Efficiently Clustering Dense Networks via Motif Counting

Halıcıoğlu Data Science Institute, UC San Diego, La Jolla

Dylan Lee
dklee@ucsd.edu

Matthew Wilson
m7wilson@ucsd.edu

Karthikeya Manchala
kmanchal@ucsd.edu

Jonathan Li
jzli@ucsd.edu

Barna Saha
bsaha@eng.ucsd.edu

Start of Pipeline

1 - SAMPLING
First, the network is sampled to obtain a subset graph, which requries less computational time and space to execute the pipeline upon.

2 - COUNTING
The motif counting algorithm is then applied onto the subset graph, obtaining the counts of a given motif in the form of a motif adjacency matrix.

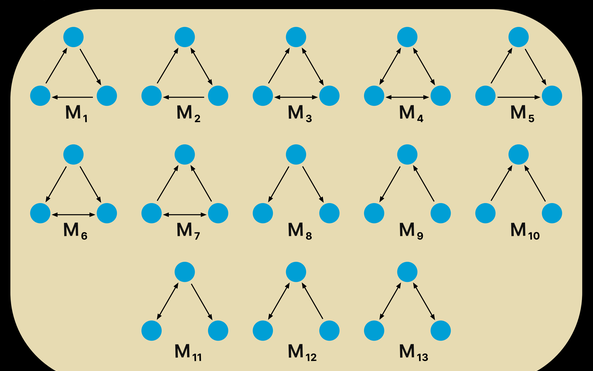
3 - CLUSTERING
The motif adjacency matrix is inputted to the spectral clustering algorithms to obtain clusters optimally separated by motif conductance.

4 - ANALYZING
Statistical metrics regarding the pipeline's execution time and cluster accuracy are measured to further analyze the overall effects of sampling.

End of Pipeline

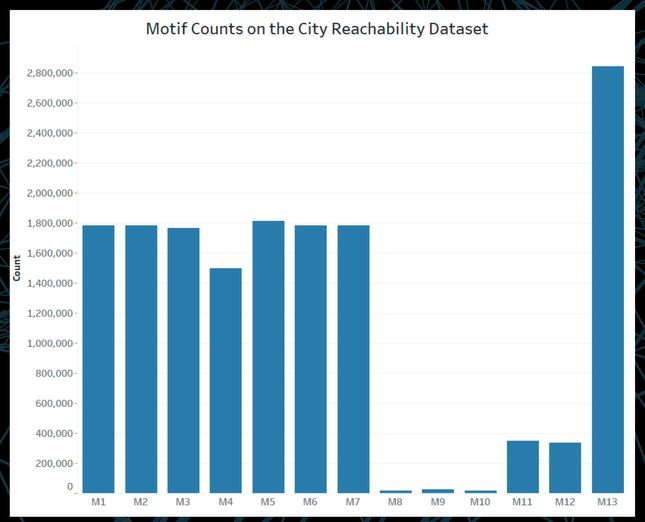
Introduction

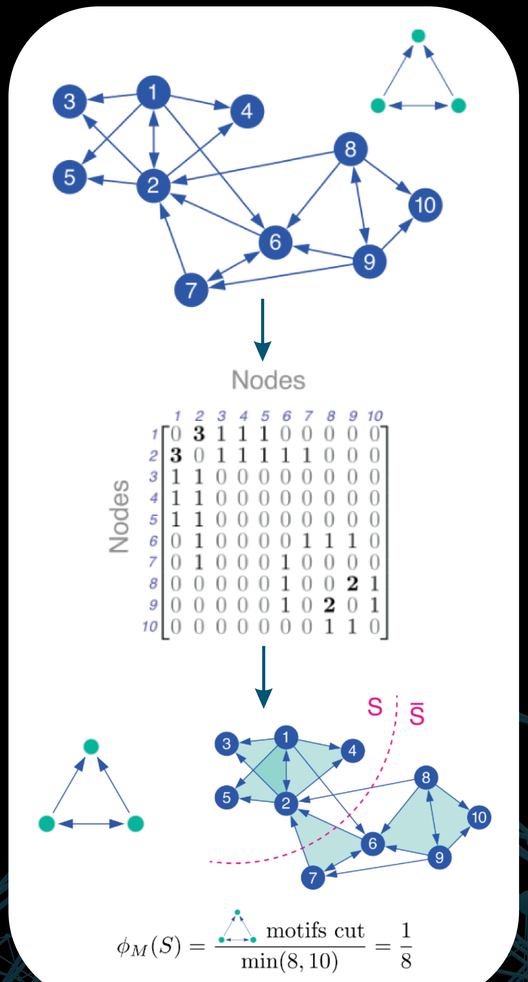
- Networks are popular in academia and industry for visualizing interconnected data structures
- Aim to identify separate and individual clusters of densely connected motif substructures in any network
- ***Goal:*** How can we perform this operation as efficiently (in terms of time and space) as possible?



Motif Counting

- Count the number of motif occurrences in the network
- Experimented with both naive methods of $O(n^3)$ complexity and efficient algorithms of $O(n^{1.5})$
- Certain motifs will be more prevalent in occurrence for context-specific graphs
- Most time-consuming step of the motif clustering pipeline



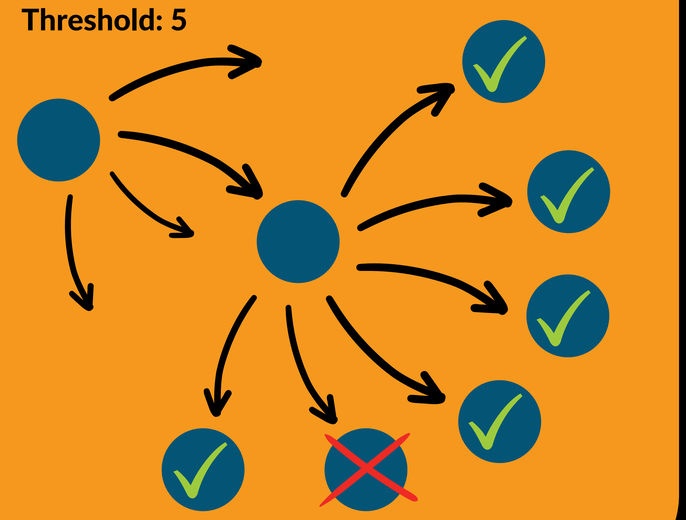


Motif Spectral Clustering

- An optimal set of clusters are those which minimize motif conductance
- Reducing the motif conductance enforces clusters to be well separated, but densely self-contained
- Motif counts are used to form the motif adjacency matrix
- Adjacency matrices are inputs to two spectral clustering algorithms, which both find clusters that minimize conductance
  - Algorithm 1: Identifies only a single cluster
  - Algorithm 2: Identifies multiple clusters

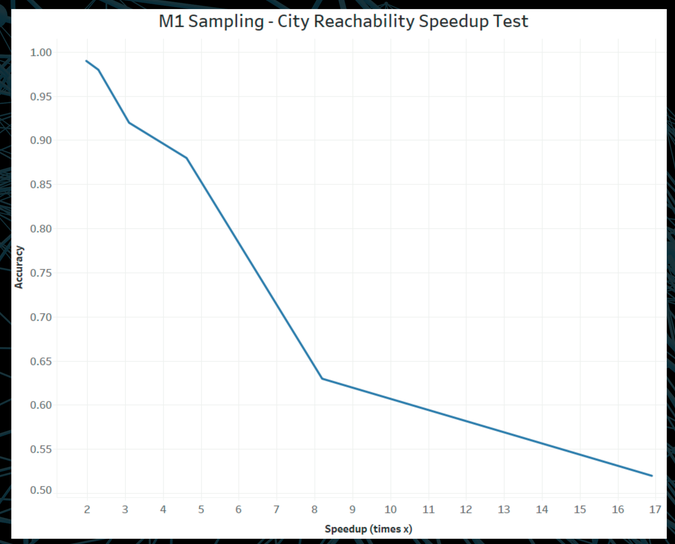
Network Sampling: Conclusion & Results

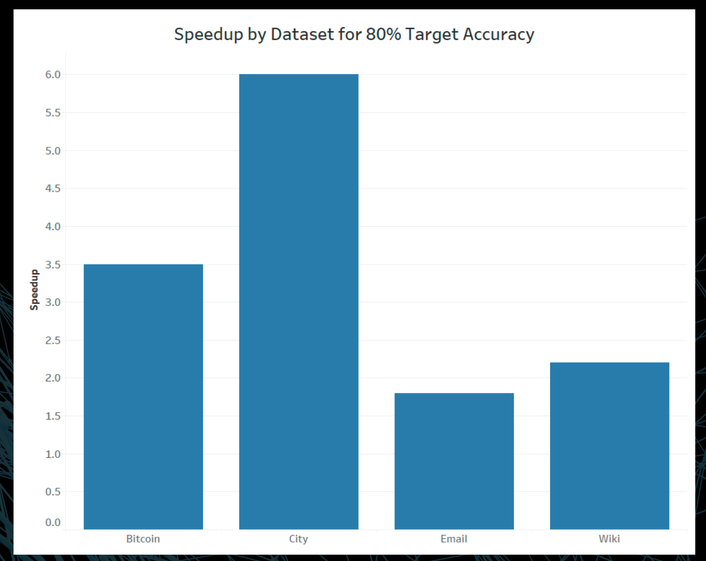
- Randomly sample neighboring nodes from the network to obtain a subset of the original graph
- Only sample neighbors from nodes which have a high number of connections (degree)
- Edges of lower degree nodes are naively counted
- Node to sample from varies with the shape and edge directionality of target motif
- Establish a threshold to differentiate high vs. low degree nodes
- Sampled motif counts are upscaled by a factor proportional to the degree of the sampled node
- ***Accuracy:*** How close a cluster obtained from a sampled graph would be from one obtained from the entire network
- ***Greatly reduces the time for motif counting, and the overall clustering pipeline - up to half the time reduced!***



Five of the six nodes are randomly sampled

Tradeoff between speeding up execution time by sampling a smaller subset against the accuracy of the resulting cluster





Obtainable speedups in execution time for 80% optimal cluster accuracy among various network graphs