



# Scale Free Networks

---

Machine Learning/ Complex Systems



**Many structures in the real world can be mapped to graphs by describing the relations between entities in terms of vertices and edges**

Stanford Network Analysis Project

**Physics Collaboration Graph**



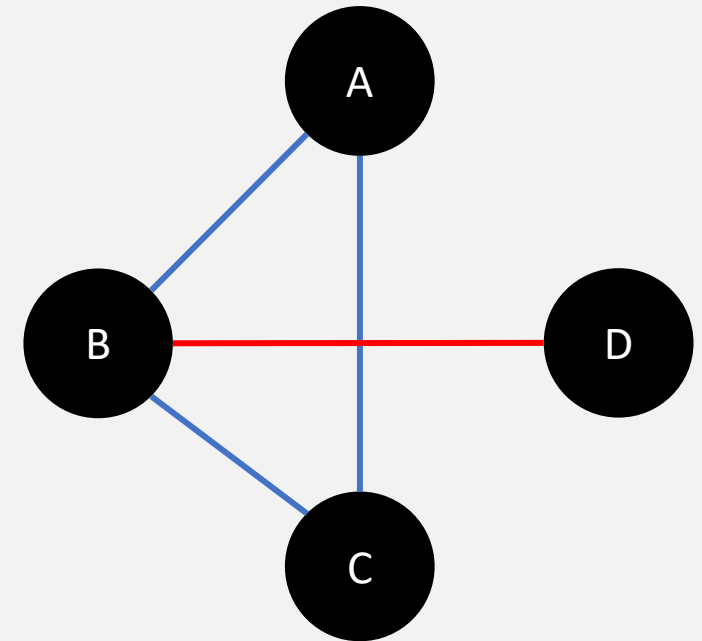
Author A  
Author B  
Author C



Twitter Higgs Dataset

Beeradvocate Reviews

Author B  
Author D



**Graph Evolution: Densification and Shrinking Diameters** - J. Leskovec, J. Kleinberg and C. Faloutsos

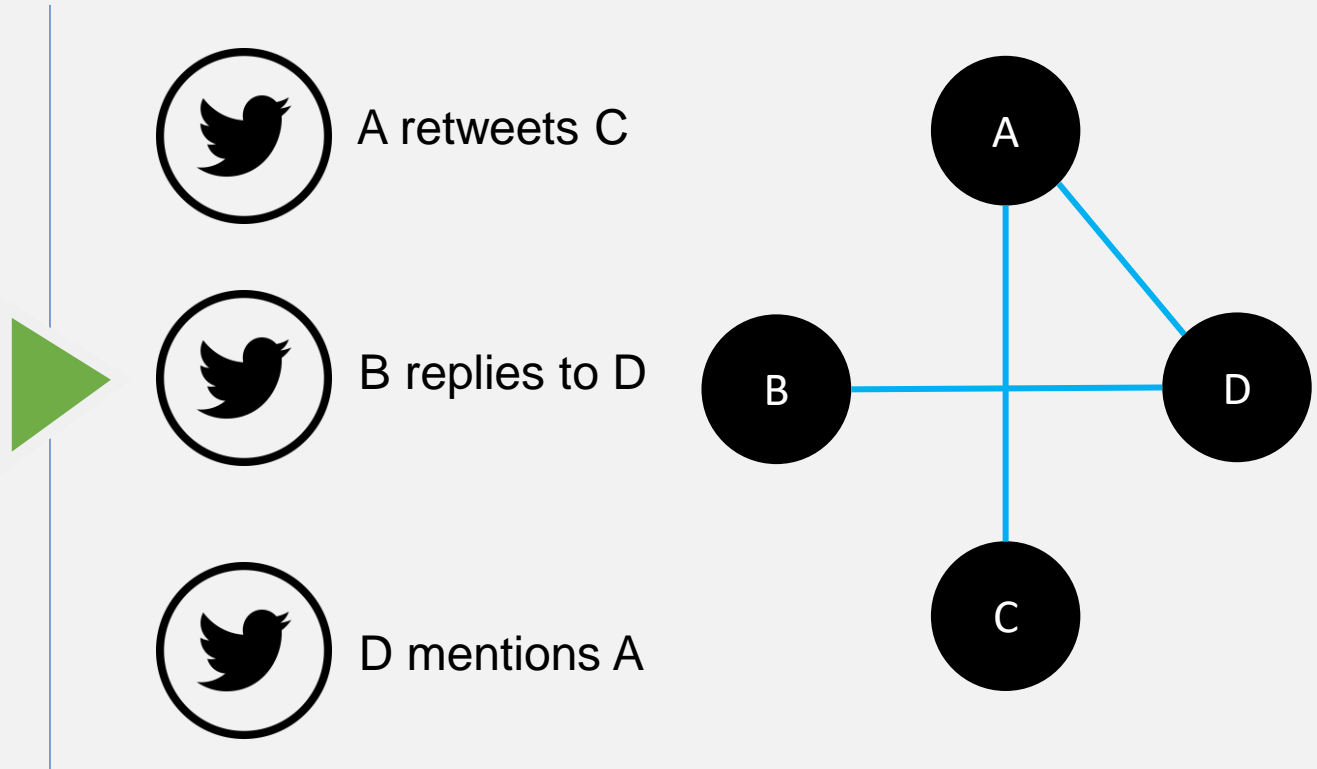
**Many structures in the real world can be mapped to graphs by describing the relations between entities in terms of vertices and edges**

Stanford Network Analysis Project

Physics Collaboration Graph

**Twitter Higgs Dataset**

Beeradvocate Reviews



**The Anatomy of a Scientific Rumor** – M. De Domenico, A. Lima, P. Mougél and M. Musolesi

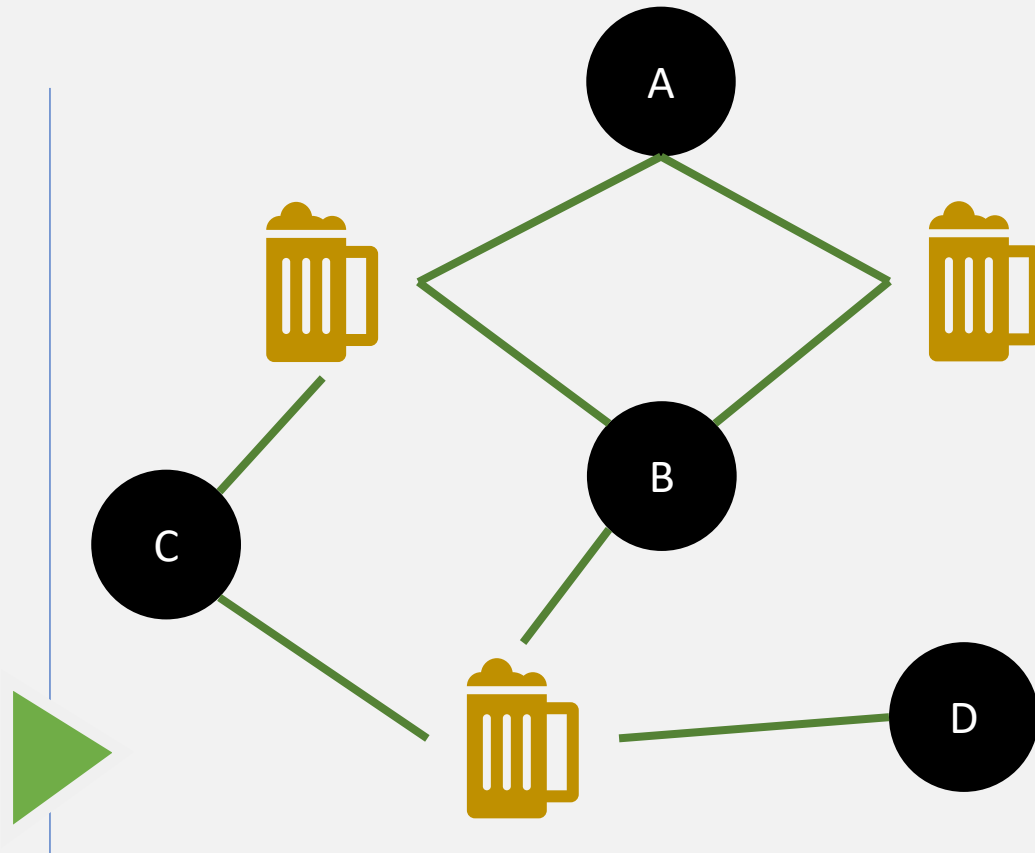
**Many structures in the real world can be mapped to graphs by describing the relations between entities in terms of vertices and edges**

## Stanford Network Analysis Project

Physics Collaboration Graph

Twitter Higgs Dataset

**Beeradvocate Reviews**



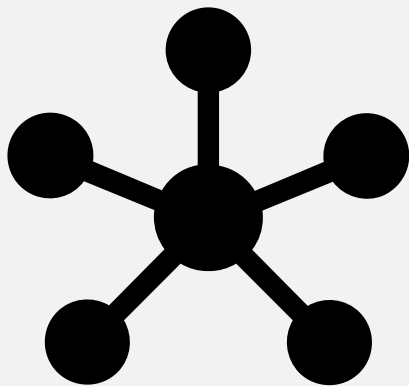
From amateurs to connoisseurs: modeling the evolution of user expertise through online reviews.

- J. McAuley and J. Leskovec

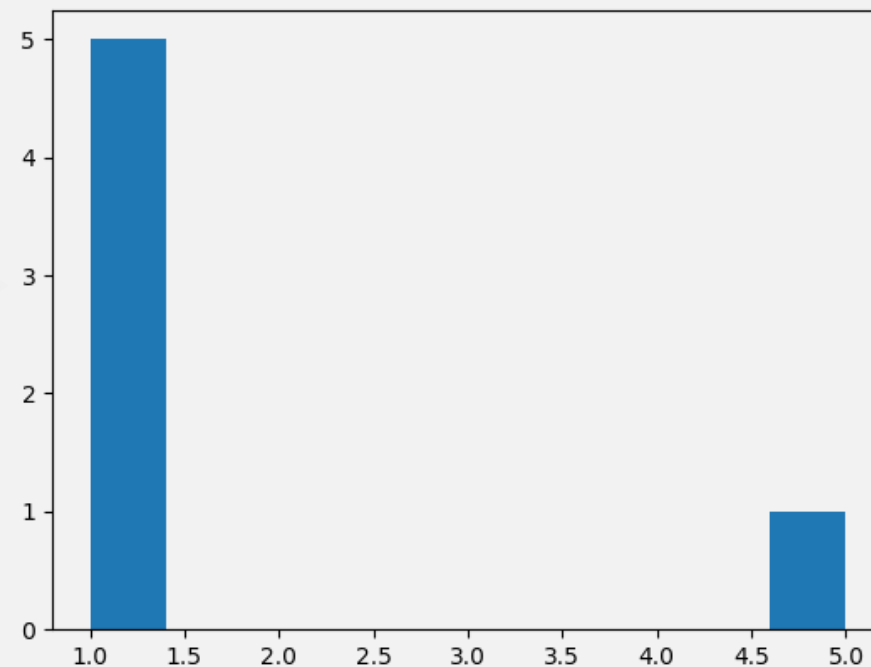
## Degree Distribution

**Degree distributions are a way to characterize networks. They are the probability distribution for a vertex to have  $k$  edges**

**Graph**



**Histogram**



**Degree distribution**

k	p(k)
0	0
1	5/6
2	0
3	0
4	0
5	1/6

In this talk: Power law distributions  $p(k) \sim k^{-\gamma}$

## Academic studies support the hypothesis that real world networks have a power law distribution

### Word Wide Web

- Vertices are websites
- Edges are links
- Size: 325729
- Average Degree: 4.51
- $\gamma_{In} = 2.1$
- $\gamma_{out} = 2.45$
- Average path length: 11.2

### Movie Actor Collaboration

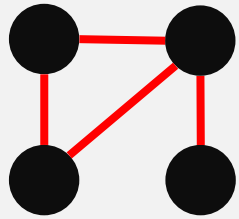
- Vertices are Actors
- Edge between actors if they appear in the same movie
- Size: 212250
- Average Degree: 28.78
- $\gamma = 2.3$
- Average path length: 4.54
- Cluster coefficient: 0.79

### Cellular Networks

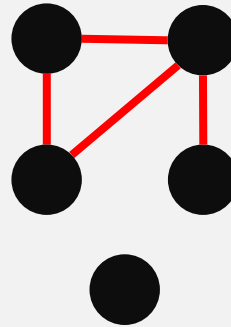
- Vertices are substrates
- Edges are chemical reactions
- $\gamma \in [2.0, 2.4]$
- Average path length: 3.3

## A power law distribution can be achieved by modelling the dynamics of real world networks

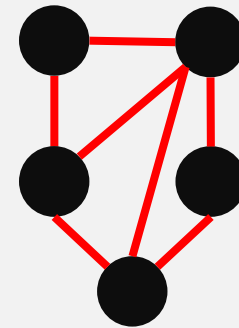
Start with random  
Network of size  $m_0$



Network Grow



Preferential Attachment



1. Start with a random network of size  $m_0$
2. Add a new vertex to the Network
3. Attach the new Node to the already present nodes with  $m \leq m_0$  edges
4. The probability to attach to a vertex with degree  $k_i$  is given by  $\Pi(k_i) = \frac{k_i}{\sum_{j=1} k_j}$
5. Repeat at step 2

**Useing the continuum approach assuming  $k$  is a continuous variable over time leads to an differential equation for  $k$**

$$\frac{\partial k_i}{\partial t} = m \Pi(k_i) = m \frac{k_i}{\underbrace{\sum_{j=1}^{N-1} k_j}} \approx \frac{k_i}{2t} \quad \Rightarrow \quad k_i(t) = m \left( \frac{t}{t_i} \right)^{\frac{1}{2}} \text{ With } k_i(t_i) = m$$

$$\sum_{j=1}^{N-1} k_j = 2mt - m$$

$$P(k_i(t) < k) = P\left(t_i > \frac{m^2 t}{k^2}\right) = 1 - \frac{m^2 t}{k^2(t + m_0)}$$

$$P(k) = \frac{\partial P(k_i(t) < k)}{\partial k} \sim \frac{1}{k^3}$$



## Apart from the degree distribution there are several other properties which can be measured

### Average Degree

- Average number of edges per node
- Barabasi-Albert:  $2m$

### Average Path length

- Average path length from one node to all other nodes averaged over all nodes
- Barabasi-Albert:  $\sim \frac{\ln(N)}{\ln(\ln(N))}$

### Average Cluster coefficient

- $\frac{3 \times \#Triangles\ in\ the\ Graph}{\#Connected\ triples}$
- No analytical result for Barabasi-Albert graphs

---

### Numerical calculation:

1. Create an ensemble of graphs
2. Compute the given property for every graph in the ensemble
3. Average over the ensemble

## Error and attack tolerance are properties of networks strongly motivated by real world observations

### Error Tolerance

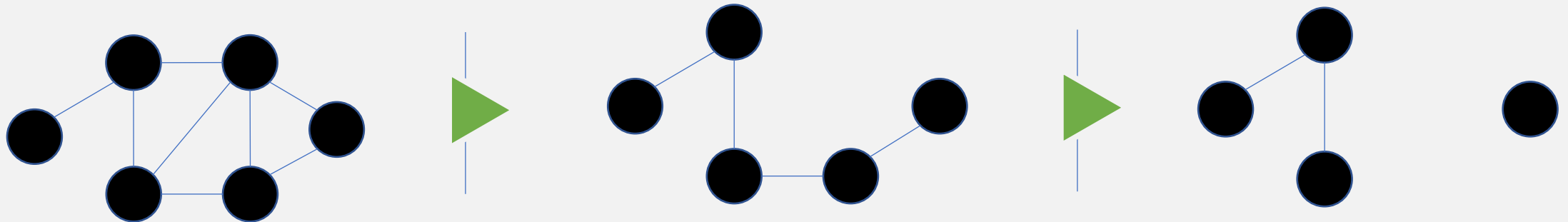
- How does the Network react to random failures of vertices

### Attack Tolerance

- How does the network react to targeted attacks

### Use the average path length of the largest component to quantify the changes under errors/attacks

- Measures differences in the network when the network stays connected
- Stays finite even when the network contains more than one component



**In the Barabási-Albert Model the oldest vertices will gain the most edges. This does not match with reality**

**Idea:** Give each node an additional constant fitness parameter to adjust the preferential attachment

$$\frac{\partial k_i}{\partial t} = m \Pi(k_i) = m \frac{\eta_i k_i}{\sum_{j=1}^{N-1} \eta_j k_j}$$

$$\Pi(k_i) = \frac{\eta_i k_i}{\sum_{j=1} \eta_j k_j} \text{ where } \eta \sim p(\eta)$$

## Bose Einstein Condensation

- Assign to each node an energy
- An edge between two nodes corresponds to two particles at the linked energy levels

Energy levels are given by:  $\epsilon_i = -\frac{1}{\beta} \ln(\eta_i)$

The distribution of energy levels transforms according to the probability density transformation law:

$$g(\epsilon) = \beta p(e^{-\beta\epsilon}) e^{-\beta\epsilon}$$

## The fitness model is closely linked to statistical physics and Bose-Einstein Condensation

$$I(\beta, \mu) = \int d\epsilon g(\epsilon) \frac{1}{e^{\beta(\epsilon - \mu)} - 1} = 1$$

Integral has a solution

Integral has no solution

### Fit get rich phase

- The fraction of particles on every energy level decays to zero as  $t \rightarrow \infty$

### Bose Einstein Condensation

- A finite fraction of particles will occupy the fittest node as  $t \rightarrow \infty$

# References

- Emergence of Scaling in Random networks – Barabasi, A.-L. and R. Albert (1999)
- Statistical Mechanics of complex – Albert R. and A.-L. Barabasi (2002)
- Competition and multiscaling in evolving networks – Bianconi, G. and A.-L. Barabasi (2000)
- Bose-Einstein condensation in complex networks – Bianconi, G. and A.-L. Barabasi (2000)

# GitHub

- [www.github.com/HOminus/ScaleFreeNet](https://www.github.com/HOminus/ScaleFreeNet)