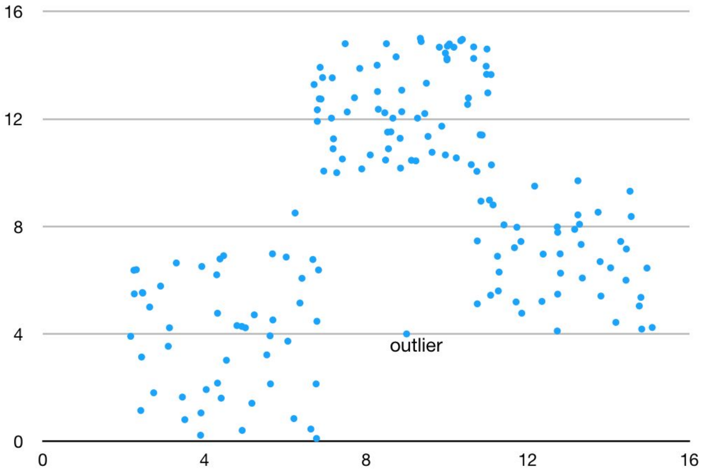
GPU Accelerated Graph Based Anomaly Detection

Ian Thomas, Enyue Lu

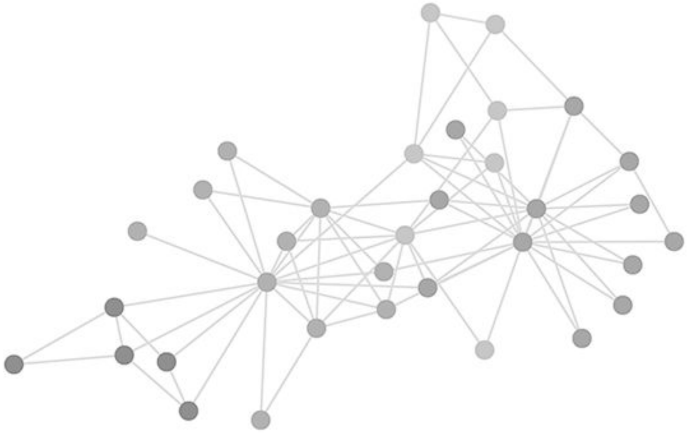
Introduction

- Anomaly detection
  - Field of data mining research
  - Identify data that deviates from datasets normal behavior
  - Commonly used to identify malicious activity
- Graph Theory
  - Study of mathematical structures used to demonstrate objects and their relationships
  - Graphs in this context: sets of vertices and edges connecting them
  - Used to represent interconnected networks, or interdependent data

Data Anomaly Example



Graph Example



Technological Introduction

- Graphics Processing Units (GPUs)
  - Commonly referred to “hardware accelerators”
  - Peripheral computer component, typically used only for rendering graphics
  - Recently being explored for general purpose computing due to arithmetic capability
- NVIDIA CUDA
  - Programming platform developed by NVIDIA corp.
  - Allows developers to write specific parts of program to run on CPU or GPU

Project aims

- Continue on-going research into graph-based anomaly detection systems
- Explore GPU accelerated implementations of common anomaly detection systems
- Compare two mainstream anomaly detection approaches

Approaches Explored

- Two common approaches are explored in this experiment
  - Graph based compression
  - Graph based clustering
- Using KDD-Cup 1999 dataset
  - Set of five million network records
  - Records labeled either normal or some attack type

Graph Compression Approach

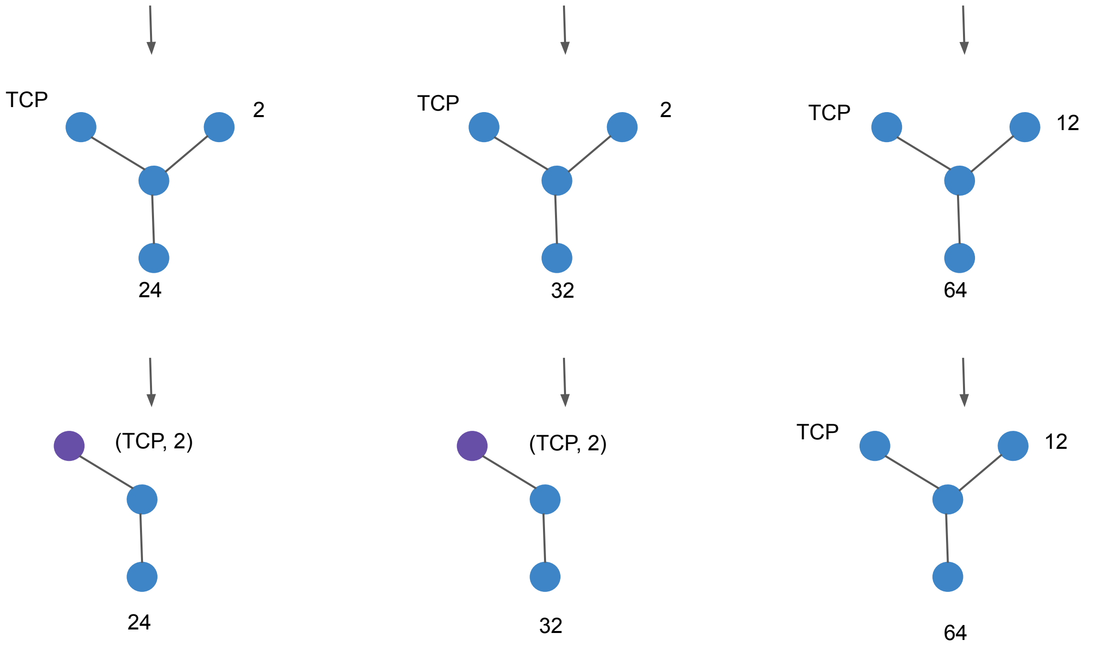
- SUBDUE graph compression algorithm
    - Commonly used in data mining applications
    - Finds most common substructures in graph, compresses substructures with a single vertex
    - Used for reducing size of graphs for storage, identifying common attributes
- Can be used in finding anomalous network records
    - Represent individual records in star-graph configurations
    - Hub vertex represents record itself, all attributes are leaf vertices
    - Orientation of leaves are compressed based on frequency among other records
    - Records given score based on how compressed star graphs are

Graph Compression Approach

Raw Network Records:	(TCP, 2, 24)	(TCP, 2, 32)	(UDP, 12, 64)

Star-graph representation:

After Compression:

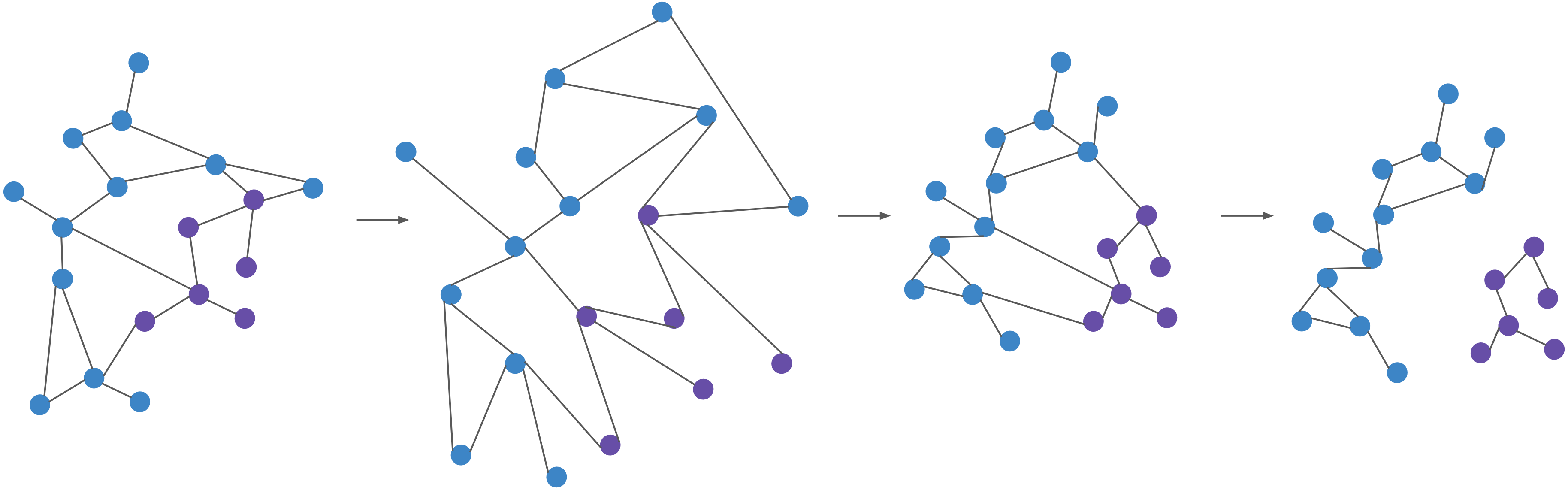


Final Score:	.33	.33	.67

Graph Clustering Approach

- Barycentric Clustering algorithm
  - Based on physical model of interconnected springs
  - Springs connected by a stronger force will naturally form clusters
  - Randomly set vertices to random locations, iteratively alter their positions based on force exerted by neighbors
- Vertices are representative of network records
  - All vertices are connected by a unique edge
  - Edge weights decided by similarity measure
  - After algorithm terminates, edges of above average weight are cut

Graph Clustering Approach



Comparison of Approaches

- Performance
  - Compression must locate most common unique pairing on each iteration (CkN * R^2 time)
  - Clustering builds graph of dataset once, taking R^2 time
- Accuracy
  - Both approaches have very high accuracy (on all datasets tested, above 98%)
  - Measured differently on each approach
- Smaller datasets more suitable for compression, larger for clustering
  - Runtime for compression makes it much slower on larger datasets
    - ~5000 records will take .89 seconds for clustering, 11.2 seconds for compression

References

[1] S. D. Bay, D. F. Kibler, M. J. Pazzani, and P.SmythThe UCI KDD Archive of Large Data Sets for Data Mining Research and Experimentation. 2000.

[2] Jayshree Ghorpade, Jitendra Parande, Madhura Kulkarni, Amit Bawaskar GPGPU Processing In CUDA Architecture. Jan. 2012.

[3] Joyce, B.Graph Based Anomaly Detection using MapReduce on Network Records. University of North Carolina, August 2016

[4] Jonathan Cohen Barycentric Graph Clustering. 2008.

[5] Corbin McNeill, Enyue Lu, Matthias Gobbert Distributed Graph-Based Clustering for Network Intrusion Detection. Wheaton College, IL, 2015