NoSQL Graph Centrality Algorithms

Concepts

Graph Centrality Algorithms

- Degree centrality
- Closeness centrality
  - Wasserman and Faust
  - Harmonic centrality
- Betweenness centrality
  - RA-Brandes (Randomized-Approximate Brandes)
- PageRank
  - Personalized PageRank

Degree Centrality

- Measures the number of relationships a node has in the graph
  - Incoming
  - Outgoing
- Compare a node to statistics for all nodes of the graph or subgraph
  - Average, median, minimum, maximum, standard deviation

Closeness Centrality

- Measures the average of the shortest path distances between a node and all other nodes
- Node with high closeness
  - Have the shortest distances to other nodes
  - Able to spread information most efficiently
- Compare a node to statistics for all nodes of the graph
  - Average, median, minimum, maximum, standard deviation

Wasserman and Faust

- Standard algorithm would see distances to nodes in disconnected subgraphs as infinity—skews calculations
- Improved closeness algorithm for graphs with disconnected subgraphs
  - Find the closeness number for a node considering reachable nodes
  - Multiply this by the percentage of reachable nodes to the number of nodes

Harmonic Centrality

- Another approach to improving closeness algorithms for graphs with disconnected subgraphs
- In the closeness algorithm, instead of summing distances, sum the inverses
- Unreachable nodes
  - Closeness number is infinity
  - Inverse of infinity is zero
  - Effect is that unreachable nodes are ignored

Betweenness Centrality

- Find all pairs' shortest paths (weighted)
- For each node, how many paths pass through the node?
- High betweenness
  - High number of paths passing through the node
  - Control point
  - Bridge
  - More influence over the flow within the graph
- Pivotal node: lies on every path between two other nodes

Betweenness Centrality of Clusters

- Group node into clusters
- Each cluster becomes a single node in a new graph
- Calculate betweenness of clusters using the new graph
- Can repeat for several layers of hierarchy
- Scale-free networks

RA-Brandes: Randomized-Approximate Brandes

- Calculating centrality on large graphs can be time-consuming and expensive
- Method to approximate betweenness centrality
- Random subset of nodes
- Two ways to choose
  1. Choose random nodes uniformly
  2. Choose a random node and throw it out if degree is less than average
- Can also limit depth of shortest path algorithm

PageRank, Part I

- Larry Page of Google
- Overall influence of a node in a graph
  - Direct influence of a node
  - Influence of incoming relationships
  - Influence of incoming relationships of the incoming relationships
  - So forth
- Knowing a lot of influential people makes you more influential

PageRank, Part II

- Relationships
  - Directional
  - Weighted
  - Incoming relationships increase a node’s influence score
- Algorithm
  - Score each node by weighted incoming relationships
  - Iterate: each pass passes scores along to the outgoing relationships
  - Stop when scores converge or when a predetermined number of iterations has been reached

PageRank, Part III

- Issues
  - Random surfers
    - Basic algorithms assumes surfers are following links
    - Surfer does not follow links, moves on to something else
    - Solution: damping factor
  - Rank sinks
    - No outbound relationships
    - Solution: random teleporting to another node

Personalized PageRank

- Perspective from a single node
- What’s important to a specific user
- Target recommendations to a specific user

The End

NoSQL Graph Centrality Algorithms

Business Cases

Social Media Ranking

- Rank social media personalities by how many followers they have
- Solution
  - Degree centrality to rank them
  - Compare to statistics on degree centrality

Detecting Fraud on Online Auction

- Fake accounts which bid to get prices higher
- Solution
  - Weighted relationships for successful vs. unsuccessful bids
  - High degree centrality without success can detect an account that bids on lots of items and never wins

Terrorist Cells

- Identify members of terrorist cells who can quickly acquire and distribute information to members
- Solution
  - Closeness centrality based on who communicates with whom

Infection Spread

- A hospital worker has been infected with a dangerous virus
- Trace the possible spread
- Who should we test first?
- Solution
  - Closeness centrality based on employees and patients they have been in contact with

Computer Network

- Identity routers that are crucial for connecting major portions of the network
- Single points of failure
- Solution
  - High betweenness centrality to identity routers
  - Add some redundancy

Interdisciplinary Employees

- Identity key employees with interdisciplinary skill
- Solution
  - High betweenness centrality for groups they meet with

PageRank Obvious Uses

- Indexing the World Wide Web
- Recommend to follow (Personalized PageRank)
  - Websites
  - Blogs
  - Social media personalities
  - Podcasts
  - Etc.

Less Obvious PageRank Uses

- Anomaly detection
  - Medical insurance claims fraud
  - Computer network traffic from hacking
- Predicting traffic flow based on usual patterns
  - Vehicles
  - People
  - Bicycles

The End

NoSQL Graph Community Detection Algorithms

Concepts

Community Detection Algorithms

- Triangle count and clustering coefficient
- SCC (strongly connected components)/connected components
- LPA (label propagation algorithm)
- Louvain modularity

Triangle Count and Clustering Coefficient

- Triangle count
  - Number of triangles that pass through a node
- Clustering coefficient
  - Node A is connected to B
  - Node A is connected to C
  - Probability that B is connected to C
  - Probability that neighbors of a node are connected to each other
  - One means full clique—every node connected to every other node

SCC: Strongly Connected Components/ Connected Components

- SCC (strongly connected components)
  - Group of nodes
  - Each node is reachable from every other node in group
  - Must use direction
- Connected components
  - Direction not considered

LPA: Label Propagation Algorithm

- Fast
- Networks where grouping is less clear
- Nodes which have labels pass those labels to neighbors
- If the neighbor gets multiple labels
  - Chooses label with highest presence in neighborhood
  - Node weights
  - Relationship weights

LPA Push vs. LPA Pull

- Push labels to other nodes
  - Unweighted
  - Less commonly used
  - Serial
- Pull labels from other nodes
  - Weighted
  - More commonly used
  - Parallel

Louvain Modularity

- What if analysis—tries different groupings
- Modularity
  - How well a node is assigned to a group
  - Compare relationship weights and densities to an average or estimate
- Creates hierarchy of groups at different scales

Modularity Algorithm Issues

- Tend to merge smaller groups into larger groups
- Brick wall
  - Several options have the same modularity
  - Local maxima
  - Cannot proceed

The End

NoSQL Graph Community Detection Algorithms

Business Cases

Determine if a Network Is Small World

- We want to determine if a network is small world
- Solution
  - High triangle counts and high clustering coefficients are present in small world networks

Predict Group Stability

- We want to determine how stable a group is in a network
- Solution
  - Compare triangle counts and clustering coefficients for the group and compare it to statistics for the overall graph
  - Above-average connectivity in a group tends to make it more stable

Buddy Deals

- We want to find companies where the executives of each company own shares in the other companies
- May indicate buddy deals not in the interest of the shareholders
- Solution
  - SCC (strongly connected components) based on stock ownership disclosures

Recommendations for New Customers

- We have new customers without a lot of connectivity
- We want to make recommendations to them
- Solution
  - SCC (strongly connected components) will tell us how connected the groups they join are as they join them, and we can recommend based on those groups

Quick Rough Cut of Groups

- Most graph community detection algorithms are intensive
- We need a very quick rough cut of groups
- Solution
  - Connect components runs quickly and can give us a quick rough cut of groups

Sentiment Analytics

- We want to categorize social media posts and messages as positive or negative
- We run AI, ML, or DL to categorize as positive or negative
- AI, ML, or DL cannot categorize all of them
- Solution
  - Use LPA (label propagation algorithm) to categorize based on relationships with the ones that were categorized

Drug Interactions

- People may be taking two or more drugs which could be causing interactions
- Label based on drug, drug family, chemical, side effects, etc.
- Solution
  - Use LPA (label propagation algorithm) to identify possible side effects caused by drug interactions

Cyber Attacks

- Network traffic is naturally hierarchical with groups at different scales
- Establish a normal pattern
- Determine what does not fit the normal pattern
- Solution
  - Use Louvain modularity, which creates hierarchy with groups at different scales

The End

NoSQL Graph Feature Engineering for Artificial Intelligence, Machine Learning, Deep Learning

Concepts

Feature Engineering

- Features: columns in a dataset
- Feature engineering
  - Feature extraction
    - Datasets often have a large number of features
    - Deciding which features in a dataset to include
    - Weighting included features
    - Adding features from secondary datasets
  - Feature creation
    - Creating new features based on manipulation of existing features
    - Graphs can be used to create new features

Graphs and Feature Engineering

- Load datasets into a graph database
- Create features by extracting data from the graph database and/or algorithms run on the graph database
- Typically secondary features to enhance the primary feature

Graphy Features

- Connection related statistics at the node level
- Number of relationships
  - Inbound
  - Outbound
- Number of triangles
- Etc.

Graph Algorithm Features

- Features created from running graph algorithms
- Path algorithms
  - Example: how many hops between two nodes
- Centrality algorithms
  - Example: find the control points
- Community detection algorithms
  - Example: find a hierarchy of groups

Model Evaluation Using Graphs

- Often run numerous models and hyperparameters for AI, ML, DL
- Evaluate the model runs to see which one performs best
- Load output data from the model runs into graphs
- Gather graph statistics
- Run graph algorithms
- Helps us to decide which model performs best

The End

NoSQL Graph Feature Engineering for Artificial Intelligence, Machine Learning, Deep Learning

Business Cases

Organized Crime

- Dataset with features that can predict fraud
- Load our dataset into a graph database
- Extract features from the graph such as:
  - Find hierarchy of groups
  - Find influential nodes
  - Find control points where money or information are flowing
  - Find other suspects that are in close proximity
- Use these features to enhance AI, ML, DL algorithms

Hurricane

- Hurricane wipes out an entire metroplex’s power and communications lines
- Cell phone towers are destroyed
- We need to set up emergency portable cell phone towers to quickly re-establish communications

AI Model Predicts

- AI model predicts the optimal location for cell towers
- AI model has a lot of hyperparameters that can be tweaked
- Run AI model with all possible combinations of hyperparameters
- For each model run, load the solution into a graph
- Use graph statistics and graph algorithms to find the best solution

Cell Tower Graph Evaluation

- Towers that will be overloaded with connections
  - Nodes with high degree
- Towers that are bridges will have a high impact if they fail
  - Nodes with high betweenness
- Towers should take advantage of communication lines out of the city that are still working
  - Nodes with working communication lines should have a high closeness centrality

Cell Tower Graph Evaluation (cont.)

- Towers should have a good hierarchical network where local calls do not travel far
  - Louvain modularity to see how good an overall hierarchical network the solution has

The End