- Depth First Search (DFS): walks down each branch first
- Shortest Path: shortest path between 2 nodes
- All-Pairs Shortest Path: all shortest paths between all pairs of nodes
- Single Source Shortest Path: shortest path from a root node to all others

- Minimum Spanning Tree: shortest path connecting all nodes

Centrality & Community Detection

- Centrality: determining which nodes are “more important” in a network compared to other nodes
    - Degree: number of connections of a node
    - Closeness: which node can most easily reach all other nodes in a graph or subgraph
    - Betweenness: which node has the most control over flow between nodes and groups
    - Page Rank: which nodes is most important?
    - Examples:
        - Dijkstra’s - single source shortest path algo for positively weighted graphs
        - A* Algorithm - Similar to Dijkstra’s with added feature of using a heuristic to guide traversal
        - Page Rank - measures the importance of each node within a graph based on the number of incoming relationships and the importance of the nodes from those incoming relationships
- Community Detection: evaluate clustering or partitioning nodes of a graph and tendency to strengthen or break apart

Neo4j

- Graph Database system that supports both transactional and analytical processing of graph-based data
- Relatively new class of no-sql DBs
- Considered schema optional (one can be imposed)
- Supports various types of indexing
- ACID compliant
- Supports distributed computing