Ashley Music

Week 6 – Community Detection Algorithm

6/9/20

BDAT 635

Provide pros and cons of each of these Community Detection Algorithms, examples where these algorithms can be applied. What are the expected outcome/insights from each of these algorithms?

Louvain

Pros: This algorithm is used to detect communities on large data sets or networks, and it is very fast in processing.

Cons: Because it runs on large networks it requires a lot of memory processing on your computer.

This algorithm is used when you want to find high modularity in communities. Modularity is used to determine the quality of the community. High modularity means that the community is densely clustered within those nodes, but not densely clustered when viewed in different modules. This will help you detect smaller, densely connected communities within a larger networks. From Neo4j’s blog [Financial Fraud Detection with Graph Data Science: Identifying Fraud Rings](https://neo4j.com/blog/financial-fraud-detection-graph-data-science-identifying-fraud-rings/), we are told that using the Louvain algorithm can help detect fraud rings by determining if “hierarchies exist among these communities” which can help narrow down on people who are leading the fraud rings.

Label Propagation

Pros: Does not need any prior information on the community because it uses the network structure.

Cons: Because it runs on large networks it requires a lot of memory processing on your computer. This algorithm can result in “different community structures when run multiple times on the same graph” according to the blog, [Graph Algorithms in Neo4j: Label Propagation](https://neo4j.com/blog/graph-algorithms-neo4j-label-propagation/)

The algorithm will apply labels to each node by updating “its label to the one that the maximum number of its neighbors belong to.” Label propagation can be used for things like assigning sematic analysis (negative and positive emotions) of tweets on social media. In the medical profession it has been used to determine drug combinations that could be dangerous.

Weakly Connected Components

Pros: Used to help determine the structure of a graph and it can be used in an undirected graph.

Cons: Because it runs on large networks it requires a lot of memory processing on your computer. It is a fairly “basic” algorithm and it often just used as a starting point in your analysis.

This is a very useful algorithm if you are looking to verify the or find the structure of your network. From the Neo4j blog, [Graph Algorithms in Neo4j: Weakly Connected Components](https://neo4j.com/blog/graph-algorithms-neo4j-weakly-connected-components/), we are provided with an excellent example of WCC graph algorithm. WCC has been used to keep track of approximate duplicate records in a database record system.

Strongly Connected Components

Pros: Uses a breadth-first search it can help you find the structure of a graph, like the WCC algorithm.

Cons: Needs a path to exist in both directions. Like WCC, this might just be a starting point in your analysis.

This is often the first type of algorithm that is used on a graph or network. It is used for things like suggesting pages/games/products to people based on the pages/games/products that were liked by people in their group. For example, someone who I’m closely connected with “liked” an advertisement for a movie on social media, since I’m connected with that person the algorithm would suggest showing me that movie advertisement because I am connected to a person who liked it.