Graph Mining: Laws, Generators and Algorithms

Deepayan Chakrabarti and Christos Faloutsos
Yahoo Research and Carnegie Mellon University

ACM Computing Surveys (CSUR)

https://qdata.github.io/deep2Read

https://qdata.github.io/deep2Read

Presenter: Arshdeep Sekhon

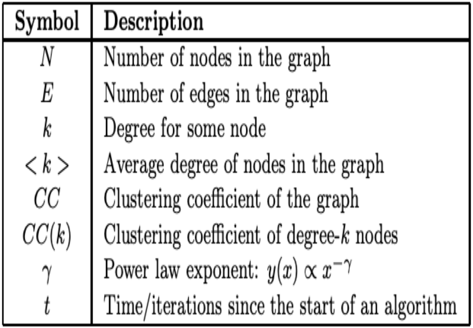
Overview

1. Introduction
2. Graph Patterns
   - Power Law
   - Graph Patterns: Small Diameters/Small World
   - Community Structure
3. Graph Patterns in Evolving Graphs
4. Graph Generators
   - Random Graph Models
   - Preferential Attachment
   - Topology from Resource Optimizations
   - Incorporating Geographical Information
5. RMAT Generator

Introduction

- ubiquitous graph data : sociology to biology
- Two questions:
  - what are the characteristics of ‘real’ graphs?
  - How to generate synthetic graphs that are realistic?

Notations



Symbol	Description
N	Number of nodes in the graph
E	Number of edges in the graph
k	Degree for some node
$<k>$	Average degree of nodes in the graph
CC	Clustering coefficient of the graph
$CC(k)$	Clustering coefficient of degree-k nodes
γ	Power law exponent: $y(x) \propto x^{-\gamma}$
t	Time/iterations since the start of an algorithm

Table of Contents

1. Introduction
2. Graph Patterns
   - Power Law
   - Graph Patterns: Small Diameters/Small World
   - Community Structure
3. Graph Patterns in Evolving Graphs
4. Graph Generators
   - Random Graph Models
   - Preferential Attachment
   - Topology from Resource Optimizations
   - Incorporating Geographical Information
5. RMAT Generator

Graph Patterns

- Finding a basic set of attributes(good patterns) of both graphs from natural and man made phenomena
- Common Patterns for multiple different man made and natural phenomenaa generated graphs
- If we want to create the pattern in simulation data: computational complexity of pattern

Graph Patterns

Three major types of graph patterns characterizing naturally occurring graphs:

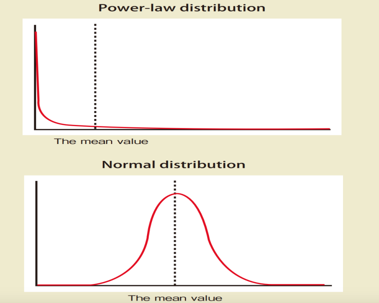
- power laws
- small diameters
- community effects

Power Law followed by degree of nodes

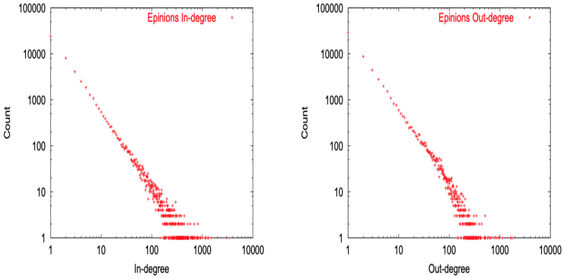
- fraction of nodes with degree k follows a power law
- probability density function related as:

$$p(x) = Ax^{-\gamma} \tag{1}$$

- $\gamma > 1$ and $x \geq x_{min}$
- example, Internet graph: count of nodes with degree k



Power Law Distribution



(a) Epinions In-degree (b) Epinions Out-degree

the in-degree and out-degree distributions on a log-log scale for an online social network Epinions graph

Detecting Power Law pattern in a graph

- scatter plot for degree distribution
- find the power law exponent γ For example, using linear regression on log-log scale
- check for goodness of fit

Deviations from power law

- Exponential Cut offs: the distribution looks like a power law over the lower range of values along the x-axis, but decays very fast for higher values.

$$y(x = k) \propto e^{-\frac{k}{\kappa}} k^{-\gamma} \tag{2}$$

- Log normal distribution: some web networks do not follow power law, instead follow log normal distribution.

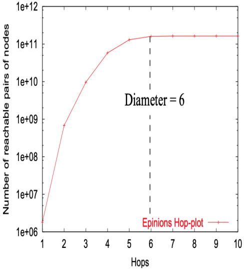
Graph Pattern 2: Small Diameters/Small World property

- Participants were asked to reach a randomly assigned target individual using a chain letter
- For complete chains, the average length of such chains was six
- The diameters of several naturally occurring graphs have been calculated, and in almost all cases they are very small compared to the graph size
- For example, effective diameter of around 4 for the Internet AS level graph and around 12 for the Router level graph.

Measuring Effective Diameter in a graph

Hop Plot

Starting from a node u in the graph, find the number of nodes $N_h(u)$ in a neighborhood of h hops. Repeat this starting from each node in the graph, and sum the results to find the total neighborhood size N_h for h hops ($N_h = \Sigma_u N_h(u)$). The hop-plot is just the plot of N_h versus h.



Ways to measure small world property from Hop plots

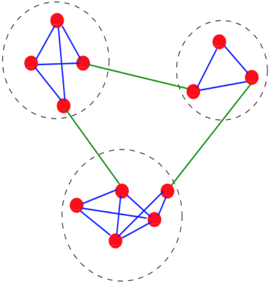
- **Effective Diameter or eccentricity**: minimum number of hops in which some fraction (say, 90%) of all connected pairs of nodes can reach each other
- **Characteristic path length**: consider the shortest paths from it to every other node in the graph. Take the average length of all these paths. Now, consider the average path lengths for all possible starting nodes, and take their median
- **Average diameter**: Same as above, instead take average in the last step

Computational Issues

- Hop plot: repeatedly multiplication of adjacency matrix : $O(N^{2.88})$ time and $O(N^2)$ memory space
- breadth first search: $O(N + E)$ space but requires $O(NE)$ time

Pattern 3: Community Structure

- Nodes belong to a community if they are closer to each other than nodes outside their community
- measure of ‘clumpiness’ of a graph
- friends of friends are likely to be friends



Measure of Community Structure: Clustering Coefficient

- Node Clustering Coefficient: Suppose a node i has k_i neighbors, and there are n_i edges between the neighbors

$$C_i = \begin{cases} \frac{n_i}{k_i} & k_i > 0 \\ 2 & k_i = 0 \text{ or } 1 \end{cases} \tag{3}$$

- Graph Clustering Coefficient

$$C = \frac{3 \times \text{number of triangles in the graph}}{\text{number of connected triples in the graph}} \tag{4}$$

$$C = \Sigma_{i=1}^{N} C_i / N \tag{5}$$

Clustering Coefficients in the real world

- almost always larger in real-world graphs than in a random graph with the same number of nodes and edges

Some other patterns

- Resilience to random failures of nodes or edges, but vulnerable to targeted attacks
- “higher-order” statistics: In and out degree correlation,Average neighbor degree,Neighbor degree correlation

Table of Contents

1. Introduction
2. Graph Patterns
   - Power Law
   - Graph Patterns: Small Diameters/Small World
   - Community Structure
3. Graph Patterns in Evolving Graphs
4. Graph Generators
   - Random Graph Models
   - Preferential Attachment
   - Topology from Resource Optimizations
   - Incorporating Geographical Information
5. RMAT Generator

Patterns in evolving graphs

- Previous cases were about static patterns in a snapshot of a graph
- How do graphs evolve over time?
  - Densification power law: graphs grow over time, degrees increase over time :the number of nodes $N(t)$ at time t is related to the number of edges $E(t)$

$$E(t) \propto N(t)^{\alpha} \quad 1 \leq \alpha \leq 2 \tag{6}$$

  - average diameter decreases over time

Table of Contents

1. Introduction

2. Graph Patterns
   - Power Law
   - Graph Patterns: Small Diameters/Small World
   - Community Structure

3. Graph Patterns in Evolving Graphs

4. Graph Generators
   - Random Graph Models
   - Preferential Attachment
   - Topology from Resource Optimizations
   - Incorporating Geographical Information

5. RMAT Generator

Graph Generators

- models to generate graphs for simulations
- mimic patterns found in natural graphs

Types of Graph Generators

- Random Graph Generators
- Preferential Attachment Generators
- Optimization-based Generators
- Geographical Models
- Domain Specific: Internet Specific Models

Graph Generator I : Random Graph Model

- Erdos and Renyi Random Graph Model
- Generalized Random Graph Model

1. Erdos and Renyi Random Graph Model

Generate a set of graphs: $G_{N,p}$

- Start with N nodes
- for every pair add an edge with probability p
- the degree distribution

$$p_k = \binom{N}{k} p^k (1-p)^{N-k} \approx \frac{z^k e^{-z}}{k!} \quad \text{with } z = p(N-1)$$

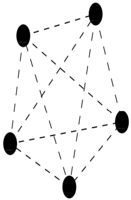


Figure: Every possible edge occurs with equal probability

Properties of Erdos and Renyi Random Graph Model

- Random graphs have a diameter concentrated around log N/ log z, where z is the average degree of the nodes in the graph.
- the diameter grows slowly as the number of nodes increases.
- The probability that any two neighbors of a node are themselves connected is the connection probability $p = \frac{<k>}{N}$, where $<k>$ is the average node degree.
- the clustering coefficient:

$$CC_{random} = \frac{<k>}{N} \tag{7}$$

Disadvantages

- Not a realistic graph:
  - Their degree distribution is Poisson, which has a very different shape from power-laws or lognormals.
  - no correlations between the degrees of adjacent nodes,
  - nor does it show any form of “community” structure (which often shows up in real graphs like the WWW).
  - For real world graphs, $CC_{random}/ < k >$ independent of N

2. Generalized Random Graph Model

- extend the basic random graph model to allow arbitrary degree distributions.
- Given a degree distribution, we can randomly assign a degree to each node of the graph so as to match the given distribution.

$$p_k \propto k^{-\beta} \tag{8}$$

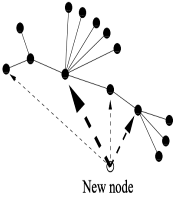
- Rest process remains same
- but still only matches one graph pattern

Graph Generator II : Preferential Attachment

- Generalized random graph models try to model the power law or other degree distribution of real graphs.
- PROCESS to generate a network
- Principle: power law tails arise when “the rich get richer”

1. Barabasi-Albert Model

- the BA model starts off with a small set of nodes and grows the network as nodes
- edges are added over time to result in an undirected network
- The probability of connecting to a node is proportional to the current degree of that node
- $$P(\text{edge to existing vertex v}) = \frac{k(v)}{\Sigma_i k(i)} \tag{9}$$



Properties

- The degree distribution of the BA model is power law with exponent 3
- this model displays the small-world effect: the distance between two nodes is, on average, far less than the total number of nodes in the graph.

2. Edge Copying Model

- Several graphs show community behavior, such as topic-based communities of websites on the WWW.
- the intuition that most webpage creators will be familiar with webpages on topics of interest to them, and so when they create new webpages, they will link to some of these existing topical webpages
- Creates a directed graph

Edge Copying Model

- In each iteration, nodes may be independently created and deleted under some probability distribution.
- For each added node, choose some node v and some number of edges k to add to node v.
- With probability β, these k edges are linked to nodes chosen uniformly and independently at random.

Edge Copying Model

- With probability $1-\beta$, edges are copied from another node: choose a node u at random, choose k of its edges (u, w), and create edges (v, w).
- If the chosen node u does not have enough edges, all its edges are copied and the remaining edges are copied from another randomly chosen node.
- Edge deletion: Random edges can be picked and deleted according to some probability distribution
- lead to both power laws as well as community effects.

Graph Generator III : Topology from Resource Optimizations

- another point of view is that power laws can result from resource optimizations
- power laws may arise in systems due to tradeoffs between yield (or profit), resources (to prevent a risk from causing damage) and tolerance to risks.

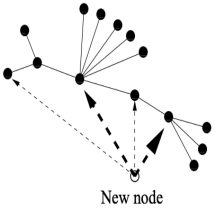
The Highly Optimized Tolerance model.

- we have n possible events
- each with an associated probability $p_i (1 \leq i \leq n)$.
- Each event can lead to some loss l_i: a function of the resources r_i allocated for that event: $l_i = f(r_i)$.
- Also, the total resources are limited: $\Sigma_i r_i \leq R$ for some given R.
- The aim is to minimize the expected cost

$$J = \left\{ \sum_i p_i l_i \mid l_i = f(r_i), \sum_i r_i \leq R \right\}$$

1. The Heuristically Optimized Tradeoffs model.

- resource allocation may not be globally optimal: go for local optimality
- nodes spread out on geographical area
- add node based on length of wire to connect link, and transmission delays
- Thus, a new node i should be connected to old node j: $\alpha d_{i,j} + h_j$
- A new node prefers to link to existing nodes which are both close in distance and occupy a “central” position



Graph Generator IV: Incorporating Geographical Information

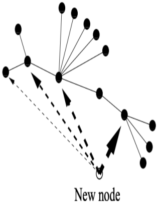
- Both the random graph and preferential attachment models have neglected one attribute of many real graphs: the constraints of geography.
- For example, easier (cheaper) to link two routers which are physically close to each other; most of our social contacts are people we meet often, and who consequently probably live close to us (say, in the same town or city), and so on.

1. The Waxman Generator

- The Waxman generator places random points in Cartesian two-dimensional space
- An edge (u, v) is placed between two points u and v:

$$P(u,v) = \beta exp(\frac{-d(u,v)}{L\alpha}) \tag{10}$$

- not a power law distribution of nodes



2. BRITE Generator

- Nodes placed randomly on a grid based on a grid or with a heavy-tailed distribution
- start off by placing all the nodes at once or we could add nodes and links incrementally
- combine preferential connectivity with geographical constraints

$$P(u,v) = \frac{w(u,v)k(v)}{\sum_i w(u,i)k(i)}$$

$$\text{where } w(u,v) = \beta \exp \frac{-d(u,v)}{L\alpha}$$

Table of Contents

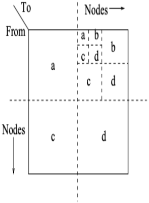
1. Introduction
2. Graph Patterns
   - Power Law
   - Graph Patterns: Small Diameters/Small World
   - Community Structure
3. Graph Patterns in Evolving Graphs
4. Graph Generators
   - Random Graph Models
   - Preferential Attachment
   - Topology from Resource Optimizations
   - Incorporating Geographical Information
5. RMAT Generator

Recursive Matrix Generator

Tries to meet the following desired characteristics:

- The generated graph should match several graph patterns, including but not limited to power-law degree distributions (such as hop-plots and eigenvalue plots).
- It should be able to generate graphs exhibiting deviations from power-laws
- exhibit a strong “community” effect.
- should be able to generate directed, undirected, bipartite or weighted graphs with the same methodology.
- should use as few parameters as possible.
- There should be a fast parameter-fitting algorithm and generation algorithm should be efficient and scalable.

RMAT



- Given 2^n nodes and E edges
- start with an empty adjacency matrix, and divide it into four equal-sized partitions.
- One of the four partitions is chosen with probabilities a, b, c, d respectively ($a + b + c + d = 1$)
- The chosen partition is again subdivided into four smaller partitions, and the procedure is repeated until we reach a simple cell ($=1 \times 1$ partition).
- The nodes corresponding to this cell are linked by an edge in the graph.
- repeated E times to generate the full graph.
- To smooth out fluctuations in the degree distributions, some noise is added to the (a, b, c, d) values at each stage of the recursion, followed by renormalization

Summary

- Universal Graph Patterns
- Detecting Patterns
- Graph generators to mimic the patterns
- Almost all graph generators focus on only one or two patterns, typically the degree distribution