Social Media Analytics: Revision of Session 8-15

Dr. Prasad Ramanathan
p_ramanathan@wilp.bits-pilani.ac.in

BITS Pilani

Pilani Campus

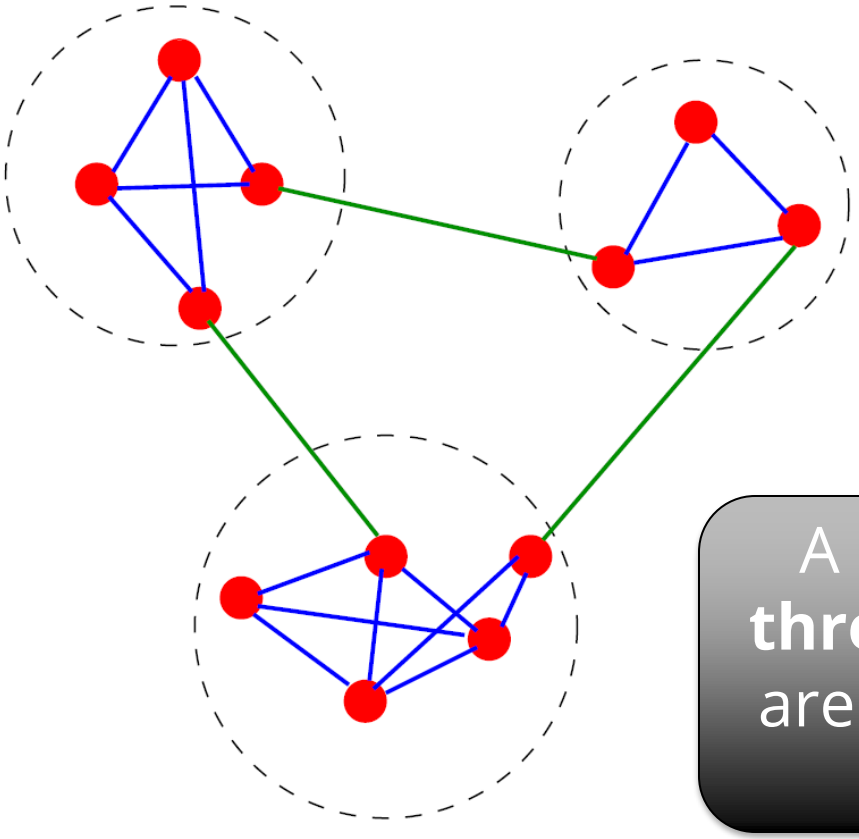
Community Analysis

Acknowledgment

Course material from the following source is gratefully acknowledged:

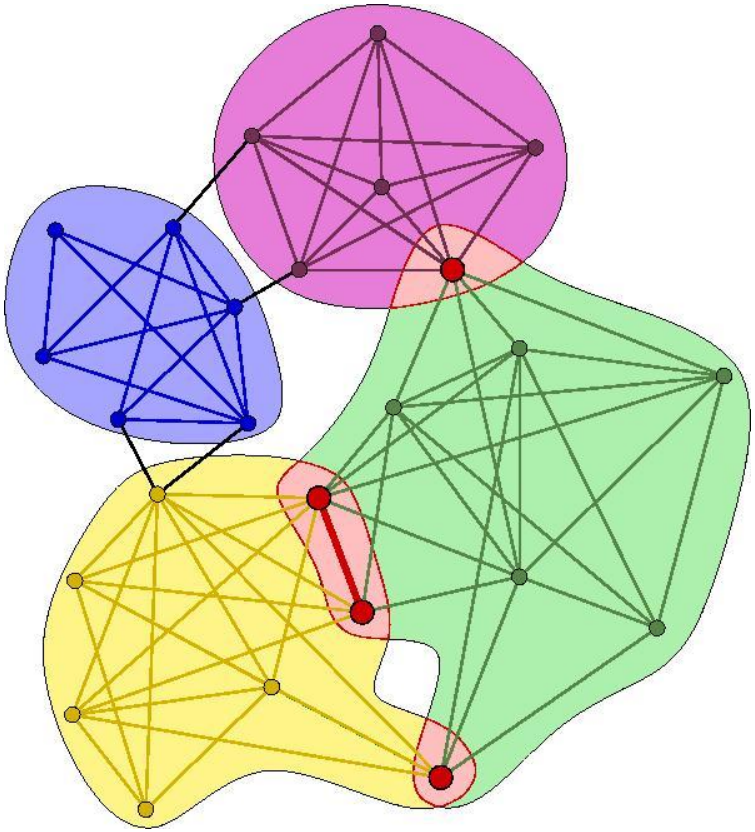
R. Zafarani, M. A. Abbasi, and H. Liu, *Social Media Mining: An Introduction*, Cambridge University Press, 2014.
Free book and slides at **http://socialmediamining.info/**

Finding Implicit Communities: An Example

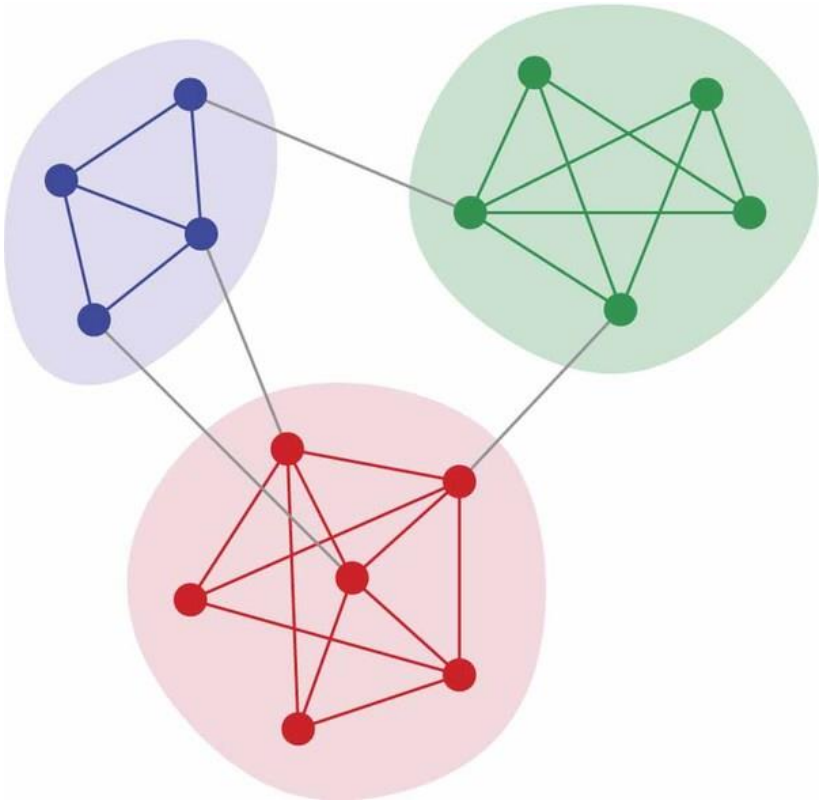


A simple graph in which **three** implicit communities are found, enclosed by the dashed circles

Overlapping vs. Disjoint Communities



Overlapping Communities



Disjoint Communities

What is Community Analysis?

innovate achieve lead

Community detection

Discovering implicit communities

Community evolution

Studying temporal evolution of communities

Community evaluation

Evaluating Detected Communities

Community Detection

What is community detection?

The process of finding clusters of nodes ("*communities*")

With **Strong** internal connections and **Weak** connections between different communities

Ideal decomposition of a large graph

Completely disjoint communities

There are no interactions between different communities.

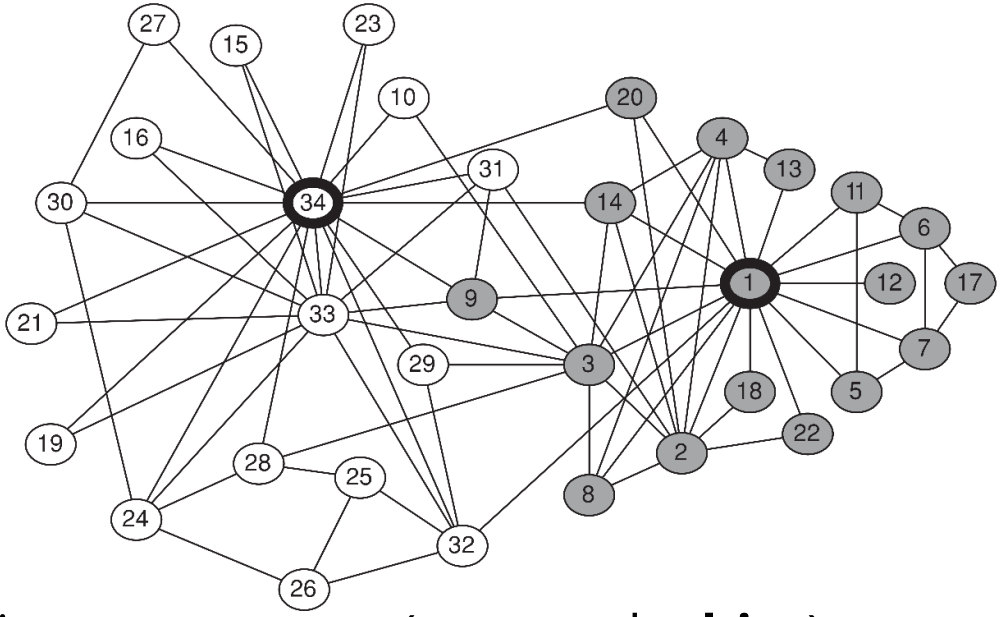
In practice,

find community partitions that are maximally decoupled.

Why Detecting Communities is Important?

Zachary's karate club

Interactions between 34 members of a karate club for over two years



- The club members split into two groups (**gray** and **white**)
- Disagreement between the administrator of the club (node **34**) and the club's instructor (node **1**),
- The members of one group left to start their own club

The same communities can be found using community detection

Community Detection vs. Clustering

Clustering

- Data is often non-linked (matrix rows)
- Clustering works on the distance or similarity matrix, e.g., k-means.
- If you use k-means with adjacency matrix rows, you are only considering the ego-centric network

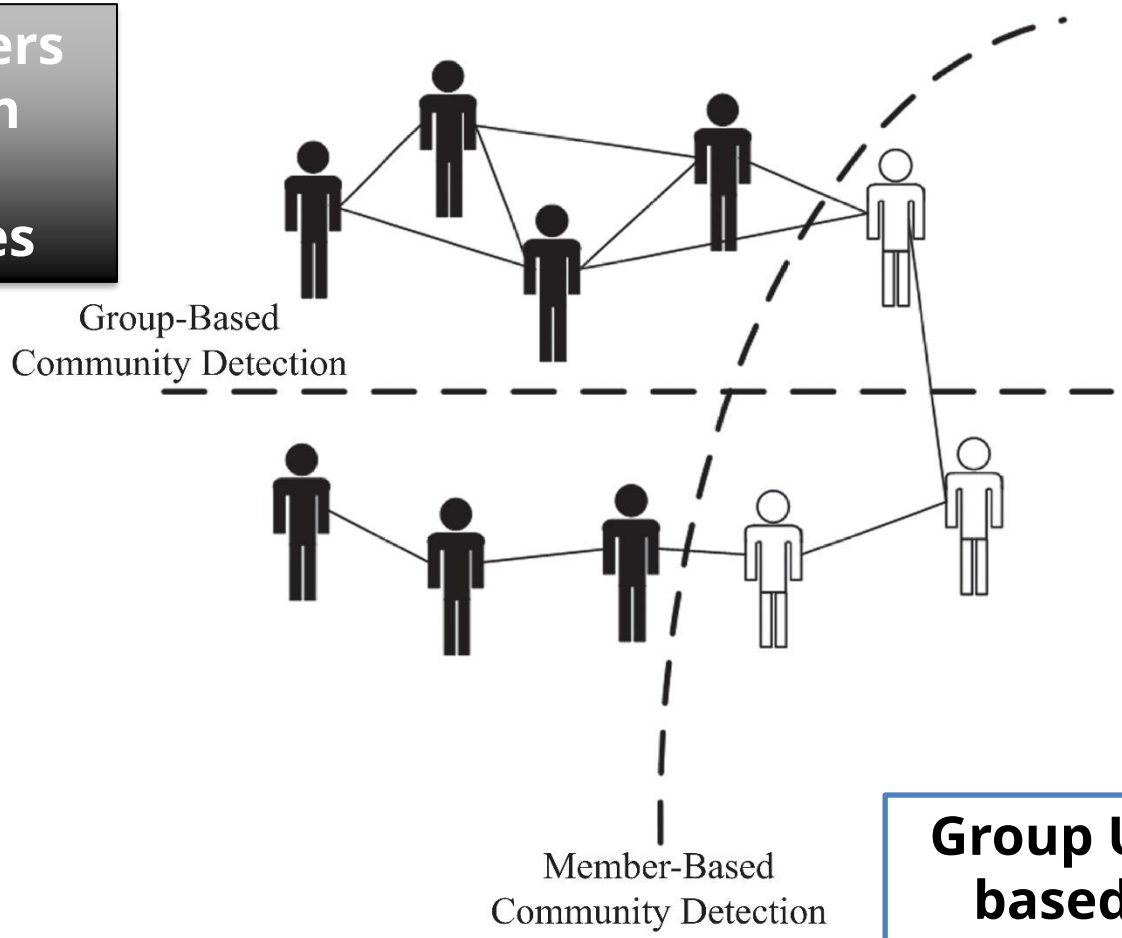
Community detection

- Data is linked (a graph)
- Network data tends to be "discrete", leading to algorithms using the graph property directly
  - k-clique, quasi-clique, or edge-betweenness

Community Detection Algorithms

innovate achieve lead

Group Users based on Group attributes



Group Users based on Member attributes

Member-Based Community Detection

Member-Based Community Detection

Look at node characteristics; and
Identify nodes with similar characteristics and consider them a community

Node Characteristics

A. Degree

Nodes with same (or similar) degrees are in one community
Example: cliques

B. Reachability

Nodes that are close (small shortest paths) are in one community
Example: k-cliques, k-clubs, and k-clans

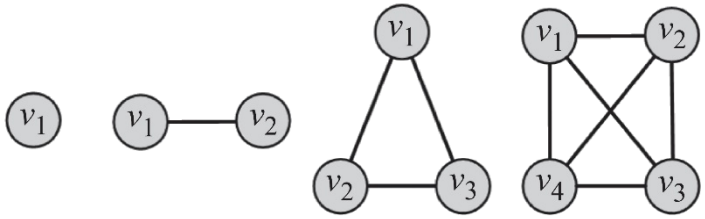
C. Similarity

Similar nodes are in the same community

A. Node Degree

innovate achieve lead

Most common subgraph searched for:
Clique: a maximum complete subgraph in which all nodes inside the subgraph adjacent to each other



Find communities by searching for

1. **The maximum clique**: **the one** with the largest number of vertices, or
2. **All maximal cliques**: cliques that are not subgraphs of a larger clique; i.e., cannot be further expanded

To overcome this, we can

I. Brute Force
II. Relax cliques
III. Use cliques as the core for larger communities

Both problems are NP-hard

I. Brute-Force Method

Can find all the maximal cliques in the graph

For each vertex v_x, we find the maximal clique that contains node v_x

Algorithm 1 Brute-Force Clique Identification

Require: Adjacency Matrix A, Vertex v_x

1: **return** Maximal Clique C containing v_x
2: CliqueStack = $\{\{v_x\}\}$, Processed = $\{\}$;
3: **while** CliqueStack not empty **do**
4: C=pop(CliqueStack); push(Processed,C);
5: v_{last} = Last node added to C;
6: $N(v_{last}) = \{v_i | A_{v_{last},v_i} = 1\}$.
7: **for all** $v_{temp} \in N(v_{last})$ **do**
8: **if** $C \bigcup \{v_{temp}\}$ is a clique **then**
9: push(CliqueStack, $C \bigcup \{v_{temp}\}$);
10: **end if**
11: **end for**
12: **end while**
13: Return the largest clique from Processed

Impractical for large networks:

- For a complete graph of only 100 nodes, the algorithm will generate at least $2^{99} - 1$ different cliques starting from any node in the graph

Enhancing the Brute-Force Performance

[Systematic] Pruning can help:

When searching for cliques of size k or larger

If the clique is found, each node should have a degree equal to or more than $k-1$

We can first prune all nodes (and edges connected to them) with degrees less than $k-1$

- More nodes will have degrees less than $k-1$
- Prune them recursively

For large k, many nodes are pruned as social media networks follow a power-law degree distribution

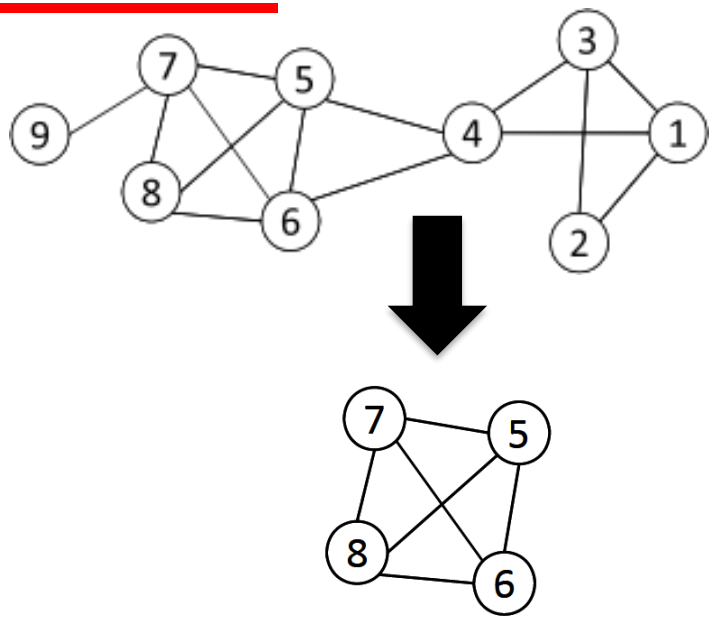
Maximum Clique: Pruning…

Example. to find a clique ≥ 4, remove all nodes with degree $\leq (4-1)-1=2$

Remove nodes 2 and 9

Remove nodes 1 and 3

Remove node 4



Even with pruning, cliques are less desirable

- Cliques are rare
- A clique of 1000 nodes, has 999x1000/2 edges
- A single edge removal destroys the clique
- That is less than 0.0002% of the edges!

II. Relaxing Cliques

k-plex: a set of vertices V in which we have

$$d_v \geq |V| - k, \forall v \in V$$

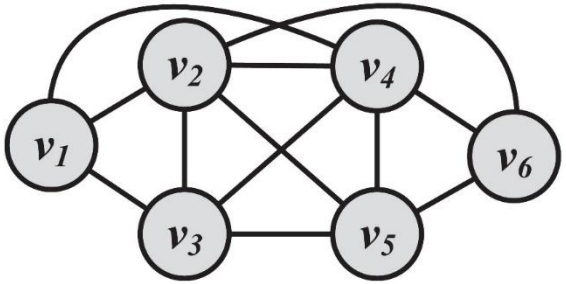
d_v is the degree of v in the induced subgraph

Number of nodes from V that are connected to v

Clique of size k is a 1-plex

Finding the maximum k-plex: **NP-hard**

In practice, relatively easier due to smaller search space.



1-plex : $\{v_2, v_3, v_4, v_5\}$

2-plex : $\{v_1, v_2, v_3, v_4, v_5\}, \{v_2, v_3, v_4, v_5, v_6\}$

3-plex : $\{v_1, v_2, v_3, v_4, v_5, v_6\}$

Maximal k-plexes

III. Using Cliques as a seed of a Community

Clique Percolation Method (CPM)

Uses cliques as seeds to find larger communities

CPM finds overlapping communities

Input

A parameter k, and a network

Procedure

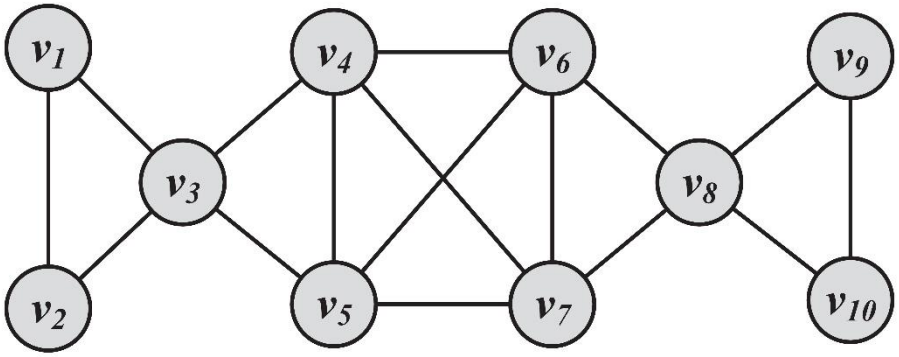
Find out all cliques of size k in the given network

Construct a clique graph.

Two cliques are adjacent if they share $k-1$ nodes

Each connected components in the clique graph form a community

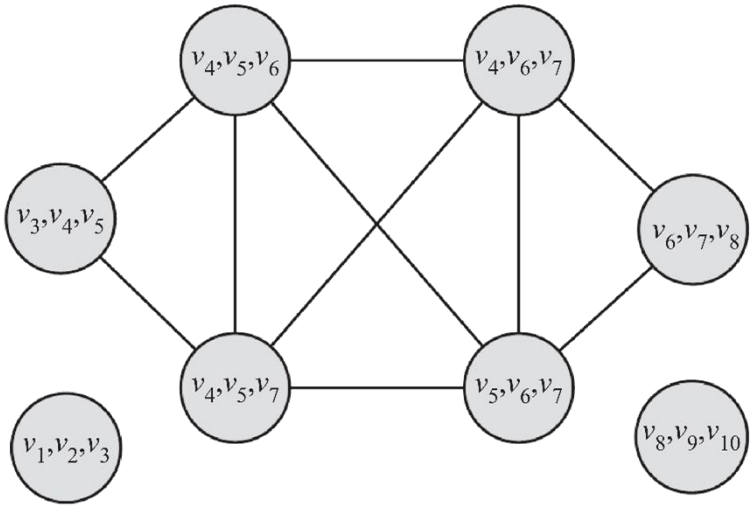
Clique Percolation Method: Example



(a) Graph

Cliques of size 3:

$\{v_1, v_2, v_3\}$, $\{v_3, v_4, v_5\}$,
$\{v_4, v_5, v_6\}$, $\{v_4, v_5, v_7\}$,
$\{v_4, v_6, v_7\}$, $\{v_5, v_6, v_7\}$,
$\{v_6, v_7, v_8\}$, $\{v_8, v_9, v_{10}\}$



(b) CPM Clique Graph

Communities:

$\{v_1, v_2, v_3\}$,
$\{v_8, v_9, v_{10}\}$,
$\{v_3, v_4, v_5, v_6, v_7, v_8\}$

B. Node Reachability

The two extremes

Nodes are assumed to be in the same community

1. If there is a path between them (regardless of the distance) or
2. They are so close as to be immediate neighbors.

How? Find using BFS/DFS

Challenge: most nodes are in one community (giant component)

How? Finding Cliques

Challenge: Cliques are challenging to find and are rarely observed

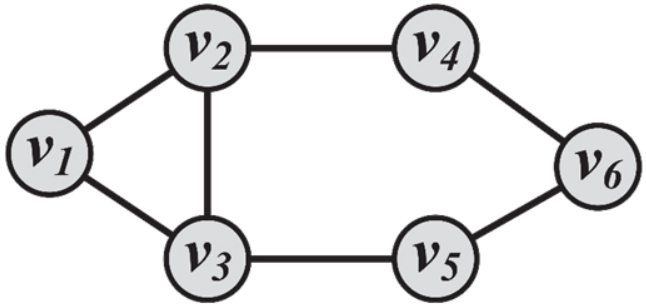
Solution: find communities that are in between cliques and connected components in terms of connectivity and have small shortest paths between their nodes

Special Subgraphs

1. k-**Clique**: a **maximal** subgraph in which the largest shortest path distance between any nodes is less than or equal to k

2. k-**Club**: follows the same definition as a k-clique
   Additional Constraint: nodes on the shortest paths should be part of the subgraph (i.e., diameter)

3. k-**Clan**: a k-clique where for all shortest paths within the subgraph the distance is equal or less than k.
   All k-clans are k-cliques, but not vice versa.



2-cliques : $\{v_1, v_2, v_3, v_4, v_5\}, \{v_2, v_3, v_4, v_5, v_6\}$

2-clubs : $\{v_2, v_3, v_4, v_5, v_6\}, \{v_1, v_2, v_3, v_4\}, \{v_1, v_2, v_3, v_5\}$

2-clans : $\{v_2, v_3, v_4, v_5, v_6\}$

C. Node Similarity

Similar (or most similar) nodes are assumed to be in the same community.

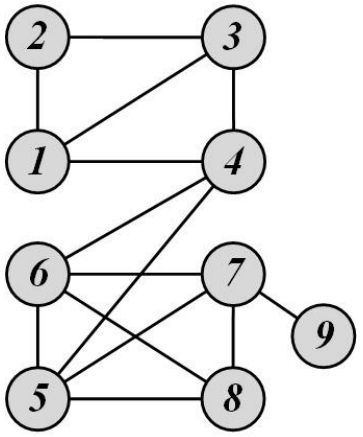
- A classical clustering algorithm (e.g., k-means) is applied to node similarities to find communities.

Node similarity can be defined

- Using the similarity of node neighborhoods (**Structural Equivalence**) – Ch. 3
- Similarity of social circles (**Regular Equivalence**) – Ch. 3

Structural equivalence: two nodes are structurally equivalent iff. they are connecting to the same set of actors

Nodes 1 and 3 are structurally equivalent, So are nodes 5 and 7.



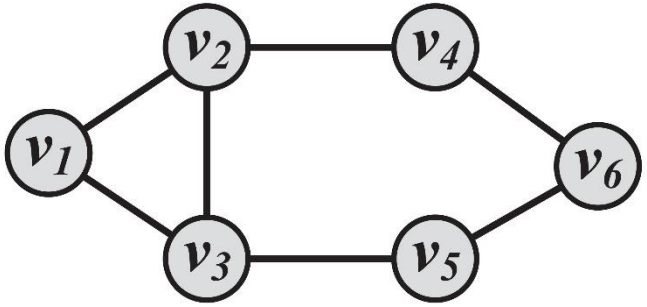
Node Similarity (Structural Equivalence)

Jaccard Similarity

$$\sigma_{\text{Jaccard}}(v_i, v_j) = \frac{|N(v_i) \cap N(v_j)|}{|N(v_i) \cup N(v_j)|}$$

Cosine similarity

$$\sigma_{\text{Cosine}}(v_i, v_j) = \frac{|N(v_i) \cap N(v_j)|}{\sqrt{|N(v_i)||N(v_j)|}}$$



$$\sigma_{\text{Jaccard}}(v_2, v_5) = \frac{|\{v_1, v_3, v_4\} \cap \{v_3, v_6\}|}{|\{v_1, v_3, v_4, v_6\}|} = 0.25$$

$$\sigma_{\text{Cosine}}(v_2, v_5) = \frac{|\{v_1, v_3, v_4\} \cap \{v_3, v_6\}|}{\sqrt{|\{v_1, v_3, v_4\}||\{v_3, v_6\}|}} = 0.40$$

Group-Based Community Detection

Group-Based Community Detection

Group-based community detection: finding communities that have certain **group properties**

Group Properties:

I. **Balanced:** Spectral clustering
II. **Robust:** k-connected graphs
III. **Modular:** Modularity Maximization
IV. **Dense:** Quasi-cliques
V. **Hierarchical:** Hierarchical clustering

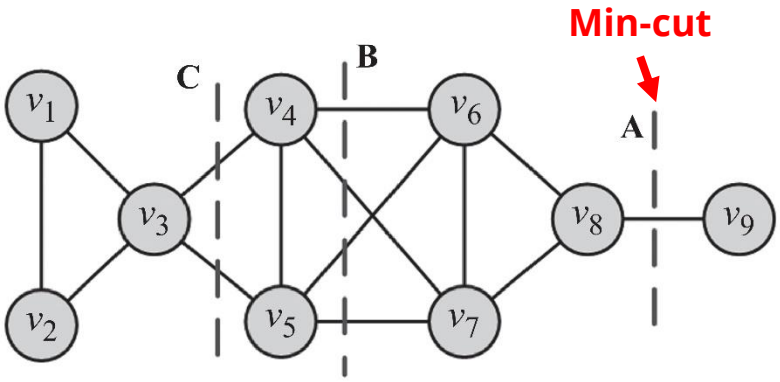
I. Balanced Communities

Community detection can be thought of *graph clustering*

Graph clustering: we cut the graph into several partitions and assume these partitions represent communities

Cut: partitioning (*cut*) of the graph into two (or more) sets (*cutsets*)
The size of the cut is the number of edges that are being cut

Minimum cut (min-cut) problem: find a graph partition such that the number of edges between the two sets is minimized



Min-cuts can be computed efficiently using the max-flow min-cut theorem

Min-cut often returns an imbalanced partition, with one set being a singleton

Ratio Cut and Normalized Cut

To mitigate the min-cut problem we can change the objective function to consider community size

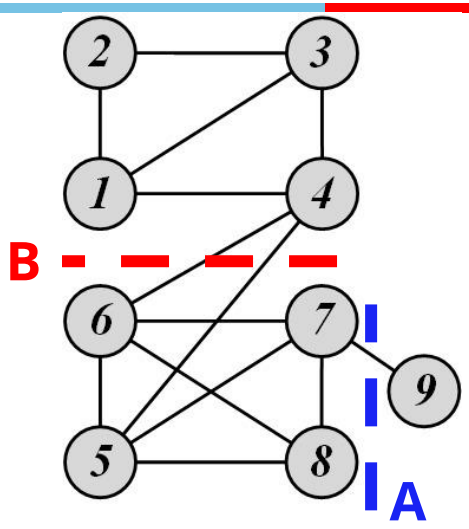
$$\text{Ratio Cut}(P) = \frac{1}{k}\sum_{i=1}^{k}\frac{\text{cut}(P_i,\bar{P}_i)}{|P_i|}$$

$$\text{Normalized Cut}(P) = \frac{1}{k}\sum_{i=1}^{k}\frac{\text{cut}(P_i,\bar{P}_i)}{\text{vol}(P_i)}$$

$\bar{P}_i = V - P_i$ is the complement cut set
$cut(P_i, \bar{P}_i)$ is the size of the cut

$$vol(P_i) = \sum_{v \in P_i} d_v$$

Ratio Cut & Normalized Cut: Example



For Cut A

$$\text{Ratio Cut}(\{1,2,3,4,5,6,7,8\},\{9\}) = \tfrac{1}{2}(\tfrac{1}{1}+\tfrac{1}{8}) = 9/16 = 0.56$$

$$\text{Normalized Cut}(\{1,2,3,4,5,6,7,8\},\{9\}) = \tfrac{1}{2}(\tfrac{1}{1}+\tfrac{1}{27}) = 14/27 = 0.52$$

For Cut B

$$\text{Ratio Cut}(\{1,2,3,4\},\{5,6,7,8,9\}) = \tfrac{1}{2}(\tfrac{2}{4}+\tfrac{2}{5}) = 9/20 = 0.45 < 0.56$$

$$\text{Normalized Cut}(\{1,2,3,4\},\{5,6,7,8,9\}) = \tfrac{1}{2}(\tfrac{2}{12}+\tfrac{2}{16}) = 7/48 = 0.15 < 0.52$$

Both ratio cut and normalized cut prefer a balanced partition

Divisive Hierarchical Clustering

innovate achieve lead

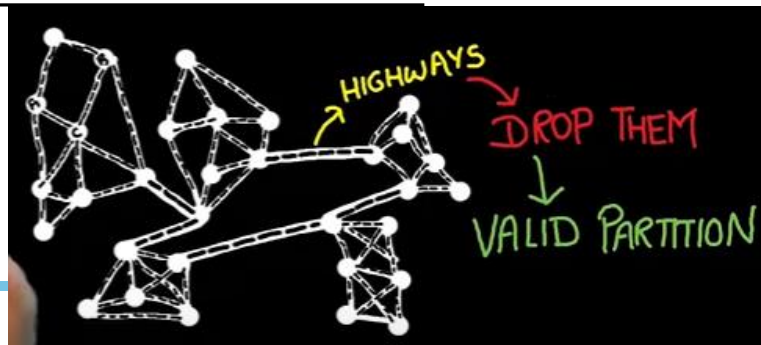
- Divisive clustering
  - Partition nodes into several sets
  - Each set is further divided into smaller ones
  - Network-centric partition can be applied for the partition
- One particular example:

Girvan-Newman Algorithm: recursively remove the "weakest" links within a "community"

- Find the edge with the weakest link
- Remove the edge and update the corresponding strength of each edge

Recursively apply the above two steps until a network is discomposed into a desired number of connected components.

Each component forms a community



Edge Betweenness

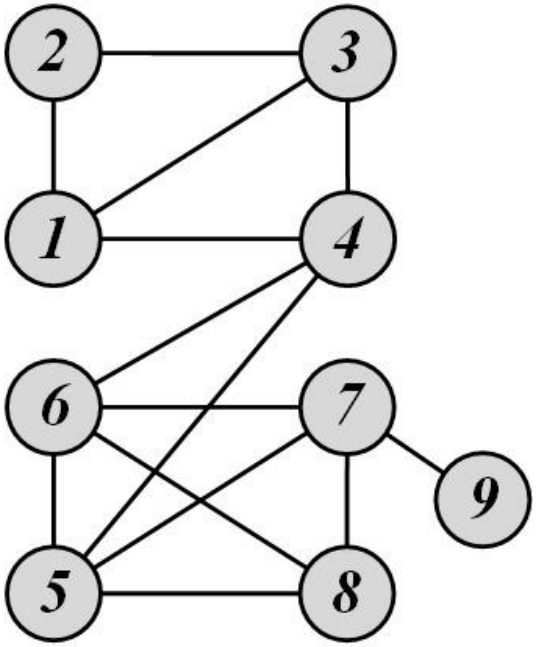
To determine weakest links, the algorithm uses **edge betweenness**.

Edge betweenness is the number of shortest paths that pass along with the edge

Edge betweenness measures the “bridgeness” of an edge between two communities

The edge with high betweenness tends to be the bridge between two communities.

Edge Betweenness: Example

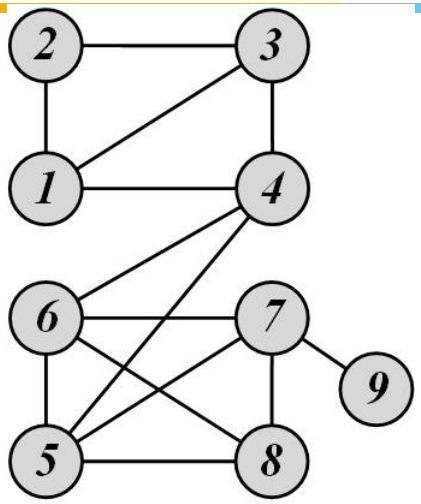


The edge betweenness of $e(1,2)$ is $6/2 + 1 = 4$, as all the shortest paths from 2 to $\{4,5,6,7,8,9\}$ have to either pass $e(1,2)$ or $e(2,3)$, and $e(1,2)$ is the shortest path between 1 and 2

The Girvan-Newman Algorithm

1. Calculate edge betweenness for all edges in the graph.
2. Remove the edge with the highest betweenness.
3. Recalculate betweenness for all edges affected by the edge removal.
4. Repeat until all edges are removed.

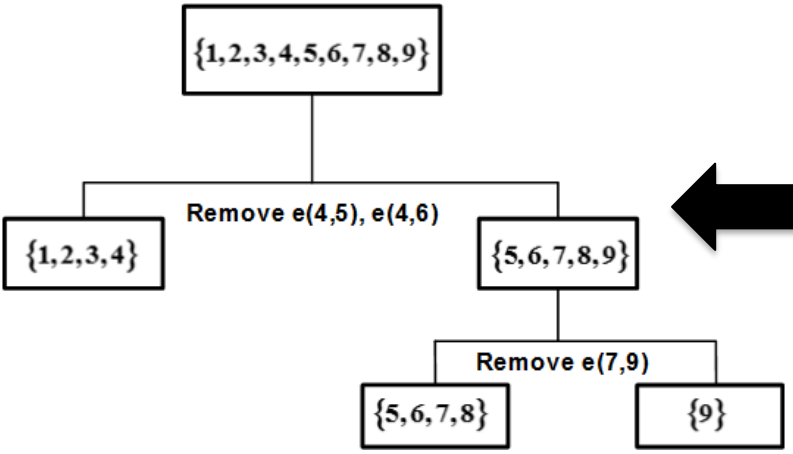
Edge Betweenness Divisive Clustering: Example



Initial betweenness value

	1	2	3	4	5	6	7	8	9
1	0	4	1	9	0	0	0	0	0
2	4	0	4	0	0	0	0	0	0
3	1	4	0	9	0	0	0	0	0
4	9	0	9	0	10	10	0	0	0
5	0	0	0	10	0	1	6	3	0
6	0	0	0	10	1	0	6	3	0
7	0	0	0	0	6	6	0	2	8
8	0	0	0	0	3	3	2	0	0
9	0	0	0	0	0	0	8	0	0

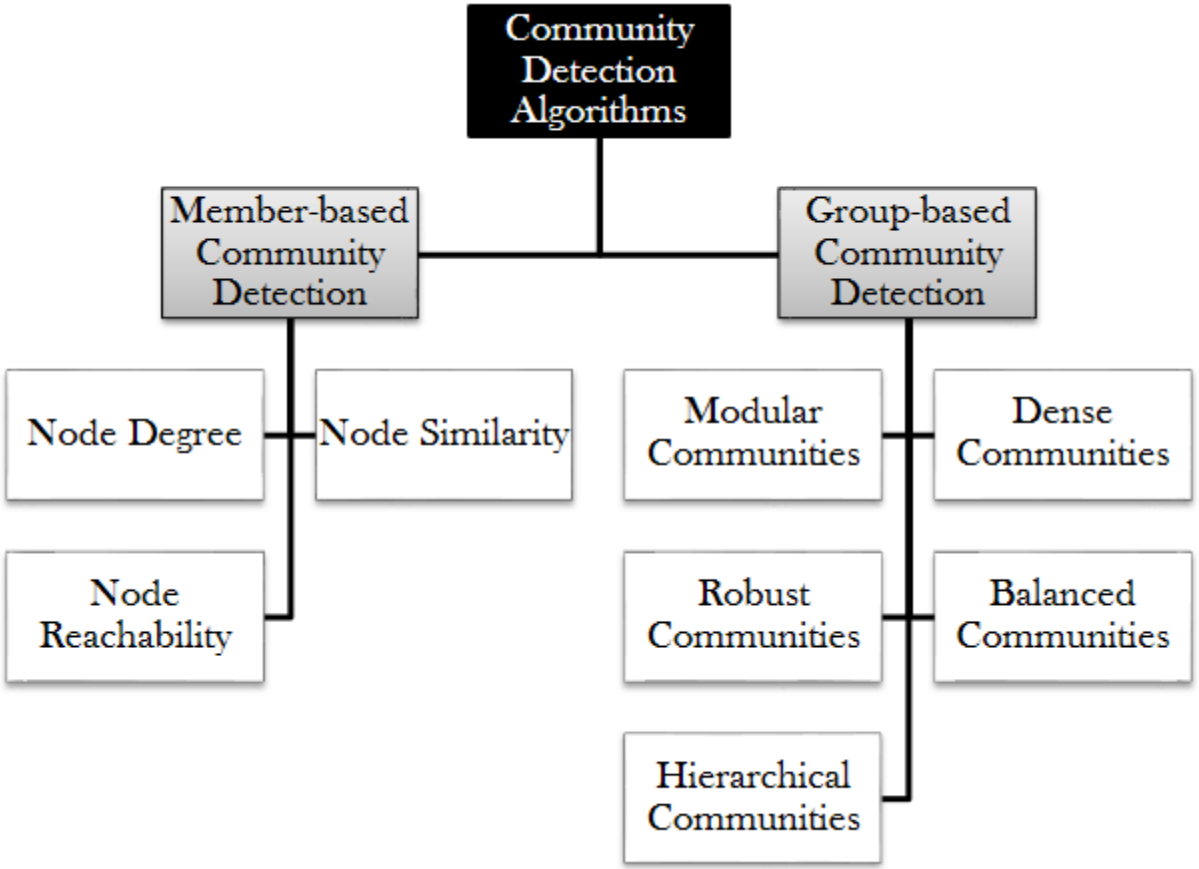
the first edge that needs to be removed is e(4, 5) (or e(4, 6))



By removing e(4, 5), we compute the edge betweenness once again; this time, e(4, 6) has the highest betweenness value: 20.

This is because all shortest paths between nodes {1,2,3,4} to nodes {5,6,7,8,9} must pass e(4, 6); therefore, it has betweenness 4×5 = 20.

Community Detection Algorithms



How Networks Evolve?

Network Growth Patterns

1. Network Segmentation
2. Graph Densification
3. Diameter Shrinkage

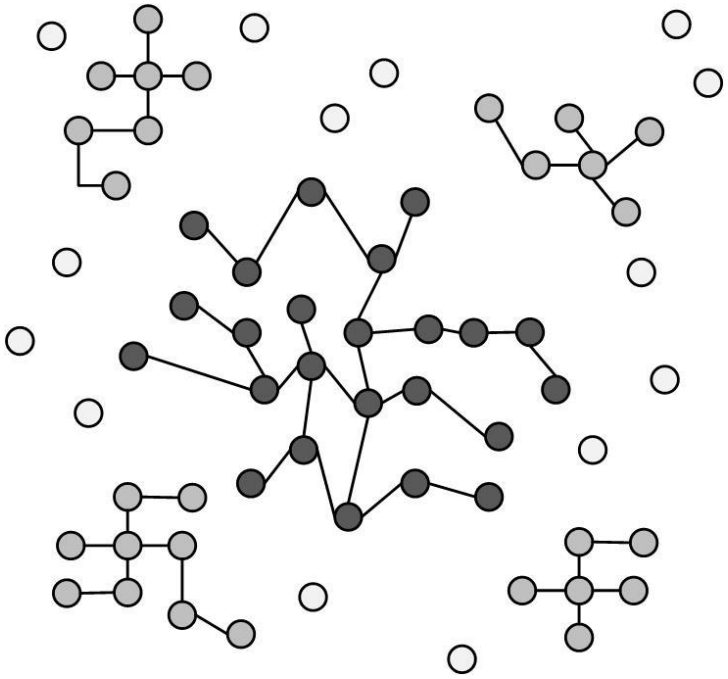
1. Network Segmentation

Often, in evolving networks, segmentation takes place, where the large network is decomposed over time into three parts

1. **Giant Component**: As network connections stabilize, a giant component of nodes is formed, with a large proportion of network nodes and edges falling into this component.

2. **Stars**: These are isolated parts of the network that form star structures. A star is a tree with one internal node and n leaves.

3. **Singletons**: These are orphan nodes disconnected from all nodes in the network.



2. Graph Densification

The density of the graph increases as the network grows

- The number of edges increases faster than the number of nodes does

$$E(t) \propto V(t)^{\alpha}$$

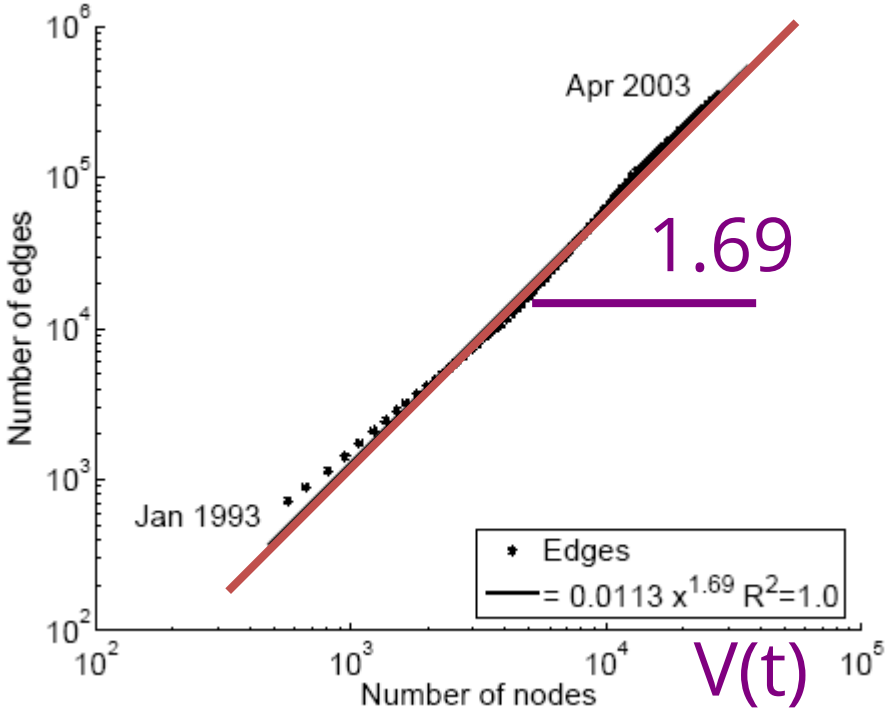
Densification exponent: $1 \leq \alpha \leq 2$:

- $\alpha = 1$: linear growth – constant out-degree
- $\alpha = 2$: quadratic growth – clique

$E(t)$ and $V(t)$ are numbers of edges and nodes respectively at time t

Densification in Real Networks

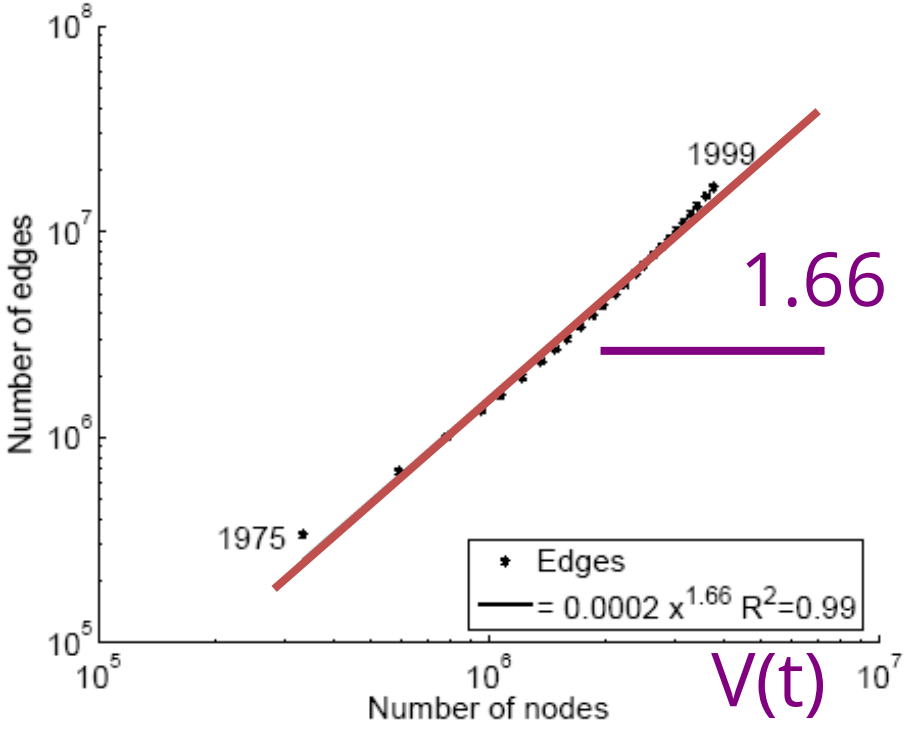
E(t)



V(t)

Physics Citations

E(t)



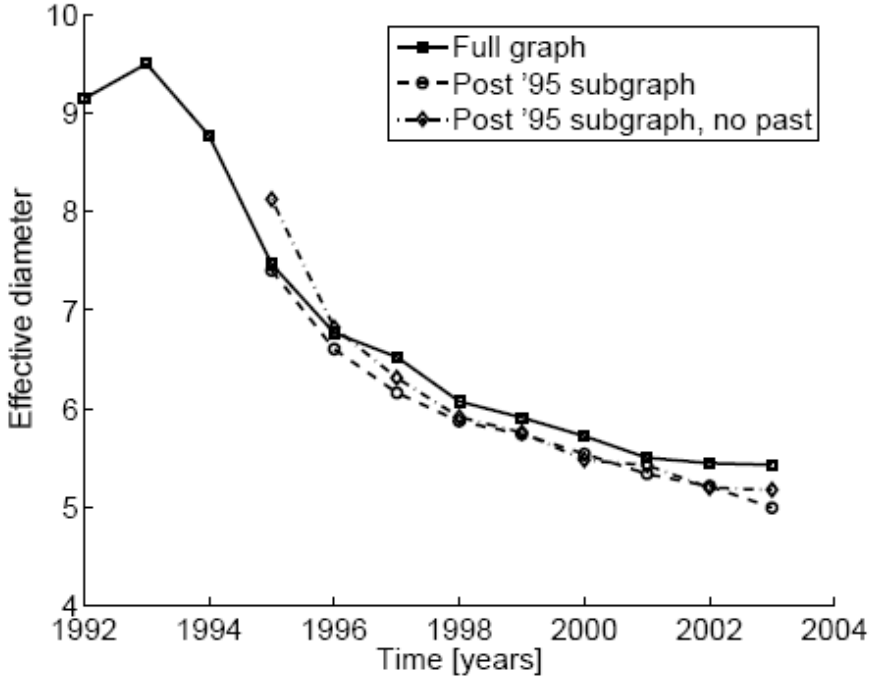
V(t)

Patent Citations

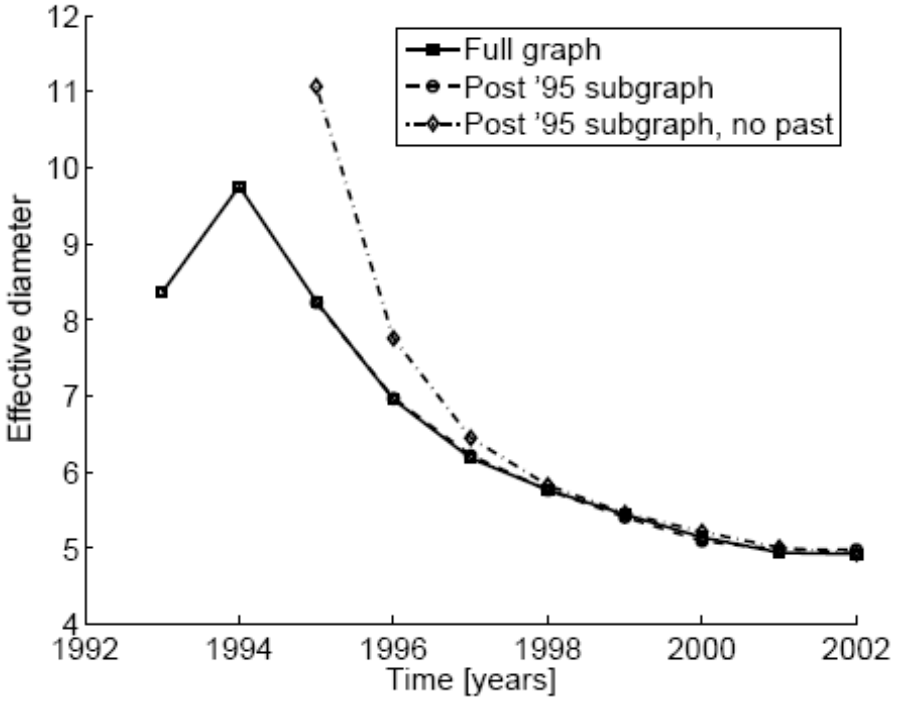
3. Diameter Shrinking

innovate achieve lead

In networks diameter shrinks over time



ArXiv citation graph

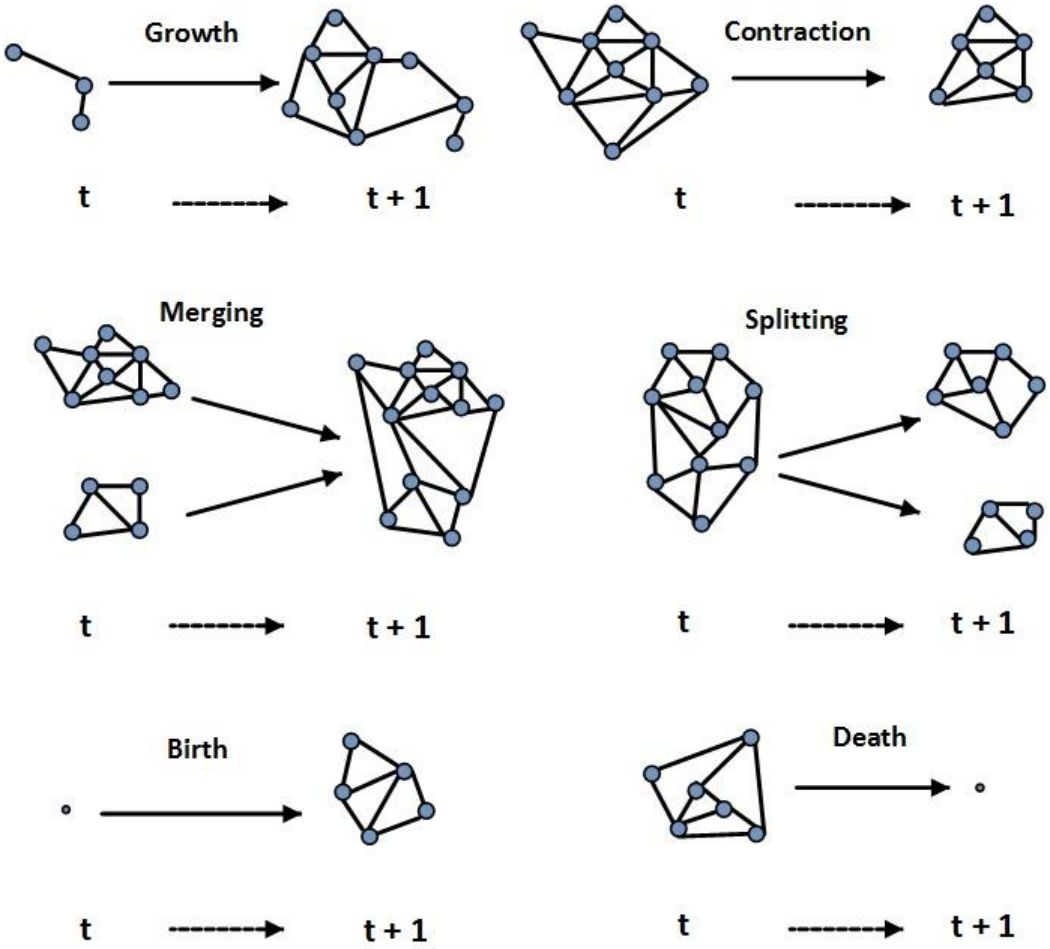


Affiliation Network

How Communities Evolve?

Community Evolution

Communities also expand, shrink, or dissolve in dynamic networks

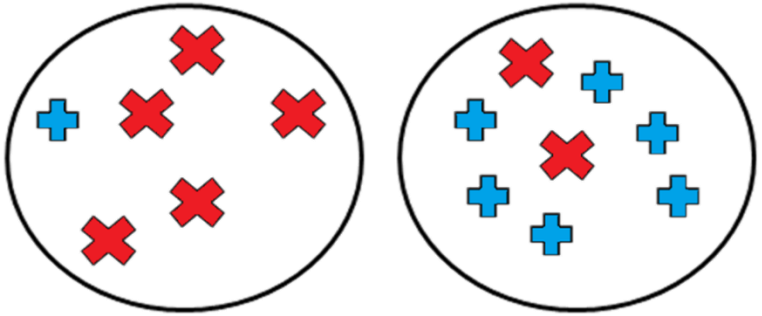


Community Evaluation

Evaluating the Communities

We are given objects of two different kinds (+, ×)

- **The perfect community:** all objects inside the community are of the same type



Evaluation with ground truth
Evaluation without ground truth

Evaluation with Ground Truth

When ground truth is available

- We have partial knowledge of what communities should look like
- We are given the correct community (clustering) assignments

Measures

- Precision and Recall, or F-Measure
- Purity
- Normalized Mutual Information (NMI)

Precision and Recall

$$Precision = \frac{Relevant\ and\ retrieved}{Retrieved}$$

$$P = \frac{TP}{TP + FP}$$

$$Recall = \frac{Relevant\ and\ retrieved}{Relevant}$$

$$R = \frac{TP}{TP + FN}$$

True Positive (TP) :

- When similar members are assigned to the same communities
- A **correct** decision.

True Negative (TN) :

- When dissimilar members are assigned to different communities
- A **correct** decision

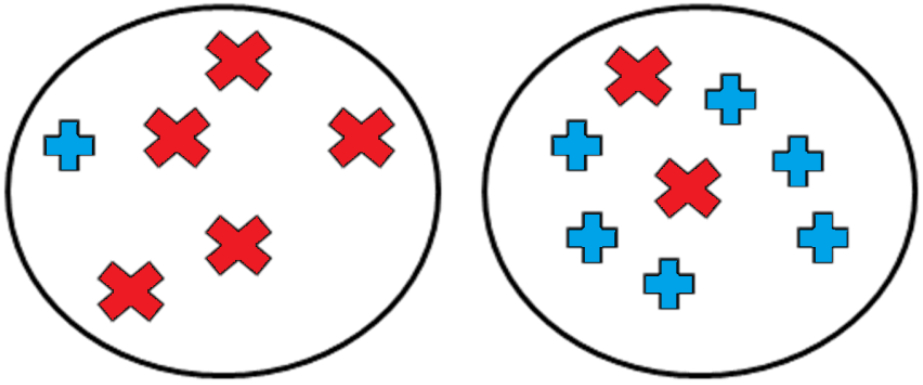
False Negative (FN) :

- When similar members are assigned to different communities
- An **incorrect** decision

False Positive (FP) :

- When dissimilar members are assigned to the same communities
- An **incorrect** decision

Precision and Recall: Example



$$TP = \binom{5}{2} + \binom{6}{2} + \binom{2}{2} = 26,$$

$$FP = (5 \times 1) + (6 \times 2) = 17,$$

$$FN = (5 \times 2) + (6 \times 1) = 16,$$

$$TN = (6 \times 5) + (2 \times 1) = 32.$$

For TP, we need to compute the number of pairs with the same label that are in the same community.
For instance, for label X and cluster 1, we have ${}^5C_2$ such pairs.
For FP, we need to compute dissimilar pairs that are in the same community.
FN computes similar members that are in different communities.
TN computes the number of dissimilar pairs in dissimilar communities

$$P = \frac{26}{26+17} = 0.60$$

$$R = \frac{26}{26+16} = 0.61$$

F-Measure

Either P or R measures one aspect of the performance,

To integrate them into one measure, we can use the harmonic mean of precision of recall

$$F = 2 \cdot \frac{P \cdot R}{P + R}$$

For the example earlier,

$$F = 2 \times \frac{0.6 \times 0.61}{0.6 + 0.61} = 0.60$$

Purity

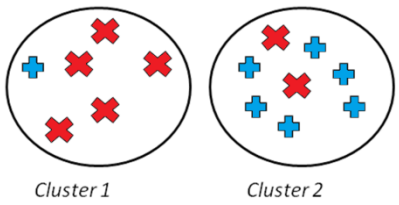
We can assume the majority of a community represents the community

We use the label of the majority against the label of each member to evaluate the communities

Purity. The fraction of instances that have labels equal to the community's majority label

$$Purity = \frac{1}{N}\sum_{i=1}^{k}\max_{j}|C_i \cap L_j|$$

- k: the number of communities
- N: total number of nodes,
- L_j: the set of instances with label j in all communities
- C_i: the set of members in community i



$purity\ is: \frac{6+5}{14} = 0.78$

Purity can be easily **tampered** by

- Points being singleton communities (of size 1); or by
- Very large communities

Mutual Information

innovate achieve lead

Mutual information (MI). The amount of information that two random variables share.

By knowing one of the variables, it measures the amount of uncertainty reduced regarding the others

$$MI = I(H, L) = \sum_{h \in H} \sum_{l \in L} \frac{n_{h,l}}{n} \log \frac{n \cdot n_{h,l}}{n_h n_l}$$

- L and H are labels and found communities;
- n_h and n_l are the number of data points in community h and with label l, respectively;
- $n_{h\ l}$ is the number of nodes in community h and with label l; and n is the number of nodes

Normalizing Mutual Information (NMI)

Mutual information (MI) is unbounded
To address this issue, we can normalize MI

How? We know that

$$MI \leq min(H(L), H(H)),$$
$$(MI)^2 \leq H(H)H(L).$$
$$MI \leq \sqrt{H(H)}\sqrt{H(L)}.$$

$H(.)$ is the entropy function

$$H(L) = -\sum_{l \in L} \frac{n_l}{n} \log \frac{n_l}{n}$$
$$H(H) = -\sum_{h \in H} \frac{n_h}{n} \log \frac{n_h}{n}.$$

Normalized Mutual Information

Normalized Mutual Information

$$NMI = \frac{MI}{\sqrt{H(L)}\sqrt{H(H)}}.$$

$$NMI = \frac{\sum_{h \in H} \sum_{l \in L} n_{h,l} \log \frac{n \cdot n_{h,l}}{n_h n_l}}{\sqrt{(\sum_{h \in H} n_h \log \frac{n_h}{n})(\sum_{l \in L} n_l \log \frac{n_l}{n})}}.$$

We can also define it as

Note that $MI < 1/2(H(H) + H(L)$

$$NMI = \frac{I(H;L)}{\frac{1}{2}(H(L) + H(H))}$$

Normalized Mutual Information

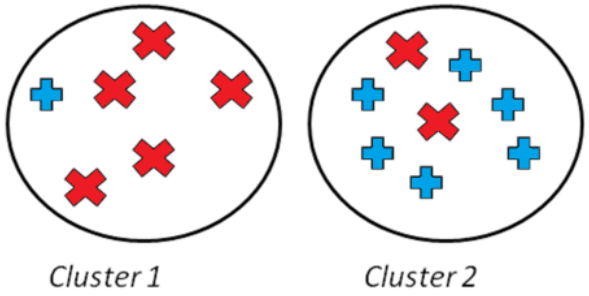
$$NMI = \frac{\sum_{h,l} n_{h,l} \log \frac{n \cdot n_{h,l}}{n_h n_l}}{\sqrt{(\sum_h n_h \log \frac{n_h}{n})(\sum_l n_l \log \frac{n_l}{n})}}$$

- where l and h are known (with labels) and found communities, respectively
- n_h and n_l are the number of members in the community h and l, respectively,
- $n_{h\ l}$ is the number of members in community h and labeled l,
- n is the size of the dataset

NMI values close to one indicate high similarity between communities found and labels

Values close to zero indicate high dissimilarity between them

Normalized Mutual Information: Example



Found communities (H)

[1,1,1,1,1,1, 2,2,2,2,2,2,2,2]

Actual Labels (L)

– [2,1,1,1,1,1, 2,2,2,2,2,2,1,1]

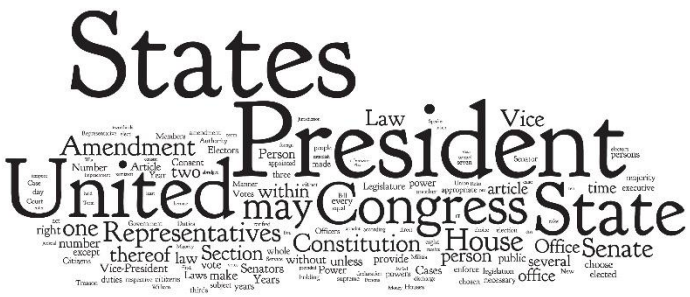
$n = 14$

	$\mathbf{n}_h$
h=1	6
h=2	8

	$\mathbf{n}_l$
$l = 1$	7
$l = 2$	7

$\mathbf{n}_{h,l}$	$l = 1$	$l = 2$
h=1	5	1
h=2	2	6

Evaluation without Ground Truth



(a) U.S . Constitution

(b) Sports

Evaluation with Semantics

A simple way of analyzing detected communities is to analyze other attributes (posts, profile information, content generated, etc.) of community members to see if there is a coherency among community members

The coherency is often checked via human subjects.

Or through labor markets: Amazon Mechanical Turk

To help analyze these communities, one can use word frequencies. By generating a list of frequent keywords for each community, human subjects determine whether these keywords represent a coherent topic.

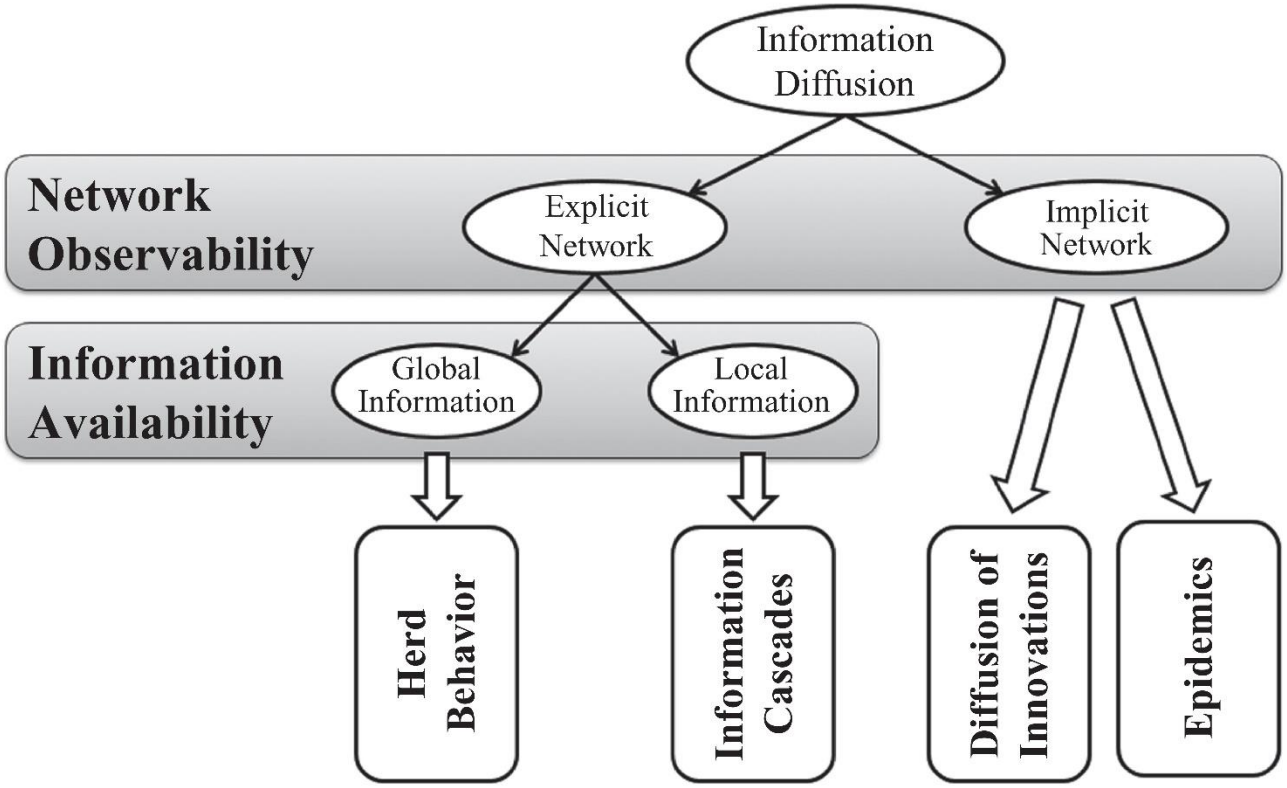
Evaluation Using Clustering Quality Measures

Use clustering quality measures (SSE)

Use more than two community detection algorithms and compare the results and pick the algorithm with better quality measure

Information Diffusion

Information Diffusion Types



We define the process of interfering with information diffusion by expediting, delaying, or even stopping diffusion as Intervention

Herd Behavior

- Network is observable
- Only public information is available

Herd Behavior: Popular Restaurant Example

Assume you are on a trip in a metropolitan area that you are less familiar with.

Planning for dinner, you find restaurant **A** with excellent reviews online and decide to go there.

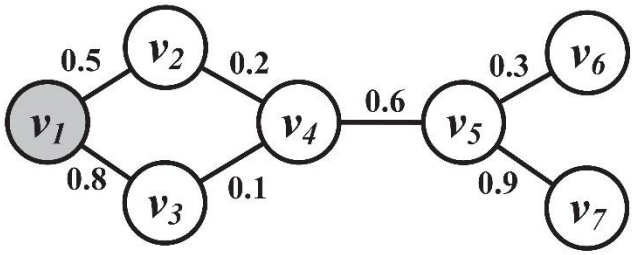
When arriving at **A**, you see **A** is almost empty and restaurant **B**, which is next door and serves the same cuisine, almost full.

Deciding to go to **B**, based on the belief that other diners have also had the chance of going to **A**, is an example of herd behavior

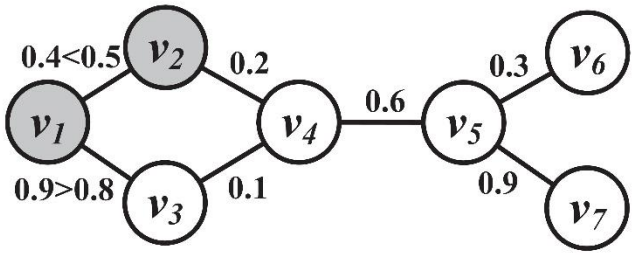
Information Cascade

- In the presence of a network
- Only local information is available

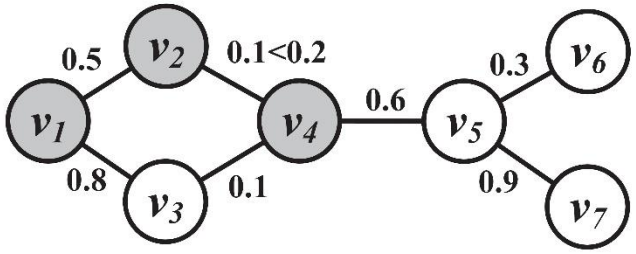
Independent Cascade Model: An Example



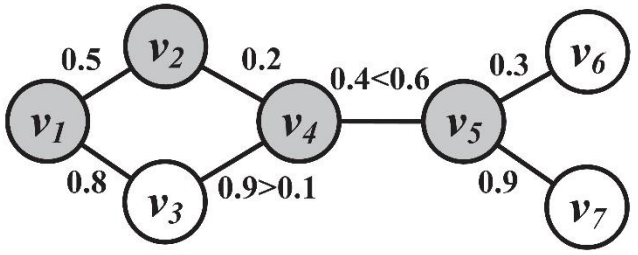
Step 1



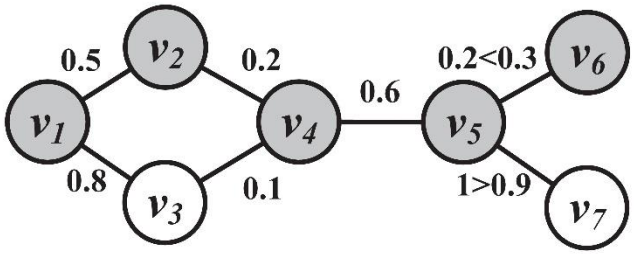
Step 2



Step 3



Step 4

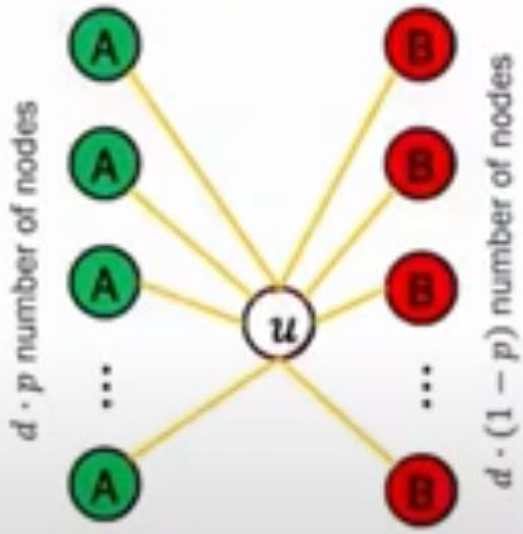


Step 5

Decision-based Cascade Model

innovate achieve lead

Decision-based Cascade Model: Two-player Coordination Game



- Node u has d neighbours
  - p fraction of neighbours adopt strategy A
  - Rest adopts strategy B
- Total payoff for node u if it goes with strategy A $= a \cdot d \cdot p$
- Total payoff for node u if it goes with strategy B $= b \cdot d \cdot (1-p)$
- Node u would adopt contagion A if

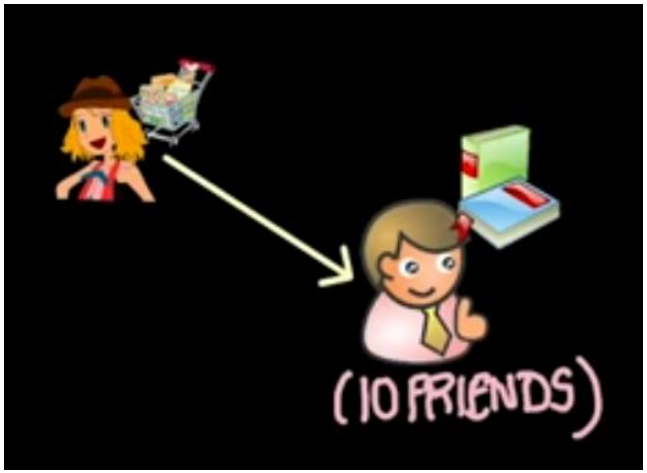
$$p \geq \frac{b}{a+b}$$

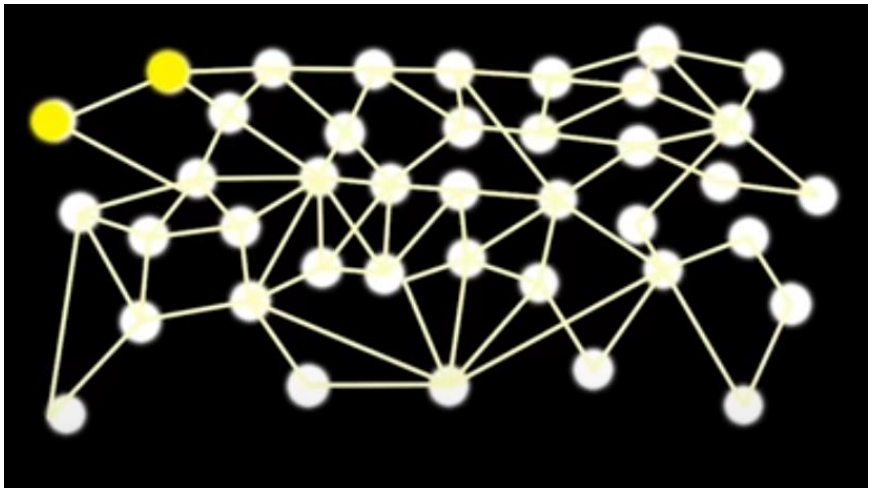
Source: [Tanmoy Chakraborty]

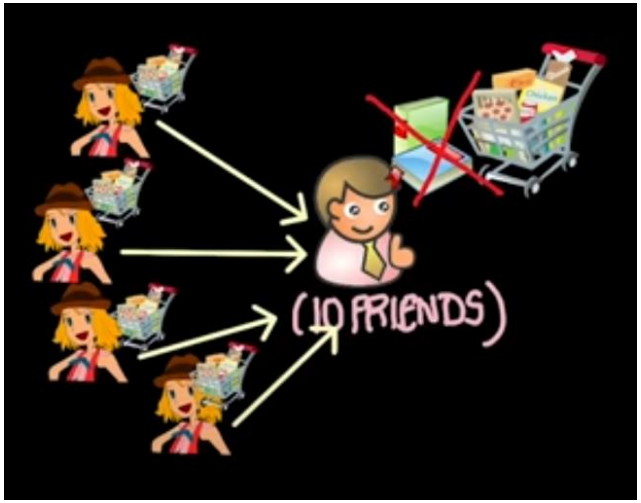
Example

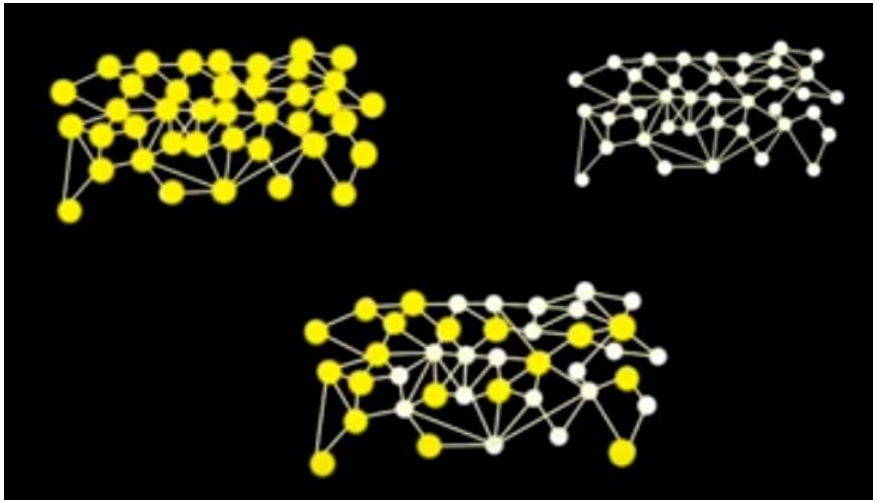
Source: SRS Iyengar, NPTEL / IIT Ropar

Diffusion through networks



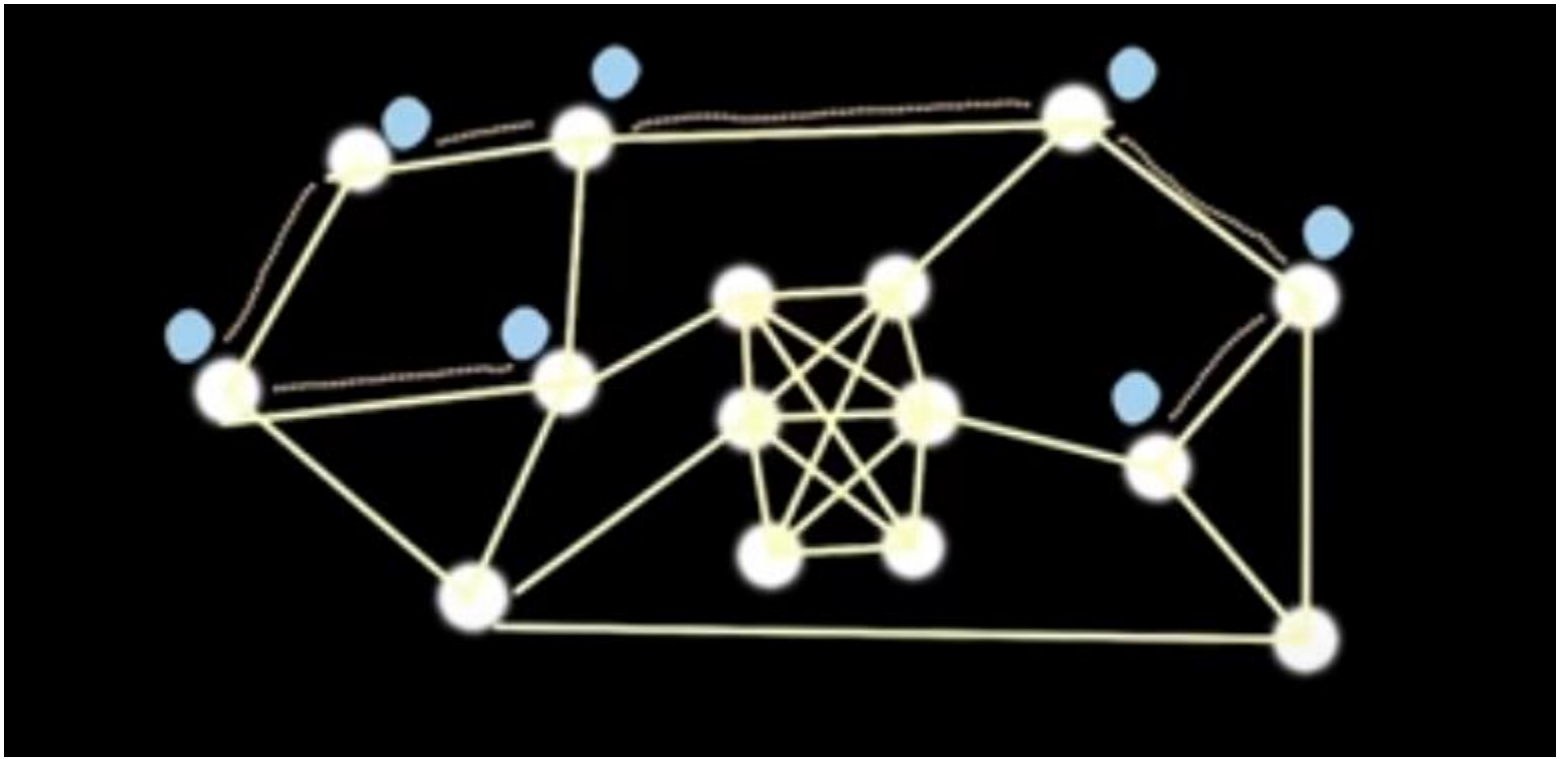






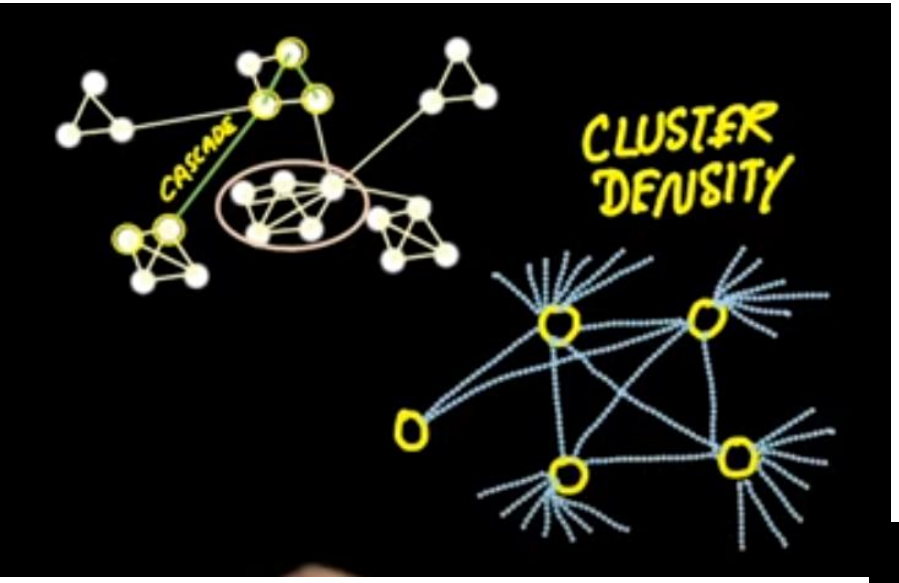
Source: SRS Iyengar, NPTEL / IIT Ropar

Impact of communities on diffusion

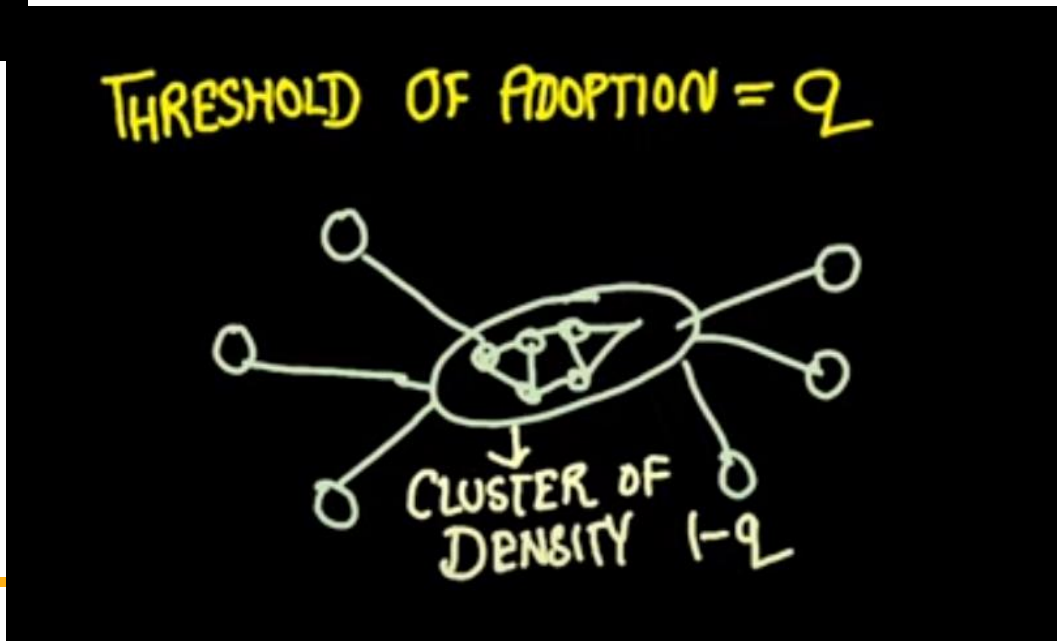


Source: SRS Iyengar, NPTEL / IIT Ropar

Cascade cannot complete if…

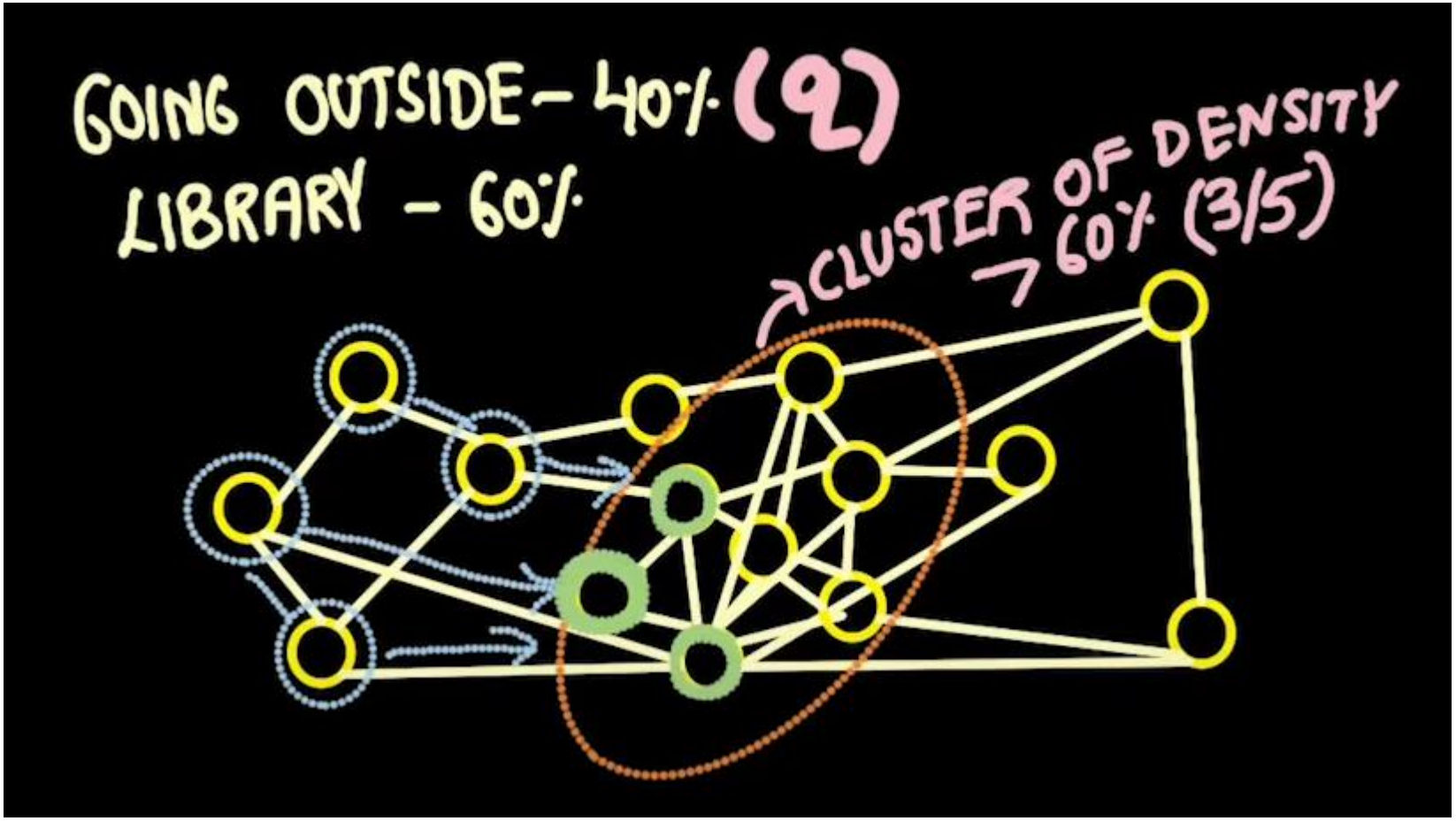


Cluster Density is ‘q’ if q fraction of nodes connected to each node lies within the cluster



Source: SRS Iyengar, NPTEL / IIT Ropar

Example of a community that prevents information cascades



Source: SRS Iyengar, NPTEL / IIT Ropar

Diffusion of Innovations

- The network is not observable
- Only public information is observable

Diffusion of Innovations Models

- First model was introduced by Gabriel Tarde in the early 20th century

I. The Iowa Study of Hybrid Corn Seed

Farmers received information through two main channels

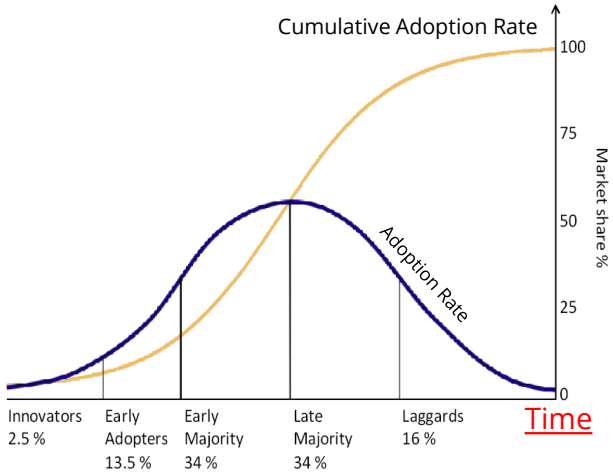
Mass communications from companies selling the seeds (i.e., information)

Interpersonal communications with other farmers. (i.e., influence)

Adoption depended on a combination of information and influence.

The study showed that the adoption rate follows an *S*-shaped curve and that there are 5 different types of adopters based on the order that they adopt the innovations, namely:

1) **Innovators** (top **2.5%**)
2) **Early Adopters** (next **13.5%**)
3) **Early Majority** (next **34%**)
4) **Late Majority** (next **34%**)
5) **Laggards** (last **16%**)

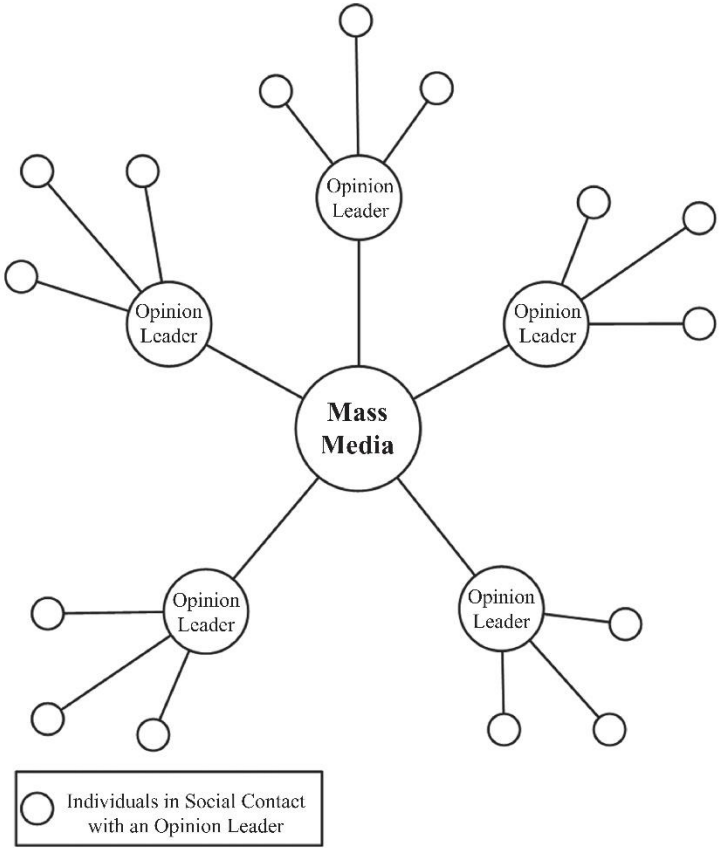


II. Katz Two-Step Flow Model

innovate achieve lead

Two-step Flow Model. most information comes from mass media, which is then directed toward influential figures called *opinion leaders*.

These leaders then convey the information (or form opinions) and act as hub for other members of the society



III. Rogers: Diffusion of Innovations: The Process

Adoption process:

1. **Awareness**
   The individual becomes aware of the innovation, but her information regarding the product is limited

2. **Interest**
   The individual shows interest in the product and seeks more information

3. **Evaluation**
   The individual tries the product in his mind and decides whether or not to adopt it

4. **Trial**
   The individual performs a trial use of the product

5. **Adoption**
   The individual decides to continue the trial and adopts the product for full use

Modeling Diffusion of Innovations

The diffusion of innovations models can be concretely described as

$$\frac{dA(t)}{dt} = i(t)[P - A(t)]$$

- $A(t)$ is the total population that adopted the innovation
- $i(t)$ denotes the coefficient of diffusion corresponding to the innovativeness of the product being adopted
- P is the total number of potential adopters (till time t)

- The rate depends on how innovative the product is
- The rate affects the potential adopters that have not yet adopted the product.

Modeling Diffusion of Innovations

We can rewrite $A(t)$ as

$$A(t) = \int_{t_0}^{t} a(t)dt$$

Let $A_0 = A(0)$

The adopters at time t

We can define the diffusion coefficient $i(t)$ as a function of number of adopters $A(t)$

$$i(t) = \alpha + \alpha_0 A_0 + \ldots + \alpha_t A(t) = \alpha + \sum_{i=t_0}^{t} \alpha_i A(i)$$

We can simplify this linear combination

Diffusion Models

innovate achieve lead

Three models of diffusion:
i.e., each having different ways to compute i(t):

$$\frac{dA(t)}{dt} = i(t)[P - A(t)]$$

$$i(t) = \alpha, \qquad \text{External-Influence Model}$$
$$i(t) = \beta A(t), \qquad \text{Internal-Influence Model}$$
$$i(t) = \alpha + \beta A(t). \qquad \text{Mixed-Influence Model}$$

- α - **Innovativeness factor of the product**
- β - **Imitation factor**

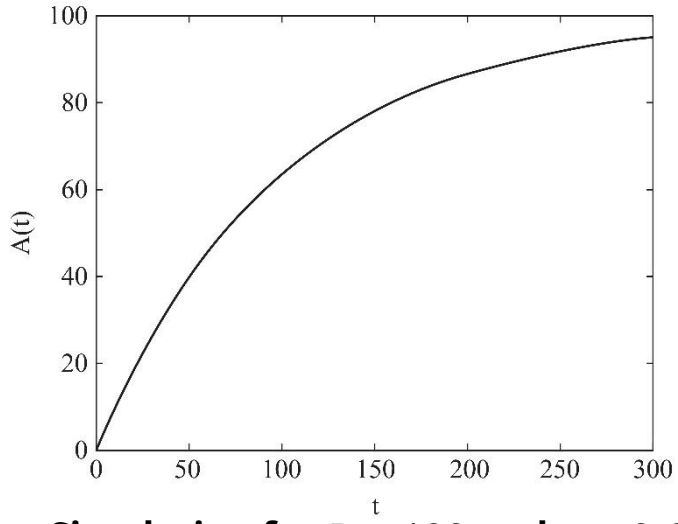
1. External-Influence Model

The adoption rate is a function that depends on external entities, $i(t) = \alpha$

Assuming $A(t = t_0 = 0) = 0$

$$\frac{dA(t)}{dt} = \alpha[P - A(t)] \quad \Rightarrow \quad A(t) = P(1 - e^{-\alpha t})$$

The number of adopters increases exponentially and then saturates near P



Simulation for $P = 100$ and $\alpha = 0.01$

2. Internal-Influence Model (Pure imitation Model)

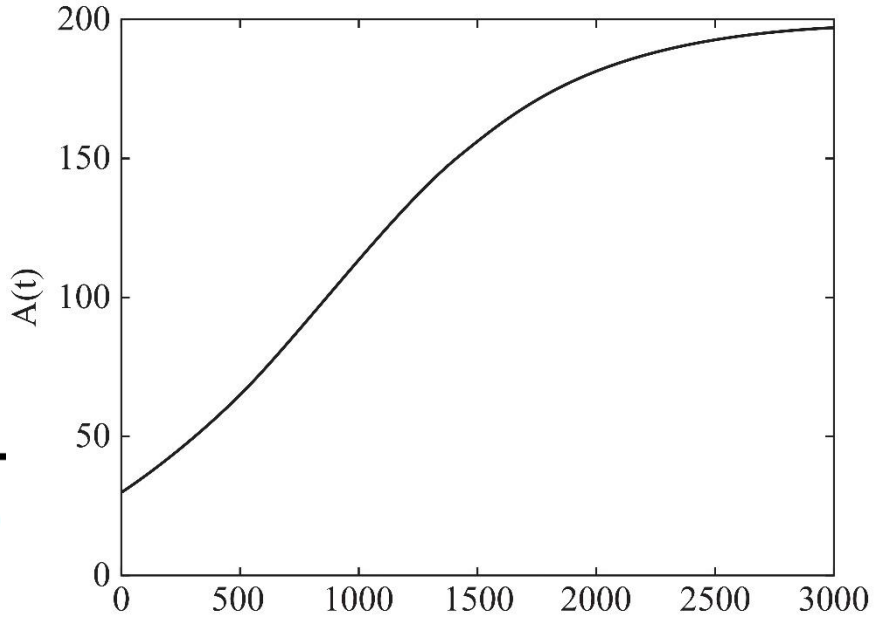
innovate achieve lead

The adoption rate is a function that depends only on the number of already activated individuals

$$i(t) = \beta A(t)$$

$$\frac{dA(t)}{dt} = \beta A(t)[P - A(t)]$$

$$A(t) = \frac{P}{1 + \frac{P - A_0}{A_0} e^{-\beta P (t - t_0)}}$$



Simulation for $A_0 = 30$, $P = 200$ and $\beta = 0.00001$

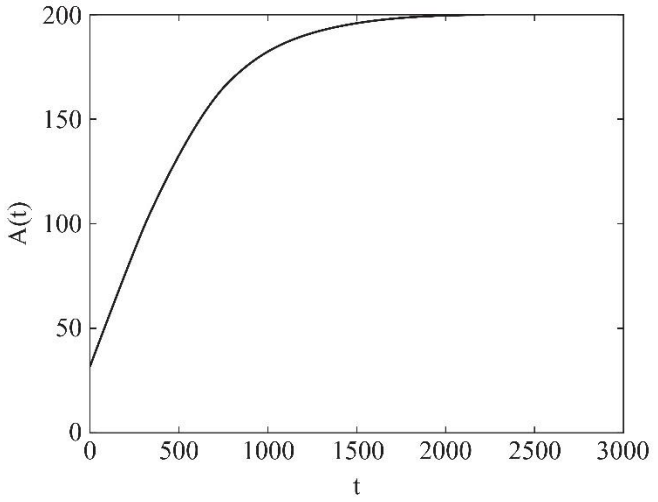
3. Mixed-Influence Model

The adoption rate is a function that depends on both the number of already activated individuals and external forces,

$$i(t) = \alpha + \beta A(t)$$

$$\frac{dA(t)}{dt} = \alpha + \beta A(t)A(t)[P - A(t)]$$

$$A(t) = \frac{P - \frac{\alpha(P-A_0)}{\alpha+\beta A_0} e^{-(\alpha+\beta P)(t-t_0)}}{1 + \frac{\beta(P-A_0)}{\alpha+\beta A_0} e^{-(\alpha+\beta P)(t-t_0)}}$$



Simulation for
$P = 200$
$A_0 = 30$,
$\beta = 0.00001$ **and** $\alpha = 0.001$

Diffusion of Innovation: Intervention

1. **Limiting the distribution of the product or the audience that can adopt the product.**

2. **Reducing interest in the product being sold.**

   A company can inform adopters of the faulty status of the product.

3. **Reducing interactions within the population.**

   Reduced interactions result in less imitations on product adoptions and a general decrease in the trend of adoptions.

Homophily & Influence

Why are connected people similar?

innovate achieve lead

Influence

- The process by which a user (i.e., influential) affects another user
- The influenced user becomes more similar to the influential figure.
  Example: If most of our friends/family members switch to a cellphone company, we might switch [i.e., become influenced] too.

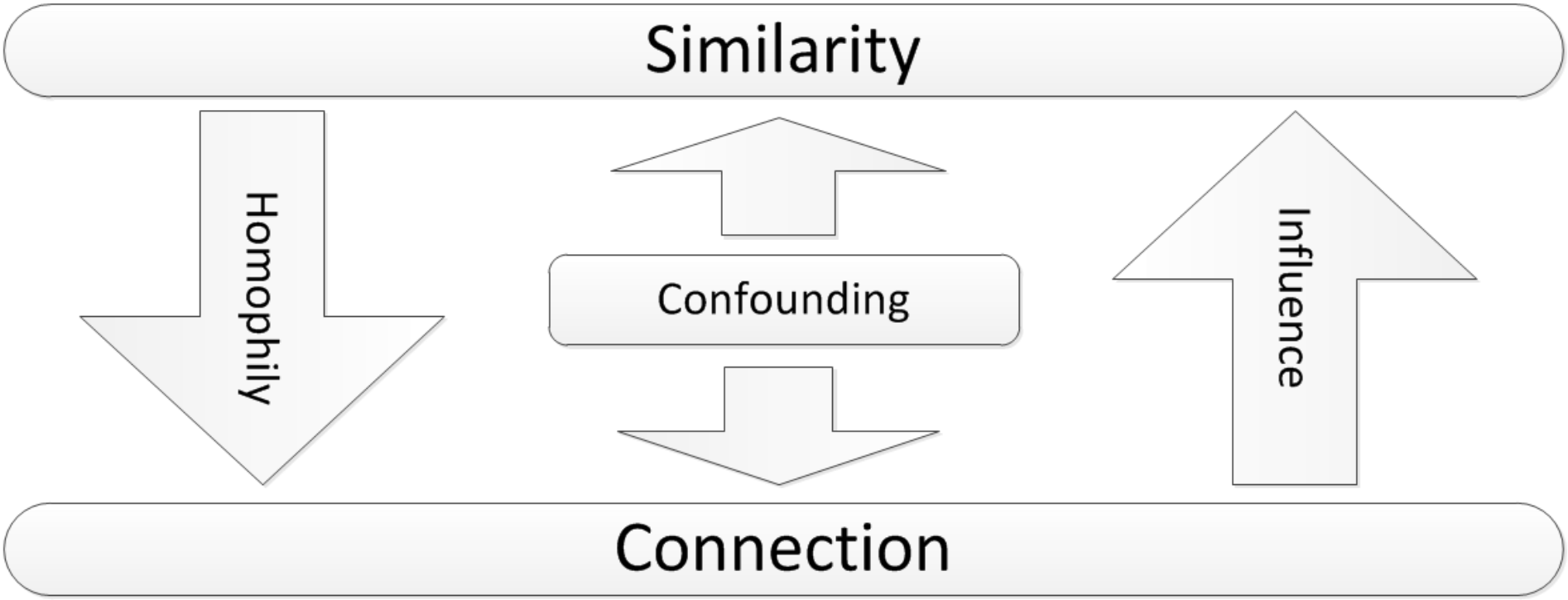
Homophily

- Similar individuals becoming friends due to their high similarity
  - **Example:** Two musicians are more likely to become friends.

Confounding

- The environment's effect on making individuals similar
  - **Example:** Two individuals living in the same city are more likely to become friends than two random individuals

Influence, Homophily, and Confounding



Assortativity: An Example

The friendship network in a US high school in 1994

Colors represent races,

- White: whites
- **Grey**: blacks
- **Light Grey**: hispanics
- **Black**: others

High assortativity between individuals of the same race

Assortativity Significance

innovate achieve lead

Assortativity significance

The difference between **measured assortativity** and **expected assortativity**

The higher this difference, the more significant the assortativity observed

Example

In a school, 50% of the population is **white** and the other 50% is **hispanic**.

We expect 50% of the connections to be between members of different races.

If all connections are between members of different races, then we have a significant finding

Influence

- Measuring Influence
- Modeling Influence

Prediction-based Measurement

innovate achieve lead

We assume that

- an individual's attribute, or
- the way the user is situated in the network

predicts how influential the user **will** be

Example 1:

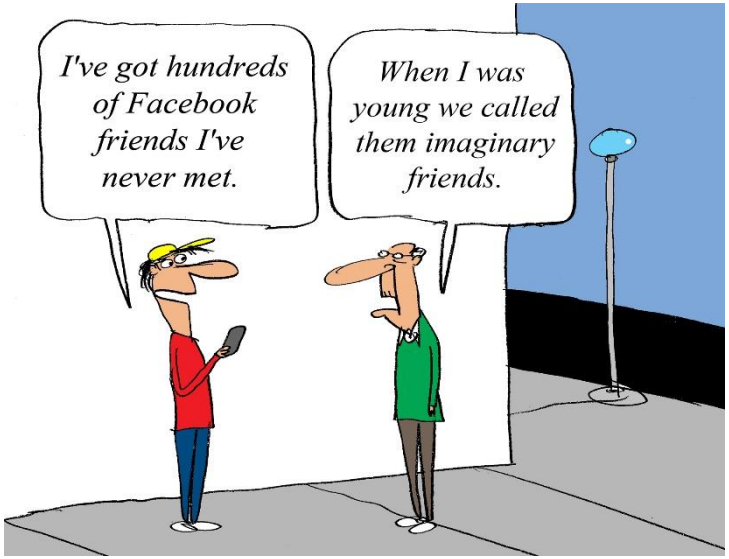
We can assume that the number of friends of an individual is correlated with how influential she will be

It is natural to use any of the centrality measures discussed (Chapter 3) for prediction-based influence measurements

How strong are these friendships?

Example 2:

On Twitter, in-degree (number of followers) is a benchmark for measuring influence commonly used

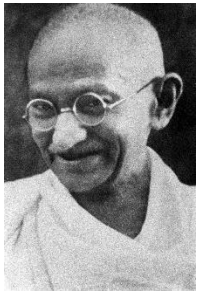


TWEETS 42.7K FOLLOWING 117K FOLLOWERS 214K

Observation-based Measurement

innovate achieve lead

We quantify influence of an individual by measuring the amount of influence attributed to the individual



I. When an individual is the role model

Influence measure: size of the audience that has been influenced

II. When an individual spreads information

Influence measure: the size of the cascade, the population affected, the rate at which the population gets influenced

III. When an individual increases values

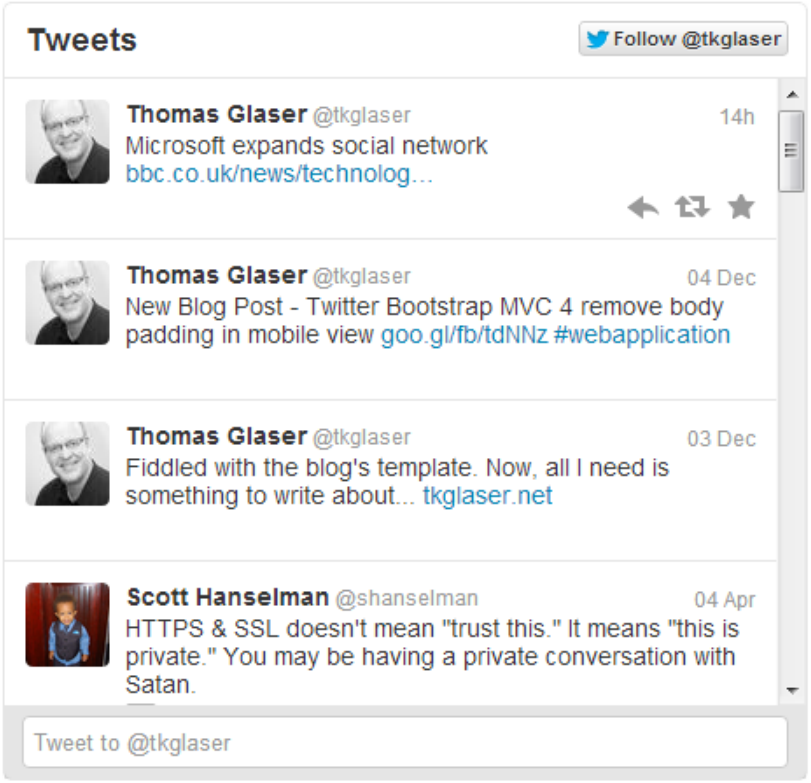
Influence measure: the increase (or rate of increase) in the value of an item or action

The second person who bought the fax machine increased its value dramatically

Measuring Social Influence on Twitter

In **Twitter**, users have an option of following individuals, which allows users to receive tweets from the person being followed

Intuitively, one can think of the number of followers as a measure of influence (in-degree centrality)



Measuring Social Influence on Twitter: Measures

innovate achieve lead

In-degree

The number of users following a person on **Twitter**
Indegree denotes the "audience size" of an individual.

Number of Mentions

The number of times an individual is mentioned in a tweet, by including @username in a tweet.
The number of mentions suggests the "ability in engaging others in conversation"

Number of Retweets

Twitter users have the opportunity to forward tweets to a broader audience via the retweet capability.
The number of retweets indicates individual's ability in generating content that is worth being passed on.

Measuring Social Influence on Twitter: Measures

Each one of these measures by itself can be used to identify influential users in Twitter.

- We utilize the measure for each individual and then rank users based on their measured influence value.

Observation: contrary to public belief, number of followers is considered an inaccurate measure compared to the other two.

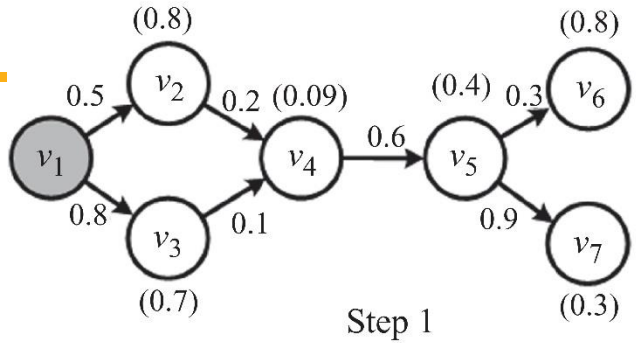
- (a) Followers can be bought (b) Numbers do not equal sales (c) Engagement is more important (d) Consistent growth (e) Focusing on the right numbers

We can rank individuals on twitter independently based on these three measures.

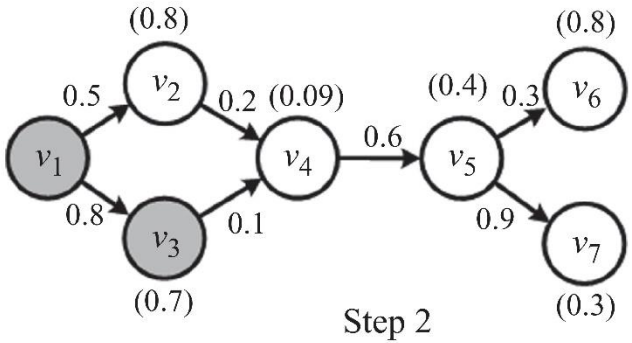
To see if they are correlated or redundant, we can compare ranks of an individuals across three measures using **rank correlation** measures.

Influence Modeling

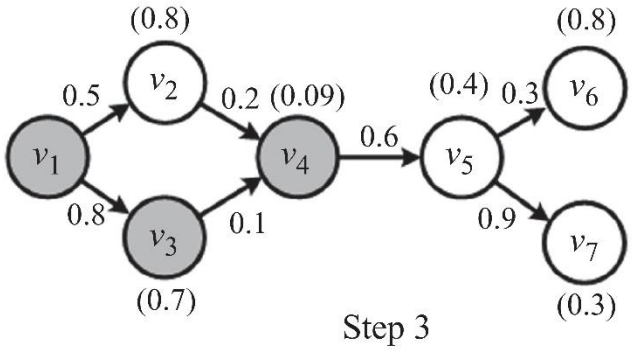
Linear Threshold Model (LTM) - An Example



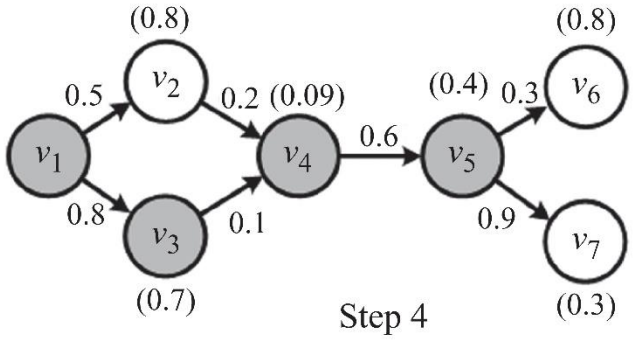
Step 1



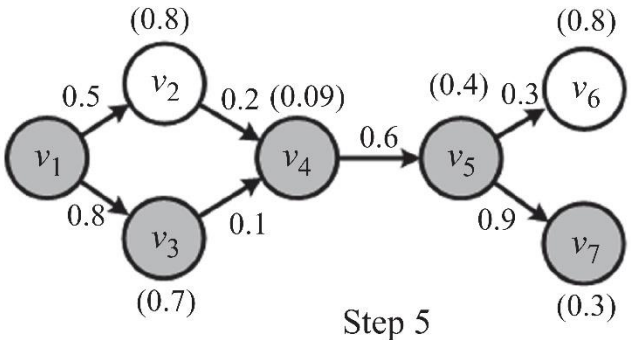
Step 2



Step 3



Step 4



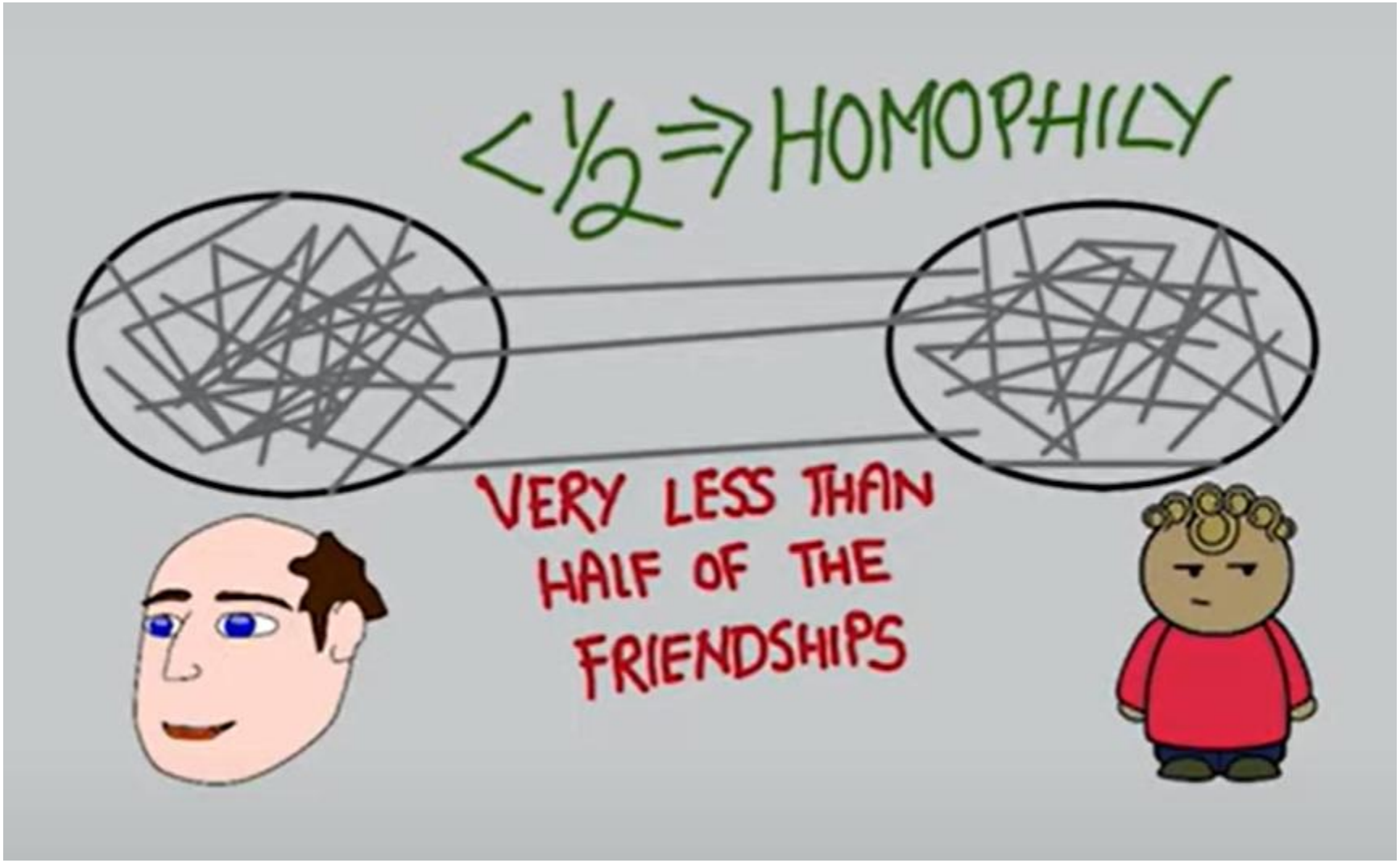
Step 5

Thresholds are on top of nodes

Homophily

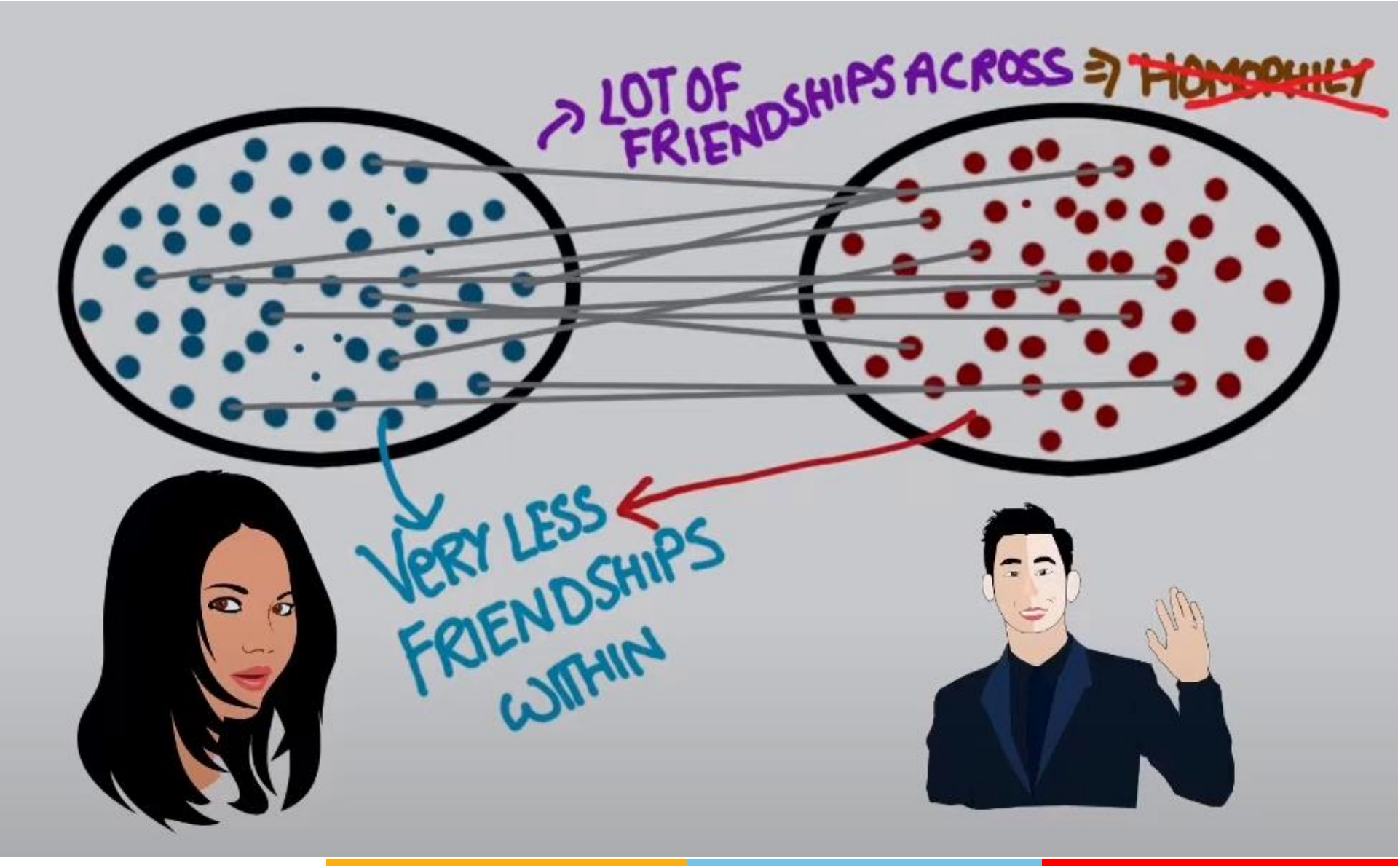
"Birds of a feather flock together"

Homophily Example



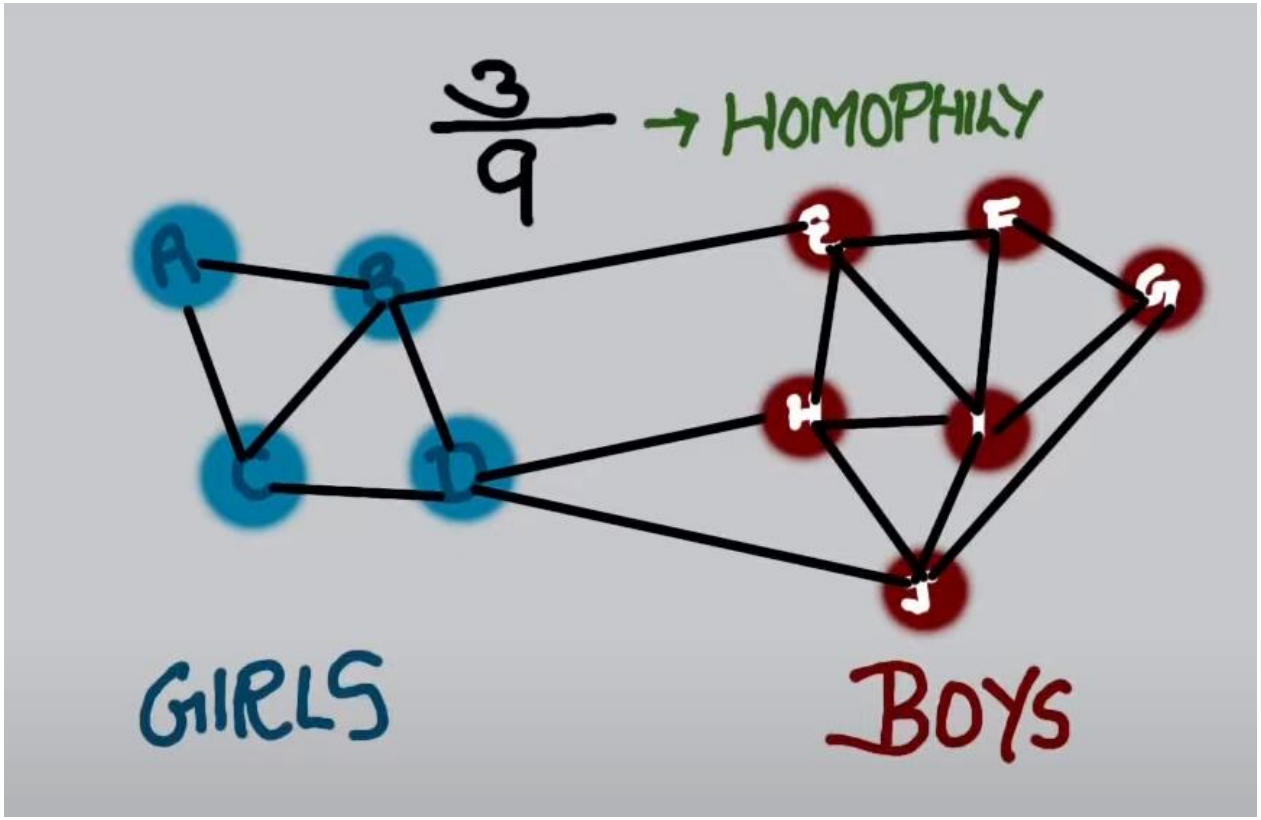
Source: Prof. S.R.S. Iyengar

Example of No Homophily



Source: Prof. S.R.S. Iyengar

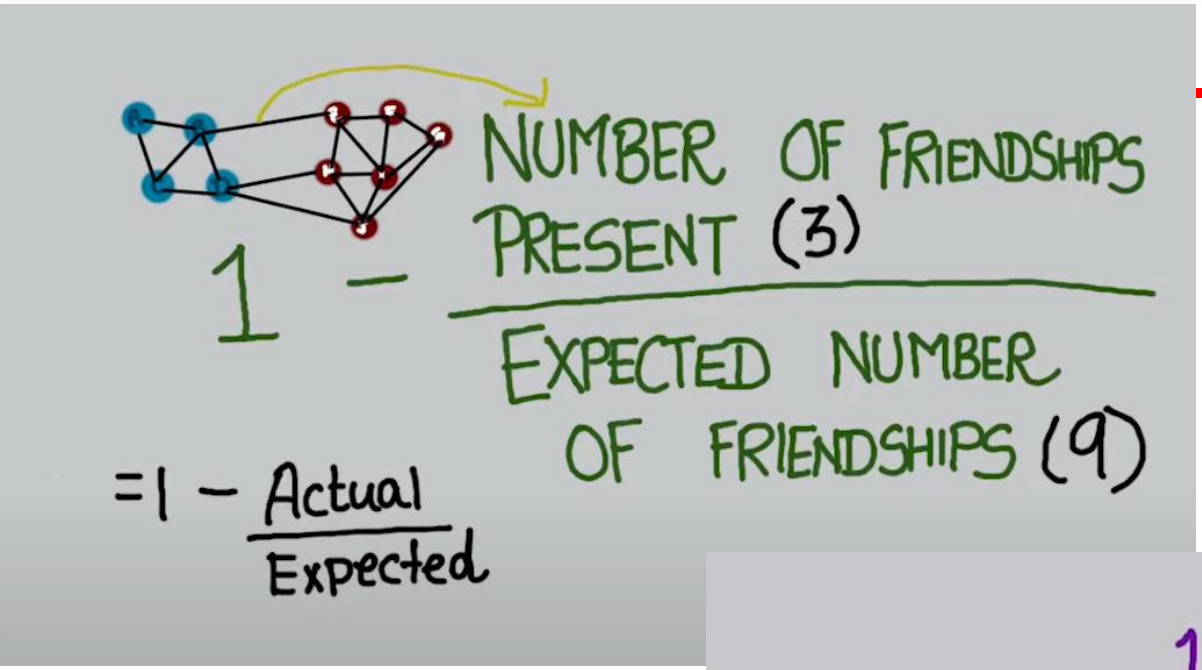
Homophily Measurement: Simple Example



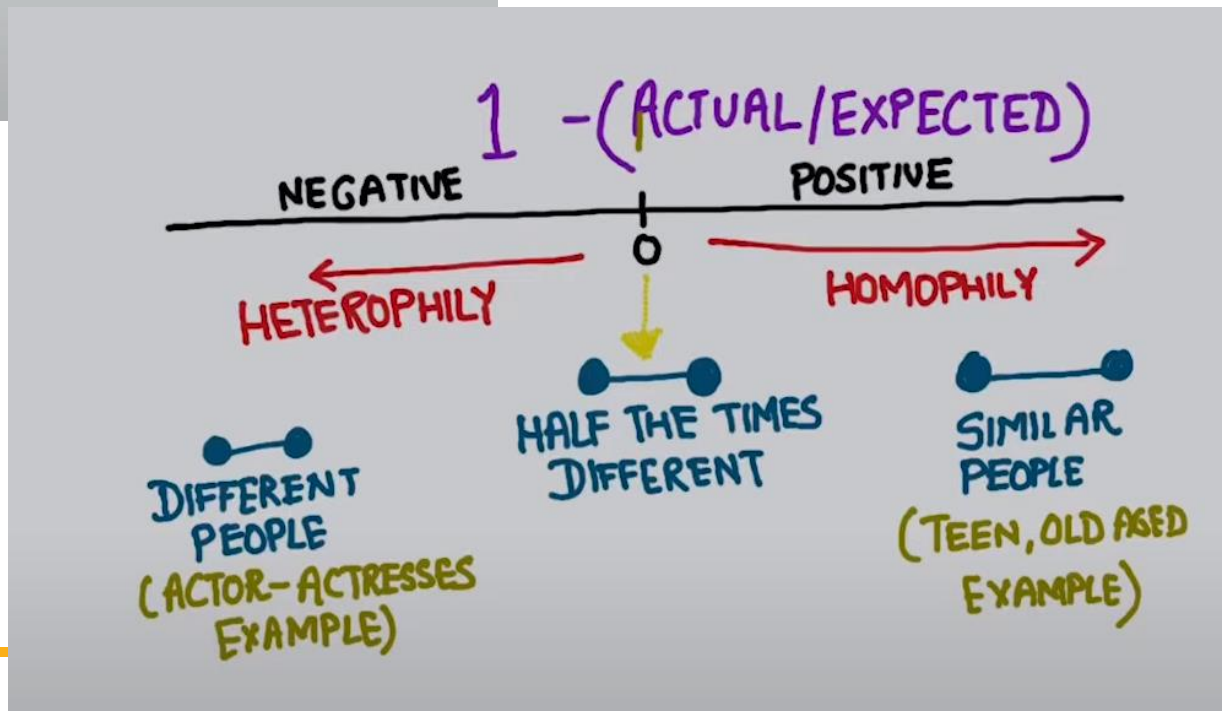
- 4 Girls
- 5 Boys
- 18 friendships
- If no homophily, then half the friendships will be between boys and girls
- Only 3 friendships between boys and girls
- 3/9 < 0.5 → Homophily

Source: Prof. S.R.S. Iyengar

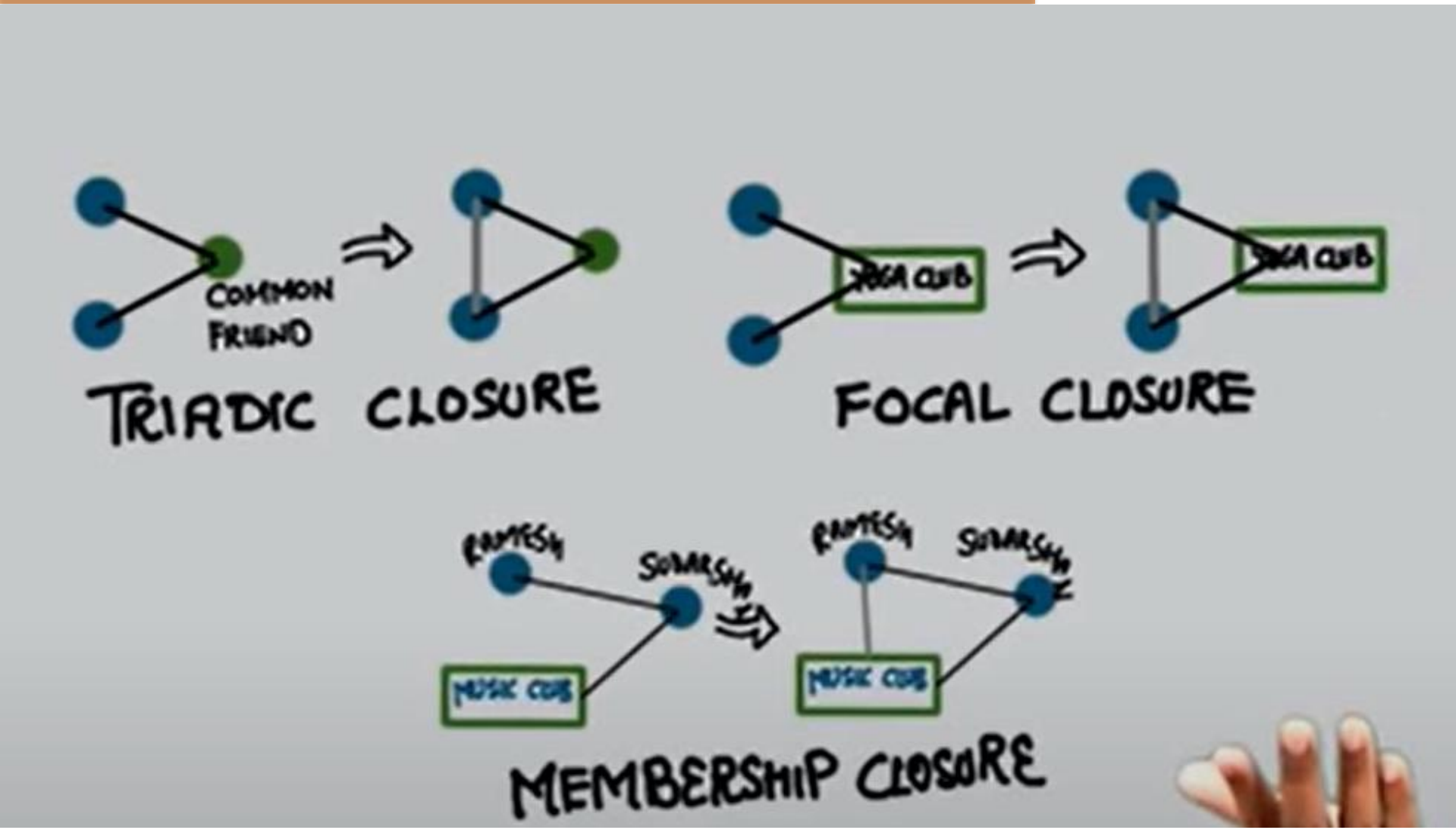
Homophily Measure



- If this value equals Zero, then Actual = Expected → No homophily
- If this value is Negative, then Actual > Expected → Heterogeneous connections → No homophily
- If this value is Positive, then Actual < Expected → Homophily



Source: Prof. S.R.S. Iyengar



Source: [Prof. S.R.S. Iyengar]

Recommendation

When Does This Problem Occur?

There are many choices
There are no obvious advantages among them
We do not have enough resources to check all options (**information overload**)
We do not have enough knowledge and experience to choose, or
- I'm lazy, but don't want to miss out on good stuff
- Defensive decision making

Goal of Recommendation:
To come up with a short list of items that fits user's interests

Classical Recommendation Algorithms

- Content-based algorithms
- Collaborative filtering

Content-Based Methods

Assumption: a user's interest should match the description of the items that the user should be recommended by the system.

The more similar the item's description to that of the user's interest, the more likely that the user finds the item's recommendation interesting.

Goal: find the similarity between the user and all of the existing items is the core of this type of recommender systems

Content-based Recommendation: An Example

Book Database

Title	Genre	Author	Type	Price	Keywords
The Night of the Gun	Memoir	David Carr	Paperback	29.90	press and journalism, drug addiction, personal memoirs, New York
The Lace Reader	Fiction, Mystery	Brunonia Barry	Hardcover	49.90	American contemporary fiction, detective, historical
Into the Fire	Romance, Suspense	Suzanne Brockmann	Hardcover	45.90	American fiction, murder, neo-Nazism
...					

User Profile

Title	Genre	Author	Type	Price	Keywords
...	Fiction, Suspense	Brunonia Barry, Ken Follett	Paperback	25.65	detective, murder, New York

Content-based Recommendation Algorithm

1. Describe the items to be recommended
2. Create a profile of the user that describes the types of items the user likes
3. Compare items with the user profile to determine what to recommend

The profile is often created, and updated automatically in response to feedback on the desirability of items that are presented to the user

More formally

We represent user profiles and item descriptions by vectorizing them using a set of k keywords

We can vectorize (e.g., using **TF-IDF**) both users and items and compute their similarity

$$I_j = (i_{j,1}, i_{j,2}, \ldots, i_{j,k}) \quad U_i = (u_{i,1}, u_{i,2}, \ldots, u_{i,k})$$

$$sim(U_i, I_j) = cos(U_i, I_j) = \frac{\sum_{l=1}^{k} u_{i,l} i_{j,l}}{\sqrt{\sum_{l=1}^{k} {u_{i,l}}^2} \sqrt{\sum_{l=1}^{k} {i_{j,l}}^2}}$$

We can recommend the top most similar items to the user

Collaborative Filtering

Collaborative filtering: the process of selecting information or patterns using techniques involving collaboration among multiple agents, viewpoints, data sources, etc.

Advantage: we don't need to have additional information about the users or content of the items

- Users' rating or purchase history is the only information that is needed to work

Collaborative Filtering

Types of Collaborative Filtering Algorithms:

Memory-based: Recommendation is directly based on previous ratings in the stored matrix that describes user-item relations

Model-based: Assumes that an underlying model (hypothesis) governs how users rate items.

- This model can be approximated and learned.
- The model is then used to recommend ratings.
- **Example**: users rate low budget movies poorly

Memory-Based Collaborative Filtering

Two memory-based methods:

User-based CF

Users with similar **previous** ratings for items are likely to rate future items similarly

	I1	I2	I3	I4
U1	1	2	4	4
U2	1	2	4	?
U3	2	5	2	2
U4	5	2	3	3

Item-based CF

Items that have received similar ratings **previously** from users are likely to receive similar ratings from future users

	I1	I2	I3	I4
U1	1	2	4	4
U2	1	2	4	?
U3	2	5	2	2
U4	5	2	3	3

Measuring Similarity between Users (or Items)

Cosine Similarity

$$sim(U_u, U_v) = cos(U_u, U_v) = \frac{U_u \cdot U_v}{||U_u|| \ ||U_v||} = \frac{\sum_i r_{u,i} r_{v,i}}{\sqrt{\sum_i {r_{u,i}}^2} \sqrt{\sum_i {r_{v,i}}^2}}.$$

Pearson Correlation Coefficient

$$sim(U_u, U_v) = \frac{\sum_i (r_{u,i} - \bar{r}_u)(r_{v,i} - \bar{r}_v)}{\sqrt{\sum_i (r_{u,i} - \bar{r}_u)^2} \sqrt{\sum_i (r_{v,i} - \bar{r}_v)^2}}$$

User-based CF

Updating the ratings:

User u's mean rating

User v's mean rating

$$r_{u,i} = \bar{r}_u + \frac{\sum_{v \in N(u)} sim(u,v)(r_{v,i} - \bar{r}_v)}{\sum_{v \in N(u)} sim(u,v)}$$

Predicted rating of user u for item i

Observed rating of user v for item i

User-based CF, Example

	Lion King	Aladdin	Mulan	Anastasia
John	3	0	3	3
Joe	5	4	0	2
Jill	1	2	4	2
Jane	3	?	1	0
Jorge	2	2	0	1

Predict Jane's rating for Aladdin

1- Calculate average ratings

$$\bar{r}_{John} = \frac{3+3+0+3}{4} = 2.25$$

$$\bar{r}_{Joe} = \frac{5+4+0+2}{4} = 2.75$$

$$\bar{r}_{Jill} = \frac{1+2+4+2}{4} = 2.25$$

$$\bar{r}_{Jane} = \frac{3+1+0}{3} = 1.33$$

$$\bar{r}_{Jorge} = \frac{2+2+0+1}{4} = 1.25$$

2- Calculate user-user similarity

$$sim(Jane, John) = \frac{3\times3+1\times3+0\times3}{\sqrt{10}\sqrt{27}} = 0.73$$

$$sim(Jane, Joe) = \frac{3\times5+1\times0+0\times2}{\sqrt{10}\sqrt{29}} = 0.88$$

$$sim(Jane, Jill) = \frac{3\times1+1\times4+0\times2}{\sqrt{10}\sqrt{21}} = 0.48$$

$$sim(Jane, Jorge) = \frac{3\times2+1\times0+0\times1}{\sqrt{10}\sqrt{5}} = 0.84$$

User-based CF, Example- continued

3- Calculate Jane's rating for Aladdin, Assume that neighborhood size = 2

$$r_{Jane,Aladdin} = \bar{r}_{Jane} + \frac{sim(Jane,Joe)(r_{Joe,Aladdin} - \bar{r}_{Joe})}{sim(Jane,Joe) + sim(Jane,Jorge)} + \frac{sim(Jane,Jorge)(r_{Jorge,Aladdin} - \bar{r}_{Jorge})}{sim(Jane,Joe) + sim(Jane,Jorge)}$$

$$= 1.33 + \frac{0.88(4 - 2.75) + 0.84(2 - 1.25)}{0.88 + 0.84} = 2.33$$

Item-based CF

Calculate the similarity between items and then predict new items based on the past ratings for similar items

$$r_{u,i} = \bar{r}_i + \frac{\sum_{j \in N(i)} sim(i,j)(r_{u,j} - \bar{r}_j)}{\sum_{j \in N(i)} sim(i,j)}$$

Item *i*'s mean rating

i and *j* are two items

Item-based CF, Example

innovate achieve lead

1- Calculate average ratings

$$\bar{r}_{Lion\ King} = \frac{3+5+1+3+2}{5} = 2.8$$

$$\bar{r}_{Aladdin} = \frac{0+4+2+2}{4} = 2.$$

$$\bar{r}_{Mulan} = \frac{3+0+4+1+0}{5} = 1.6$$

$$\bar{r}_{Anastasia} = \frac{3+2+2+0+1}{5} = 1.6$$

2- Calculate item-item similarity

$$sim(Aladdin, Lion\ King) = \frac{0\times3+4\times5+2\times1+2\times2}{\sqrt{24}\sqrt{39}} = 0.84$$

$$sim(Aladdin, Mulan) = \frac{0\times3+4\times0+2\times4+2\times0}{\sqrt{24}\sqrt{25}} = 0.32$$

$$sim(Aladdin, Anastasia) = \frac{0\times3+4\times2+2\times2+2\times1}{\sqrt{24}\sqrt{18}} = 0.67$$

3- Calculate Jane's rating for Aladdin, Assume that neighborhood size = 2

$$r_{Jane,Aladdin} = \bar{r}_{Aladdin} + \frac{sim(Aladdin, Lion\ King)(r_{Jane,Lion\ King} - \bar{r}_{Lion\ King})}{sim(Aladdin, Lion\ King) + sim(Aladdin, Anastasia)} + \frac{sim(Aladdin, Anastasia)(r_{Jane,Anastasia} - \bar{r}_{Anastasia})}{sim(Aladdin, Lion\ King) + sim(Aladdin, Anastasia)}$$

$$= 2 + \frac{0.84(3-2.8) + 0.67(0-1.6)}{0.84+0.67} = 1.40$$

Model-Based Collaborative Filtering

In memory-based methods

We predict the missing ratings based on similarities between users or items.

In model-based collaborative filtering

We assume that an underlying model governs how users rate.

We learn that model and use it to predict the missing ratings.

Among a variety of model-based techniques, we focus on a well-established model-based technique that is based on singular value decomposition (SVD).

Singular Value Decomposition (SVD)

SVD is a linear algebra technique that, given a real matrix $X \in \mathbb{R}^{m \times n}, m \geq n$, and factorizes it into three matrices

$$X = U \Sigma V^T$$

Matrices $U \in \mathbb{R}^{m \times m}$ and $V \in \mathbb{R}^{n \times n}$ are orthogonal and matrix $\Sigma \in \mathbb{R}^{m \times n}$ is diagonal

The product of these matrices is equivalent to the original matrix

No information is lost!

Low-rank Matrix Approximation

A Low-rank matrix approximation of matrix $X \in \mathbb{R}^{m \times n}$ is another matrix $C \in \mathbb{R}^{m \times n}$

Matrix C approximates X, and C's rank (the maximum number of linearly independent columns) is a fixed number $k \ll \min(m, n)$

$$Rank(C) = k$$

The best low-rank matrix approximation is a matrix C that minimizes $\|X\|_F = \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} X_{ij}^2}$

Low-rank approximation can remove noise by assuming that the matrix is not random and has an underlying structure.

- SVD can compute a low-rank approximation of a matrix.

Low-Rank Matrix Approximation with SVD

1. Create Σ_k from Σ by keeping only the first k elements on the diagonal. This way, $\Sigma_k \in \mathbb{R}^{k \times k}$.

2. Keep only the first k columns of U and denote it as $U_k \in \mathbb{R}^{m \times k}$, and keep only the first k rows of V^T and denote it as ${V_k}^T \in \mathbb{R}^{k \times n}$.

3. Let $X_k = U_k \Sigma_k {V_k}^T$, $X_k \in \mathbb{R}^{m \times n}$.

Theorem 9.1 (Eckart-Young-Mirsky Low-Rank Matrix Approximation). *Let X be a matrix and C be the best low-rank approximation of X; if $\|X - C\|_F$ is minimized, and rank$(C) = k$, then $C = X_k$.*

X_k is the best low-rank approximation of a matrix X

Model-based CF, Example

Table 9.2: An User-Item Matrix

	Lion King	Aladdin	Mulan
John	3	0	3
Joe	5	4	0
Jill	1	2	4
Jorge	2	2	0

$$U = \begin{bmatrix} -0.4151 & -0.4754 & -0.7679 & 0.1093 \\ -0.7437 & 0.5278 & 0.0169 & -0.4099 \\ -0.4110 & -0.6626 & 0.6207 & -0.0820 \\ -0.3251 & 0.2373 & 0.1572 & 0.9018 \end{bmatrix}$$

$$\Sigma = \begin{bmatrix} 8.0265 & 0 & 0 \\ 0 & 4.3886 & 0 \\ 0 & 0 & 2.0777 \\ 0 & 0 & 0 \end{bmatrix}$$

$$V^T = \begin{bmatrix} -0.7506 & -0.5540 & -0.3600 \\ 0.2335 & 0.2872 & -0.9290 \\ -0.6181 & 0.7814 & 0.0863 \end{bmatrix}$$

Considering a rank 2 approximation (i.e., k = 2), we truncate all three matrices:

$$U_k = \begin{bmatrix} -0.4151 & -0.4754 \\ -0.7437 & 0.5278 \\ -0.4110 & -0.6626 \\ -0.3251 & 0.2373 \end{bmatrix}$$

$$\Sigma_k = \begin{bmatrix} 8.0265 & 0 \\ 0 & 4.3886 \end{bmatrix}$$

$$V_k^T = \begin{bmatrix} -0.7506 & -0.5540 & -0.3600 \\ 0.2335 & 0.2872 & -0.9290 \end{bmatrix}$$

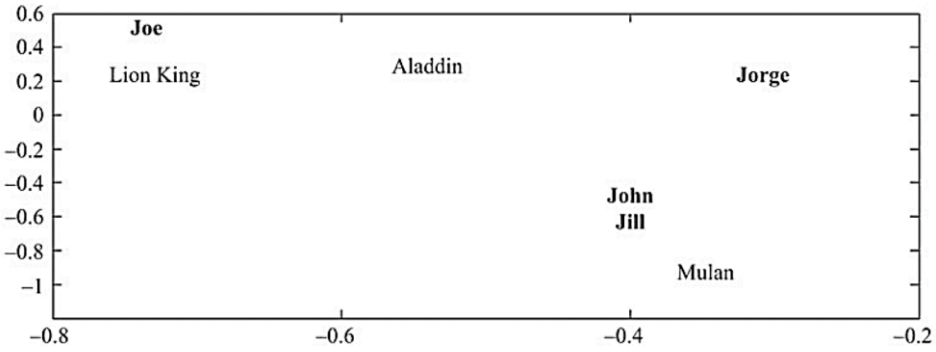


Figure 9.1: Users and Items in the 2-D Space.

Recommendation to a Group

Aggregation Strategies

innovate achieve lead

Maximizing Average Satisfaction

Average everyone's ratings and choose the max

$$R_i = \frac{1}{n} \sum_{u \in G} r_{u,i}$$

Least Misery

This approach tries to minimize the dissatisfaction among group's members (max of all mins)

$$R_i = \min_{u \in G} r_{u,i}$$

Most Pleasure

The maximum of individuals' maximum ratings is taken as group's rating

$$R_i = \max_{u \in G} r_{u,i}$$

Recommendation to Group, an Example

Table 9.3: User-Item Matrix

	Soda	Water	Tea	Coffee
John	1	3	1	1
Joe	4	3	1	2
Jill	2	2	4	2
Jorge	1	1	3	5
Juan	3	3	4	5

Consider group G ={John; Jill; Juan}

Average Satisfaction

$$R_{Soda} = \frac{1+2+3}{3} = 2.$$

$$R_{Water} = \frac{3+2+3}{3} = 2.66$$

$$R_{Tea} = \frac{1+4+4}{3} = 3.$$

$$R_{Coffee} = \frac{1+2+5}{3} = 2.66$$

Least Misery

$$R_{Soda} = \min\{1,2,3\} = 1$$

$$R_{Water} = \min\{3,2,3\} = 2$$

$$R_{Tea} = \min\{1,4,4\} = 1$$

$$R_{Coffee} = \min\{1,2,5\} = 1$$

Most Pleasure

$$R_{Soda} = \max\{1,2,3\} = 3$$

$$R_{Water} = \max\{3,2,3\} = 3$$

$$R_{Tea} = \max\{1,4,4\} = 4$$

$$R_{Coffee} = \max\{1,2,5\} = 5$$

Recommendation Using Social Context

- Recommendation using social context alone
- Extending classical methods with social context
- Recommendation constrained by social context

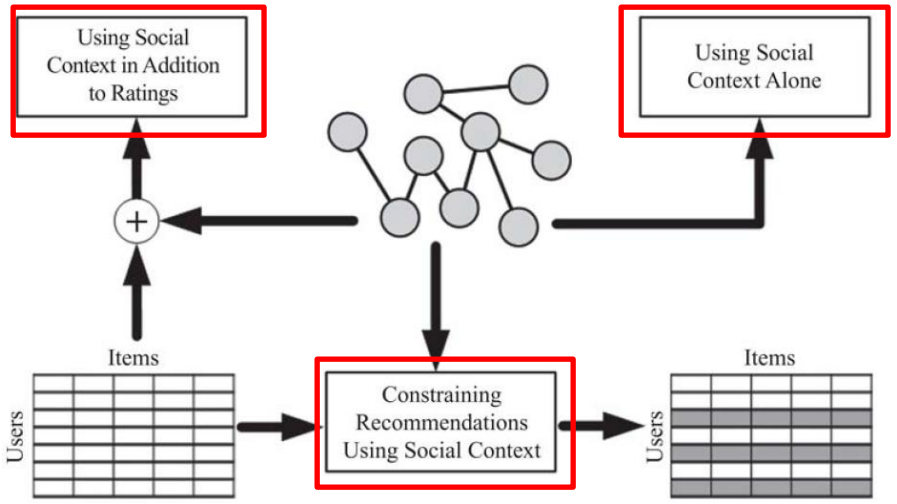
Information Available in Social Context

In social media, in addition to ratings of products, there is additional information

E.g., the friendship network

This information can be used to improve recommendations

- Assuming that friends have an impact on the ratings ascribed by the individual.
- This impact can be due to homophily, influence, or confounding



I. Recommendation Using Social Context Alone

Consider a network of friendships for which no user-item rating matrix is provided.

In this network, we can still recommend users from the network to other users for friendship.

This is an example of friend recommendation in social networks

II. Extending Classical Methods

Using Social information in addition to a user-item rating matrix to improve recommendation.

Addition of social information:

We assume that friends rate similar items similarly.

$$R = U^T V$$

$$R \in \mathbb{R}^{n \times m},\ U \in \mathbb{R}^{k \times n},\ V \in \mathbb{R}^{k \times m}$$

$$\min_{U,V} \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{m} I_{ij} (R_{ij} - U_i^T V_j)^2$$

Optimize only for non-zero elements

Modeling Social Information in Recommendation

innovate achieve lead

Incorporating similarity: The taste for user i is close to that of all his friends $j \in F(i)$

$$\sum_{i=1}^{n} \sum_{j \in F(i)} sim(i,j) \|U_i - U_j\|_F^2$$

$sim(i,j)$ denotes the similarity between user i and j (e.g., cosine between their ratings)

$F(i)$ denotes the friends of i

Final Formulation:

$$\min_{U,V} \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{m} I_{ij}(R_{ij} - U_i^T V_j)^2 + \beta \sum_{i=1}^{n} \sum_{j \in F(i)} sim(i,j) \|U_i - U_j\|_F^2$$

$$+ \frac{\lambda_1}{2} \|U\|_F^2 + \frac{\lambda_2}{2} \|V\|_F^2$$

Controlling Sparsity

3. Recommendation Constrained by Social Context

In classical recommendation,

To estimate ratings, we determine similar users or items.

Any user similar to the individual can contribute to the predicted ratings for the individual.

We can limit the set of individuals that can contribute to the ratings of a user to the set of friends of the user.

$S(i)$ is the set of k most similar **friends** of an individual

$$r_{u,i} = \bar{r}_u + \frac{\sum_{v \in S(u)} sim(u,v)(r_{v,i} - \bar{r}_v)}{\sum_{v \in S(u)} sim(u,v)}$$

Example

$$A = \begin{bmatrix} & John & Joe & Jill & Jane & Jorge \\ John & 0 & 1 & 0 & 0 & 1 \\ Joe & 1 & 0 & 1 & 0 & 0 \\ Jill & 0 & 1 & 0 & 1 & 1 \\ Jane & 0 & 0 & 1 & 0 & 0 \\ Jorge & 1 & 0 & 1 & 0 & 0 \end{bmatrix}$$

	Lion King	Aladdin	Mulan	Anastasia
John	4	3	2	2
Joe	5	2	1	5
Jill	2	5	?	0
Jane	1	3	4	3
Jorge	3	1	1	2

Considering a neighborhood of size 2, the most similar users to Jill are John and Jane:
N(Jill) = {John; Jane}
We also know that friends of Jill are
F(Jill) = {Joe; Jane; Jorge}

Predict the missing rating by taking the intersection of friends and neighbors

$$r_{Jill,Mulan} = \bar{r}_{Jill} + \frac{sim(Jill, Jane)(r_{Jane,Mulan} - \bar{r}_{Jane})}{sim(Jill, Jane)}$$
$$= 2.33 + (4 - 2.75) = 3.58.$$

Predict the missing rating by taking the two most similar friends:

$$r_{Jill,Mulan} = \bar{r}_{Jill} + \frac{sim(Jill, Jane)(r_{Jane,Mulan} - \bar{r}_{Jane})}{sim(Jill, Jane) + sim(Jill, Jorge)}$$
$$+ \frac{sim(Jill, Jorge)(r_{Jorge,Mulan} - \bar{r}_{Jorge})}{sim(Jill, Jane) + sim(Jill, Jorge)}$$
$$= 2.33 + \frac{0.72(4 - 2.75) + 0.54(1 - 1.75)}{0.72 + 0.54} = 2.72$$

Average Ratings

$$\bar{r}_{John} = \frac{4+3+2+2}{4} = 2.75.$$
$$\bar{r}_{Joe} = \frac{5+2+1+5}{4} = 3.25.$$
$$\bar{r}_{Jill} = \frac{2+5+0}{3} = 2.33.$$
$$\bar{r}_{Jane} = \frac{1+3+4+3}{4} = 2.75.$$
$$\bar{r}_{Jorge} = \frac{3+1+1+2}{4} = 1.75.$$

User Similarity

$$sim(Jill, John) = \frac{2\times4+5\times3+0\times2}{\sqrt{29}\sqrt{29}} = 0.79$$
$$sim(Jill, Joe) = \frac{2\times5+5\times2+0\times5}{\sqrt{29}\sqrt{54}} = 0.50$$
$$sim(Jill, Jane) = \frac{2\times1+5\times3+0\times3}{\sqrt{29}\sqrt{19}} = 0.72$$
$$sim(Jill, Jorge) = \frac{2\times3+5\times1+0\times2}{\sqrt{29}\sqrt{14}} = 0.54$$

Data Privacy & Ethics

Ethical challenges

Facebook
- Privacy of Personal Information
- Freedom of Speech
- Data Leakage
- Identity Theft
- Fake News

Instagram
- Terms of Service & Privacy Issues
- Selling of Private Data
- Rise of Influencer Marketing

Twitter
- Fake Accounts
- Paid Tweets
- Lack of context Tweets
- Ghost Tweets
- Data Selling

Linked In
- Job Board Issues
- Erroneous information
- Lack of legal guidance
- Invasion of Privacy

Individual Level	Organizational Level
Invasion of Privacy Re-identification of Data Profiling and misuse of data Data mining risk Mis-use of free expertise & contests Anonymous Information	Competitive Pressure Poor Quality of Data Data Sharing / Sourcing Decision Making Presentation & Information

Social Media Monitoring – Google Analytics

Google Analytics: Glossary ...1/4

- **Automatically Collected Events** – One of the main differences between Universal Analytics and GA4 is the introduction of automatically collected events. No additional tracking is required for these events, they are automatically sent to GA by the global site tag. A full list of these events can be found here: **GA4 Automatically Collected Events**

- **Connected Site Tags** – There is a feature within GA4 called Connected Site Tags. This feature makes it possible to reuse existing Universal Analytics tagging to create a connected GA4 property. This means that you don't necessarily have to add more code to your website, in order to enable GA4 tracking – you can simply reuse existing tracking tags instead.

- **Custom Dimensions** – This is an area in GA4 that contains Custom Dimensions and Metrics. Here, you can use event parameters to create custom dimensions and metrics to be used within your reports, making labeling and understanding your data much easier.

https://www.udemy.com/course/ultimate-google-analytics-training-for-beginners/?couponCode=IND21PM

Google Analytics: Glossary …2/4

- **Data Stream** – Data Stream refers to the flow of data between your website or application and Google Analytics. Within GA4, there are three different types of data streams you can use to carry website statistics to GA – Web (for websites), Android (for Android applications), and iOS (for iOS apps).

- **Debug View** – In GA4 their Debug View allows you to test conversions and monitor events in real-time, in order to check that your tracking and reporting are working as they should. You can see events and conversions at the moment they are triggered.

- **Engaged Sessions** – This is a term used quite frequently throughout Google Analytics 4. Rather than counting all sessions within a site, GA4 focuses on engaged sessions – those where the session either lasted at least 10 seconds, had one or more conversion events, or had two or more page views. Therefore sessions in GA4 are likely to be lower than those in Universal Analytics, but they are arguably a much more valuable metric to measure.

https://www.udemy.com/course/ultimate-google-analytics-training-for-beginners/?couponCode=IND21PM

Google Analytics: Glossary …3/4

- **Enhanced Measurement** – Enhanced Measurement affects only web data streams in GA4. This feature enables a number of different events, allowing you to measure a larger number of interactions between website visitors and your content. **Click here to take a look at the full list of enhanced measurement options available within GA4.**

- **Explore** – The Explore section of GA4 is an area where users can use tables and graphs to visualize their data using highly customizable and flexible tables and graphs.

- **Life Cycle** – The Life Cycle section of GA4 contains reports that help analyze data by the stage your customers are within the overall purchase journey. Within this section, you will find reports on user acquisition, engagement, monetization, and retention.

https://www.udemy.com/course/ultimate-google-analytics-training-for-beginners/?couponCode=IND21PM

Google Analytics: Glossary …4/4

- **Monetisation** – The monetization reports in GA4 make it easier to analyze purchase activity on your website/app. This is where you'll find the data previously stored in the Conversions > E-commerce area of Universal Analytics, such as e-commerce conversion rates, product promotions, coupon uses and more.

- **Tech** – The updated Tech section of the Google Analytics profile contains data previously found within the Audience report – specifically statistics regarding user platform, operating system, app version and screen resolution. You can easily see reports mapping users by platform, OS, device and more.

- **User Snapshot** – User Snapshot is a feature of GA4, one that allows you to explore individual users and their real-time engagement with your website and/or application. Rather than just grouping all real-time data into one group, you can find out behavior and engagement data associated with individual users who are visiting your site.

https://www.udemy.com/course/ultimate-google-analytics-training-for-beginners/?couponCode=IND21PM

GA4 Sample Metrics…1/4

Acquisition – Acquisition metrics show where your traffic is originating from, be it Google searches, social media links, or other websites.

Average Session Duration – The average visit length of time a user spends on your website at any given time. This is a key metric for measuring the effectiveness and quality of your website.

Average Time on Page – The average time that users spend viewing a page or group of pages.

Bounce Rate – A bounce is a single page website visit, and so your site's bounce rate is the percentage of single page visits that your site has. Generally you want this number to be as low as possible, however sites with standalone pages such as blog articles tend to have lower bounce rates by nature.

Direct Traffic – Visitors that came directly to your site by typing your company website's URL into their browser's address bar or through a saved bookmark. Direct traffic generally indicates how many visitors already know your company and URL.

https://www.udemy.com/course/ultimate-google-analytics-training-for-beginners/?couponCode=IND21PM

GA4 Sample Metrics…2/4

Event – A ‘hit’ that tracks user interactions, such as clicks, downloads, and video plays.

Exit Page – The last page a user visits before leaving your website.

Filter – A tool that allows you to include or exclude specific data in your reports. For example, you can exclude internal company traffic so that your employees are not included in the website metrics. You can also exclude known bots.

Goal Conversion – This is the completion of an activity on your site that is important to the success of your business, such as a completed sign up for your email newsletter. You must set this up first before Google will track a goal conversion.

Landing Page – The first page that someone visits when they come to your site. Often this is the homepage.

Organic Traffic – Users who come to your website from natural (or unpaid) search engine results.

Pages/Session – The average number of pages viewed during one visit.

https://www.udemy.com/course/ultimate-google-analytics-training-for-beginners/?couponCode=IND21PM

GA4 Sample Metrics…3/4

Pageviews – The total number of website pages viewed. For example, if one user visited your homepage and the contact page, then that would count as 2 pageviews.

Referral Traffic – Visitors that landed on your website through a link on another website, such as Facebook or a site that references one of your blog articles.

Returning Visitors – Visitors that have previously visited your website (on the same device).

Search Traffic – Visitors that came to your website through a search engine such as Google or Bing.

Sessions – A session is a single continual active viewing period by a visitor. If a user visits a site several different times in one day, each unique visit counts as a session.

Source/Medium – Grouped together, source is the origin of traffic (such as bing or twitter) and medium is the category of the source (such as organic or social)

https://www.udemy.com/course/ultimate-google-analytics-training-for-beginners/?couponCode=IND21PM

GA4 Sample Metrics…4/4

Unique Visitors – The number of unduplicated visitors to your website (each user only counted once).

Unique Pageviews – Combines the pageviews from the same user in the same session, counted as one unique pageview.

Users – The number of people that have visited your site at least once during a given time period. One user could have multiple sessions, but will still be counted as a single user.

% Exit – The ratio of exits to pageviews. This indicates how often users leave page(s) compared to how many pages they view

https://www.udemy.com/course/ultimate-google-analytics-training-for-beginners/?couponCode=IND21PM

Applications of Social Network Analysis: Marketing & Other Applications

Types of Social Media Analytics

- Performance Analysis
  - Impressions
  - Reach
  - Likes
  - Comments
  - Shares
  - Views
  - Clicks
  - Sales
- Audience Analytics
  - Age
  - Gender
  - Location
  - Device
- Competitor Analysis
  - # of followers
  - Engagement Rate
- Ad Analytics
  - Total number of active ads
  - Clicks
  - Click-through rate
  - Cost-per-click
  - Cost-per-engagement
  - Cost-per-action
  - Conversion rate
  - Total ad spend
- Influencer Analysis
  - Number of posts created per influencer
  - Total number of interactions per post
  - Audience size of each influencer
  - Hashtag usage and engagement
- Sentiment Analysis
  - Track Brand Sentiment
  - Relevant keywords and topics

Importance of Social Network Analysis

- **Strategic Advantage**: By understanding the connections and interactions, organisations can mobilise resources more efficiently, enhance cooperation and knowledge sharing, stimulate innovation, and gain a strategic advantage.
- **Improved Understanding**: It paves the way to understanding patterns and trends, uncovering hidden channels of information flow and decision making within and across organisations.
- **Risk Management**: SNA provides a better understanding of dependencies that could pose risks to the functionality and productivity of the system, thereby enhancing risk management.

The fantastic thing about SNA is that it **reveals the invisible** - the behind-the-scenes information flow, the influencers, gatekeepers, and liaisons. By understanding this, **businesses can enhance their strategies, communications and understand the informal and formal structures within their organisation.**

Reference: Study Smarter

SNA Examples

- Understanding political structures
- Investigating the spread of diseases
- Tracing the flow of information in an organisation
- Transaction web in cryptocurrencies

Within a corporate setting:

- Insights into the informal networks that exist alongside the official organisation chart. For instance, employees often seek guidance not from their official superiors but from experienced colleagues.
- An SNA in this scenario could help to identify these individuals, measure their importance (using measures like degree centrality and betweenness centrality), and assess the impact of their eventual retirement or departure from the company.
- SNA could also showcase structural gaps where communication or collaboration is missing but necessary.

Reference: Study Smarter

SNA in Marketing

In marketing and brand strategy, SNA can help chart the landscape of social influencers. By determining the degree centrality, one can identify individuals who, due to their vast network of connections, can be instrumental in spreading content widely.

Betweenness centrality, on the other hand, can help identify those individuals who serve as critical brokers or bridges between diverse parts of the network. They might not have the highest number of connections, but they hold influence because they link different communities or groups.

A cosmetics company planning to release a new product might use SNA to identify key influencers in the beauty community. By sending products to these individuals and securing their endorsement, the company can ensure that news of the product reaches a wide audience more effectively than through traditional advertising methods.

Reference: Study Smarter

Other SNA Applications

- **Sociology**: Just as the name suggests, SNA was first developed by sociologists to understand social structures. It can unveil the complexities of human interactions, such as analysing online communities, tracking socioeconomic disparity, and studying the diffusion of cultural trends.
- **Computer Science & IT**: SNA has become a vital part of computational data analysis, primarily for the Internet and its structure. It's employed in areas like web graph analysis, cybersecurity for tracing the proliferation of malware and even optimising cloud computing networks.
- **Political Studies**: In political science, SNA is used to study policy networks, political parties, political blogs, or even to understand power structures among nations. It also aids in tracking the diffusion of political ideologies and trends.
- **Business Operations**: SNA is actively utilised to optimise organisational structures, enhance communication networks, and improve marketing strategies

Reference: Study Smarter

SNA in Business Operations

Social Network Analysis emerges as a powerful process enhancement tool. It offers a unique perspective to aid solving many business-related issues, like enhancing team collaborations, improving inter-departmental communication or even understanding customer behaviours.

Employee Interaction and Collaboration

Organisations are essentially a complex web of interactions and relationships. SNA helps to visualise this web, further enabling the organisation to understand the communication flow and thereby, promoting better collaborations. Using measures like degree centrality and betweenness centrality, one can identify key individuals who are acting as information gatekeepers.

Example: Suppose there's an individual who doesn't have an official leadership title, but their departure greatly hampers the workflow. This could possibly be because they hold a pivotal position within the informal network, answering colleagues' queries, mediating discussions, or ensuring coordination. Understanding these informal roles through SNA could significantly enhance workflow management.

Reference: Study Smarter

SNA in Business Operations

Organisational Knowledge Management

Knowledge and information in an organisation do not follow a clear-cut path as depicted by official hierarchies. Instead, it flows across organisational boundaries in rather unexpected ways. SNA allows the identification of such unconventional paths.

Example

‘T-shaped' skills, for instance, where a person has depth of knowledge in one subject (the vertical bar of the T) along with the ability to collaborate across disciplines and apply knowledge in areas of expertise other than their own (the horizontal bar of the T), are essential for innovation. SNA can help identify such individuals with 'T-shaped' skills and foster cross-disciplinary learning.

Reference: Study Smarter

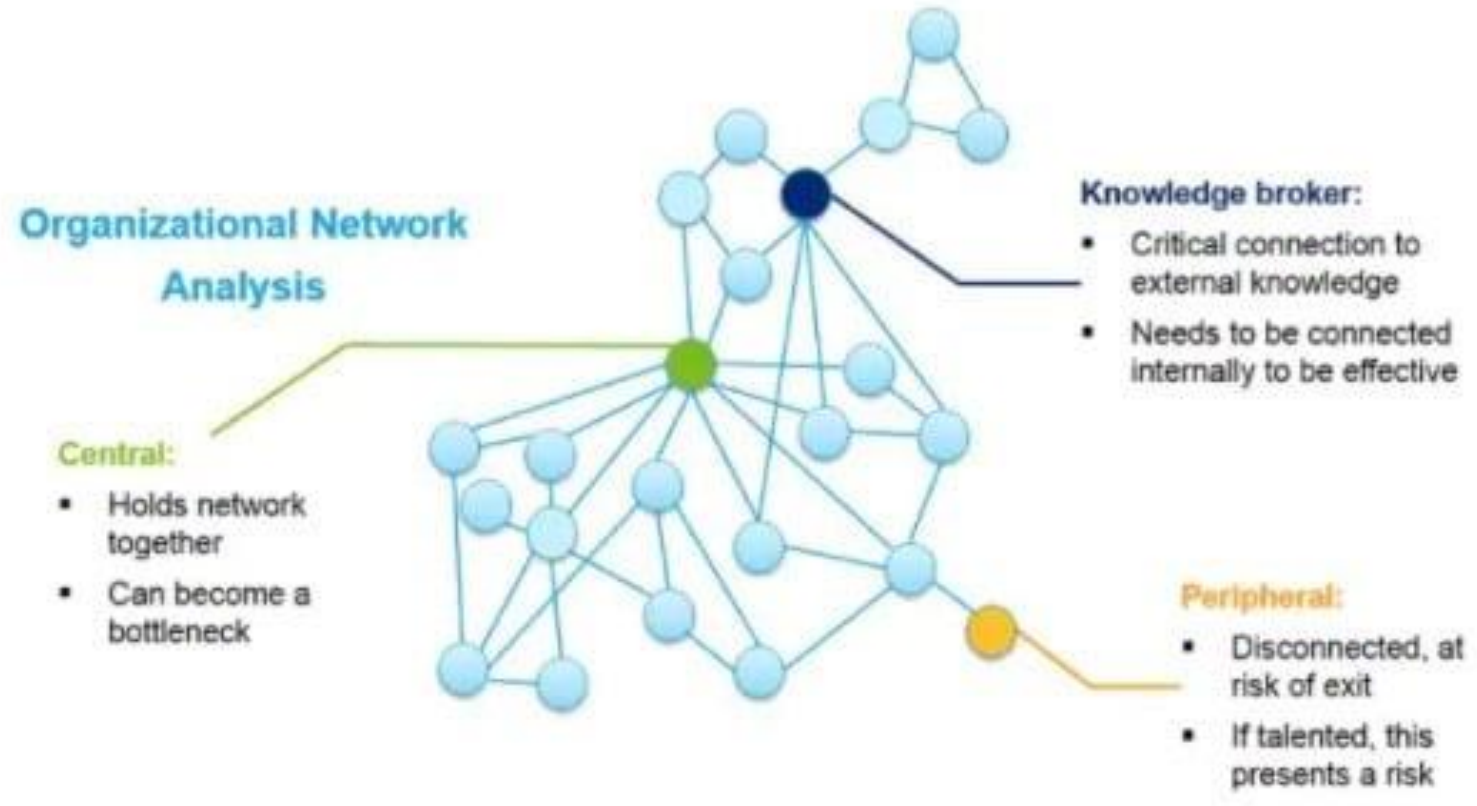
SNA in Business Operations

Consumer Behaviour Analysis

On the marketing front, SNA can help understand consumer behaviours, preferences, and their decision-making process. By studying consumer networks, organisations can identify influences that impact purchasing decisions or track the diffusion of new product knowledge. With this, companies can serve more targeted advertisements and understand the potential buyer's journey.

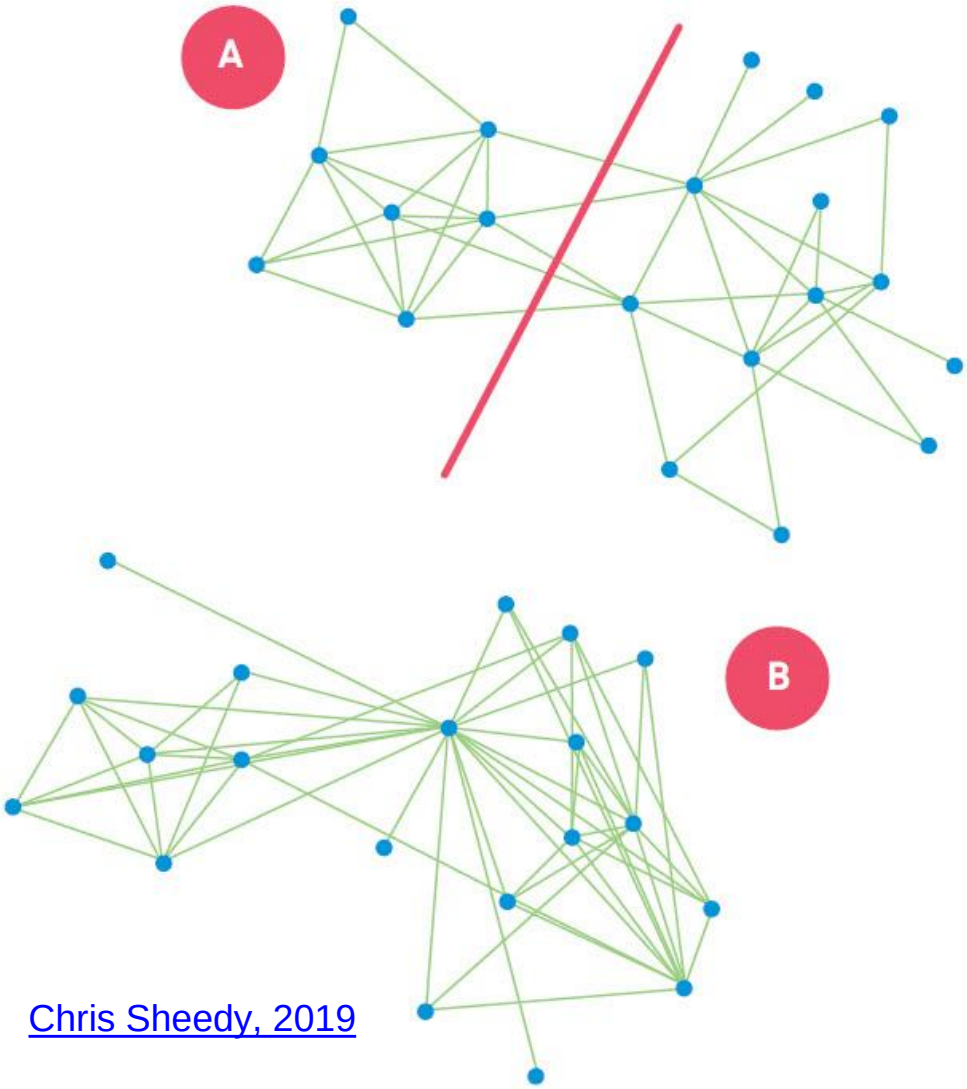
Reference: Study Smarter

Organizational Network Analysis



Source: https://www2.deloitte.com/us/en/pages/human-capital/articles/organizational-network-analysis.html

SNA in Teams



- Scenario (A), the group was split into two subgroups that were relatively comfortable communicating with each other.
- The group had, in fact, been two teams, brought together under a single manager. They were co-located, but still largely working as two separate groups.
- The company's management thought team building might help bring the subgroups together.
- Scenario (B) shows the team three months later. There were more connections within the group, with one individual particularly pivotal in unifying the team.
- What caused this change? Not traditional team-building exercises, but "targeted self-disclosure exercises".

Chris Sheedy, 2019

Building Trust Through Skillful Self-Disclosure

SNA in Medicine

Day nurse 1 > Night nurse 1	never gave info (0)
Day nurse 1 > Night nurse 2	often gave info (3)
Day nurse 1 > Night nurse 3	constantly gave info (4)
etc. for all night nurses	
Day nurse 2 > Night nurse 1	seldom gave info (1)
Day nurse 2 > Night nurse 2	seldom gave info (1)
Day nurse 2 > Night nurse 3	constantly gave info (4)
etc. for all night nurses	

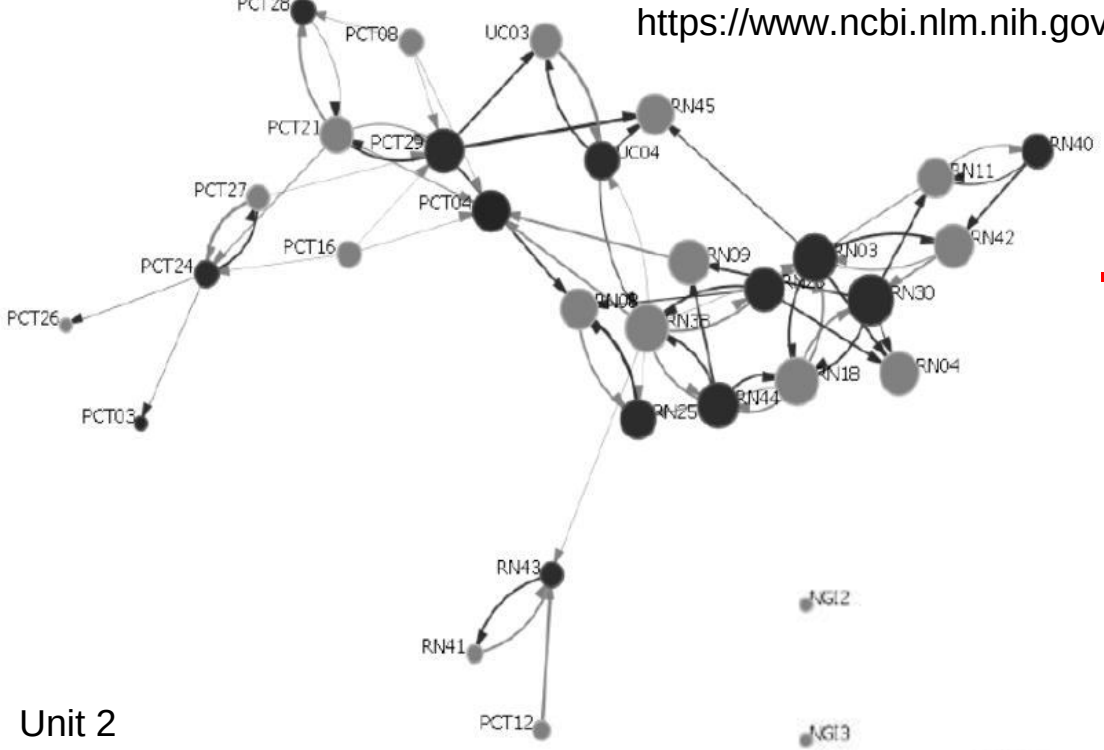
then etc. for all day nurses

https://www.statistics.com/social-network-analysis-sna-in-medicine/

https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4156878/

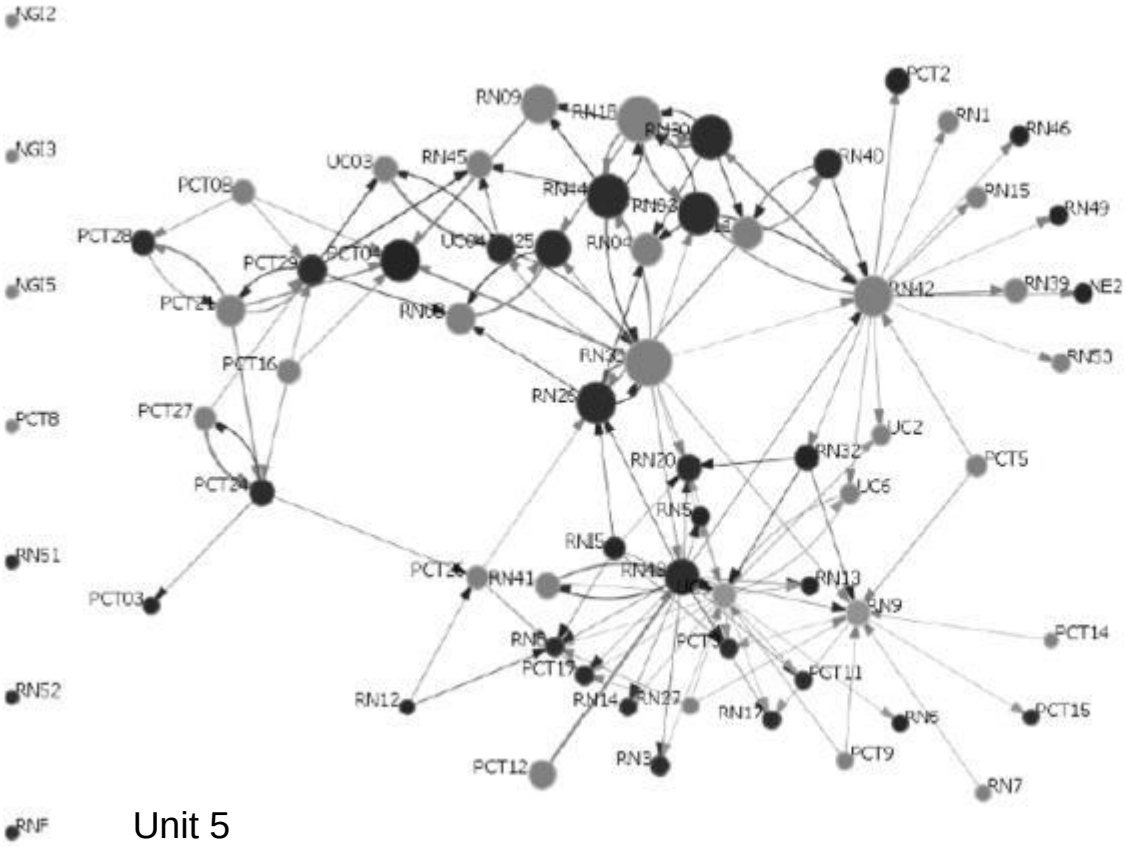
innovate achieve lead

- PCT: Patient Care Technician
- RN: Registered Nurse
- Gray circles: Day shift
- Dark circles: Night shift
- Thicker links: Higher comm frequency
- Larger node size represents higher Eigenvector Centrality values



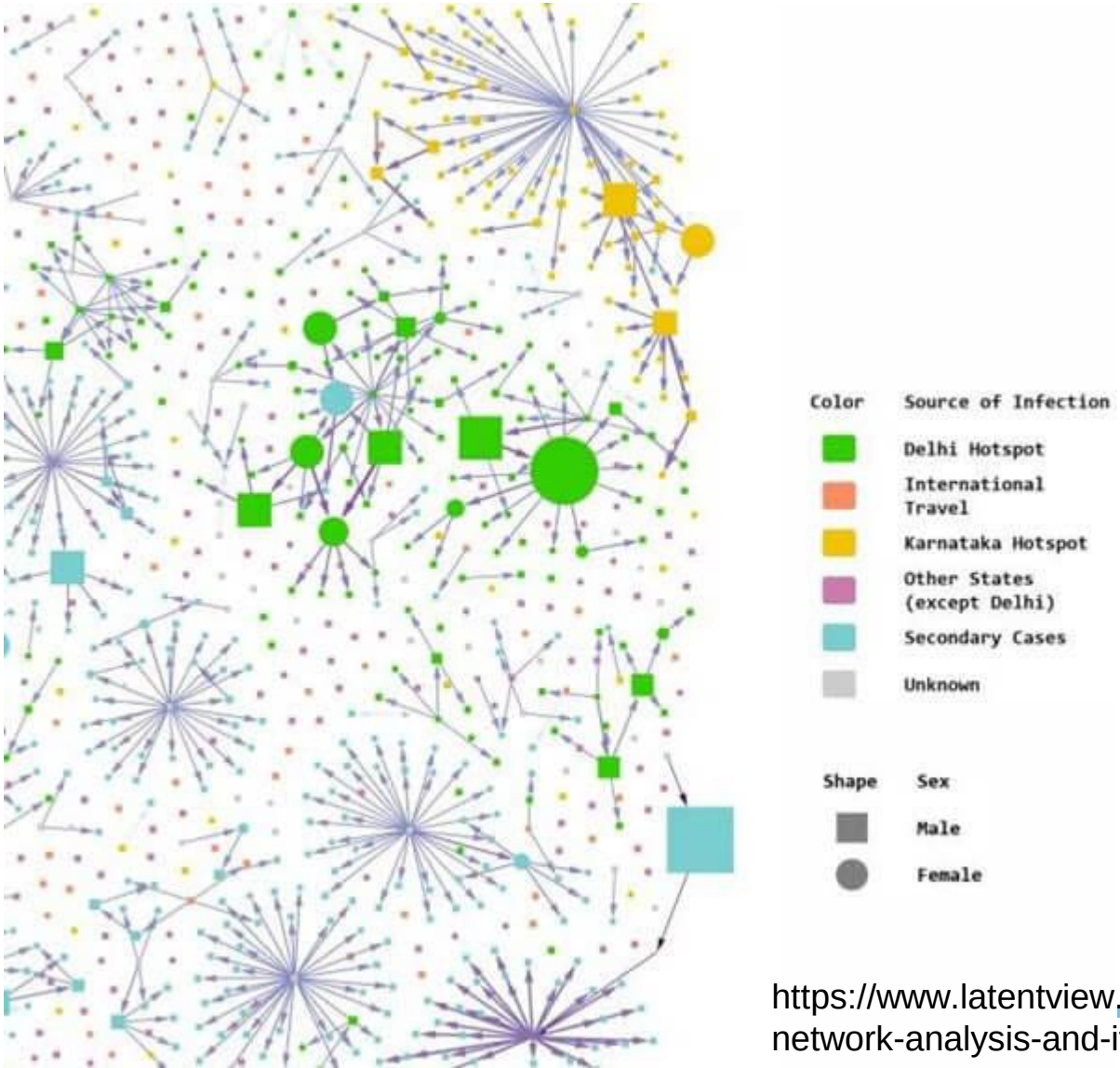
Unit 2

- Unit 2's diffusion metric is nearly twice as high (.43) as that for the much larger Unit 5 (.28). Information might diffuse quickly across the network of Unit 2
- The average Eigenvector Centrality value for Unit 2 is .45, compared with that of Unit 5 (.20). This suggests that on the smaller unit, more individuals are connected to highly connected staff.
- In Unit 5, two of the more influential RNs (high Eigenvector Centrality) are not communicating with staff not on their shift, and there are a number of "pendants" (people with single links). The pendants are usually PCTs.



Unit 5

SNA for Infectious Disease Tracking



https://www.latentview.com/blog/a-guide-to-social-network-analysis-and-its-use-cases/

Diffusion of microfinance

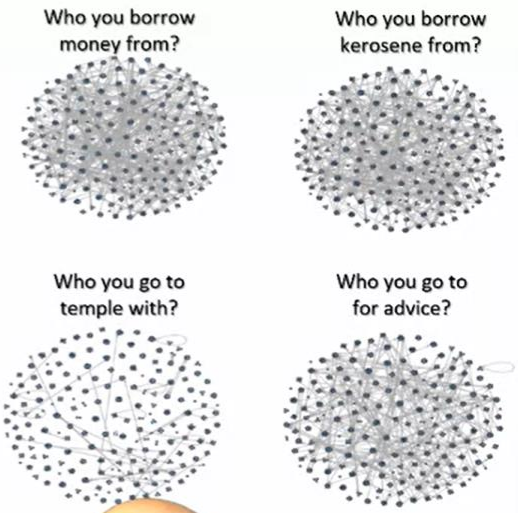
Centrality Application

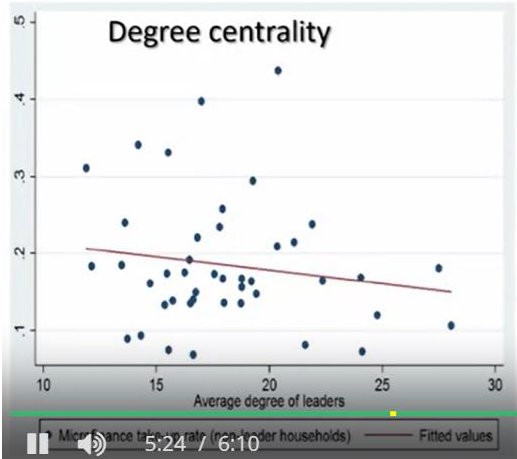
➢ **Diffusion of microfinance**

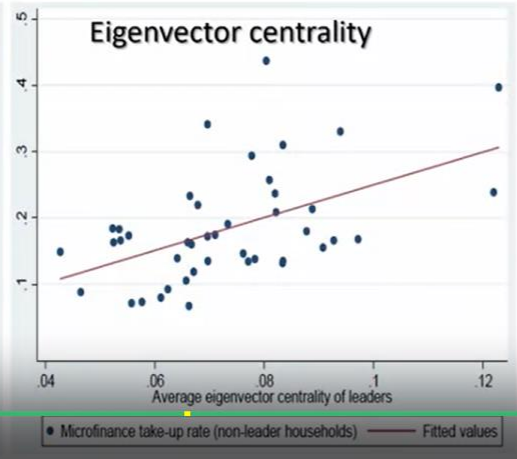
Banerjee, Chandrasekhar, Duflo & Jackson (2013). The Diffusion of Microfinance. *Science*, *341*(6144), 1236498. https://doi.org/10.1126/science.1236498

- 75 rural villages in Karnataka/ India, without microfinance
- Bank entered 43 of them and offered microfinance

- Question:
  Who to contact first to spread the innovation?
- Challenge:
  How to map the network: "who would you borrow from?"
  …they created 13 different/ multiplex networks…



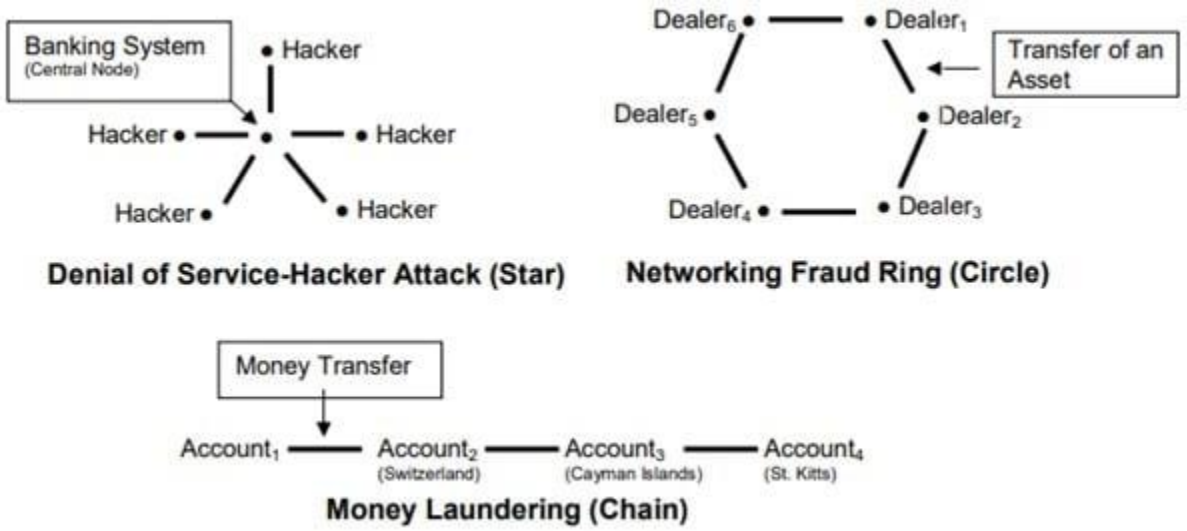




=> contact those whose friends have many friends!

Reference: https://www.coursera.org/learn/social-network-analysis/home/week/2

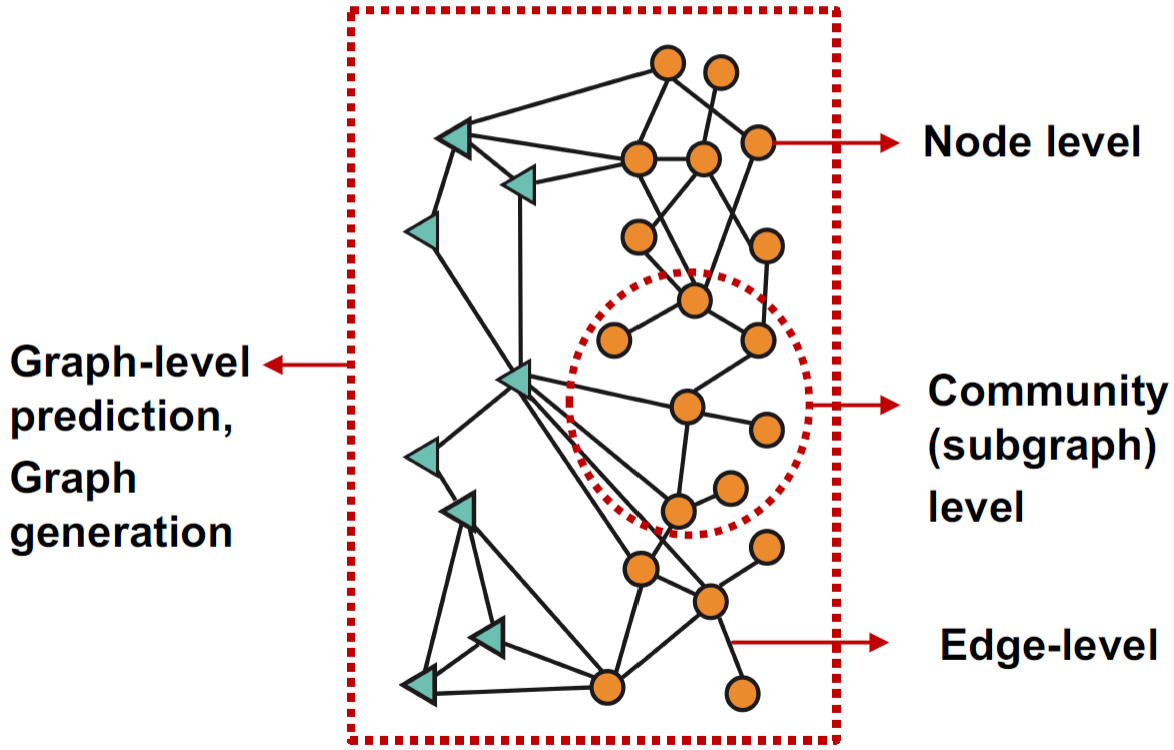
SNA in Finance / Fraud Detection



https://www.latentview.com/blog/a-guide-to-social-network-analysis-and-its-use-cases/

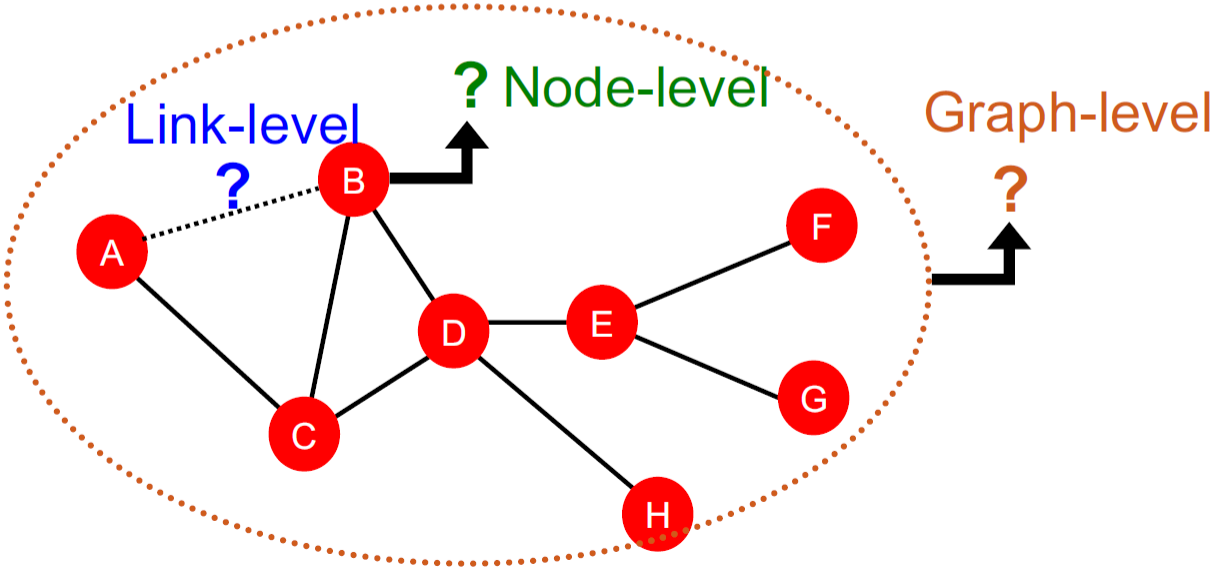
Graph Representation Learning

Different Types of Tasks

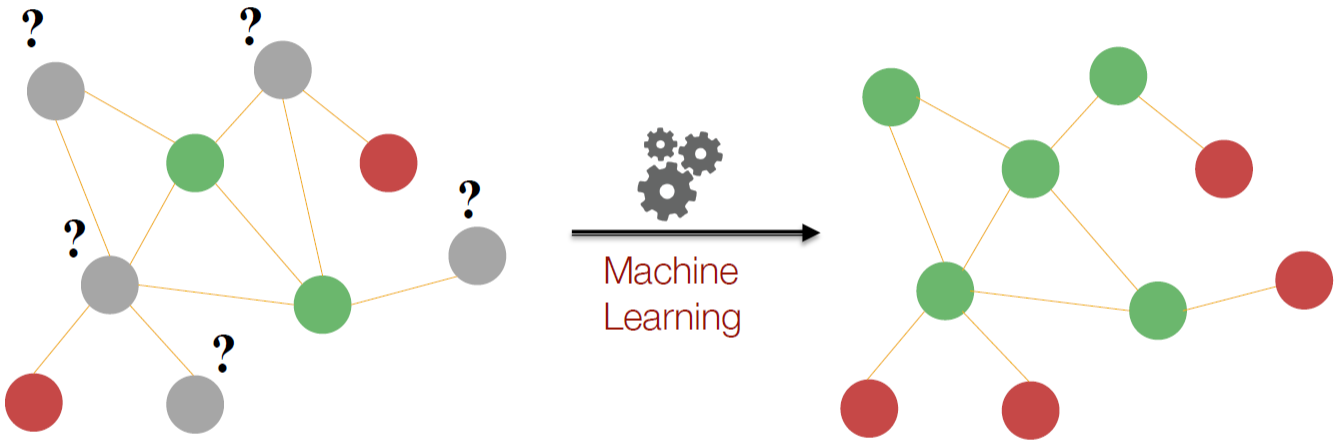


Machine Learning Tasks: Review

- Node-level prediction
- Link-level prediction
- Graph-level prediction



Node-Level Tasks

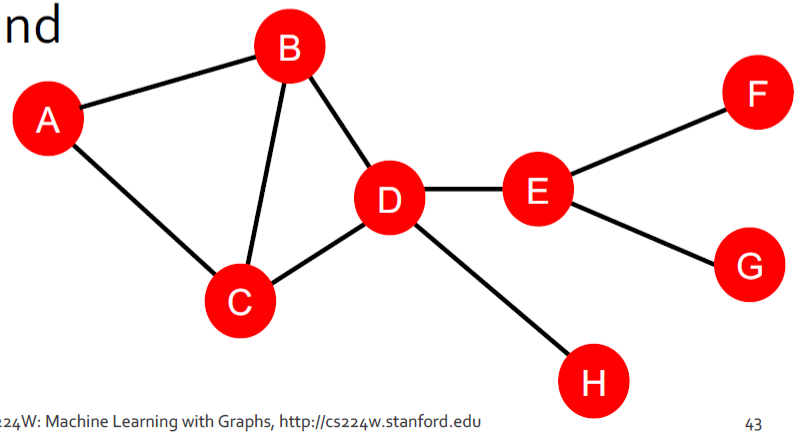


Node classification

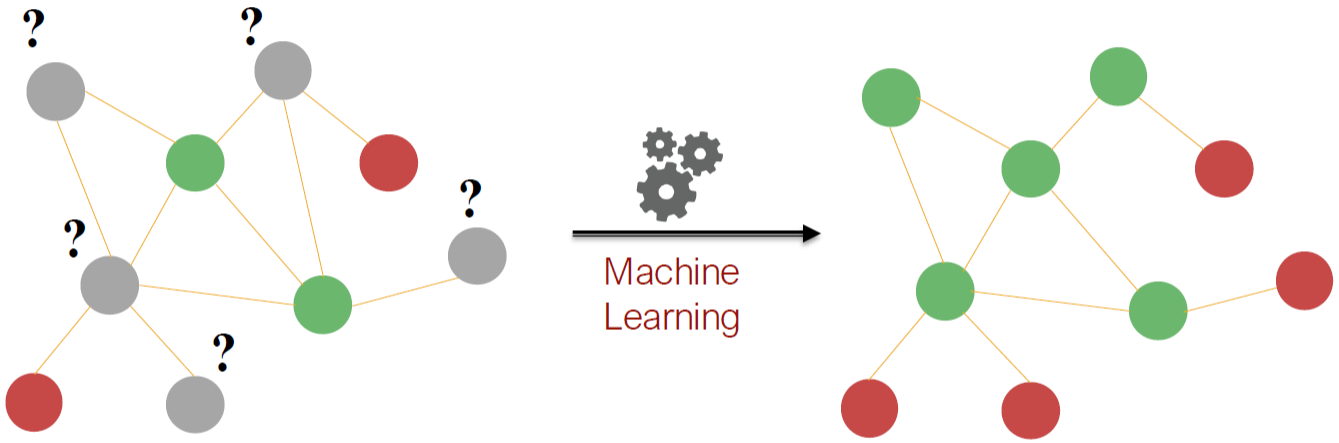
Node-Level Network Structure

Goal: Characterize the structure and position of a node in the network:

- Node degree
- Node importance & position
  - E.g., Number of shortest paths passing through a node
  - E.g., Avg. shortest path length to other nodes
- Substructures around the node



Node-Level Tasks



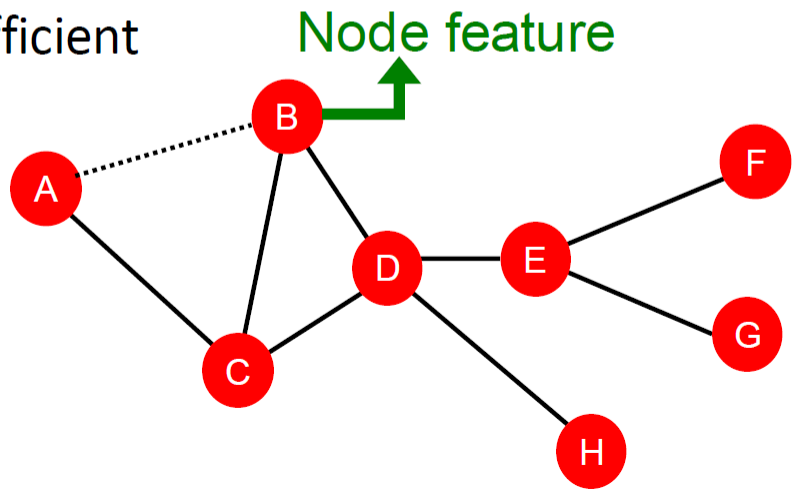
Node classification

ML needs features.

Node-Level Features: Overview

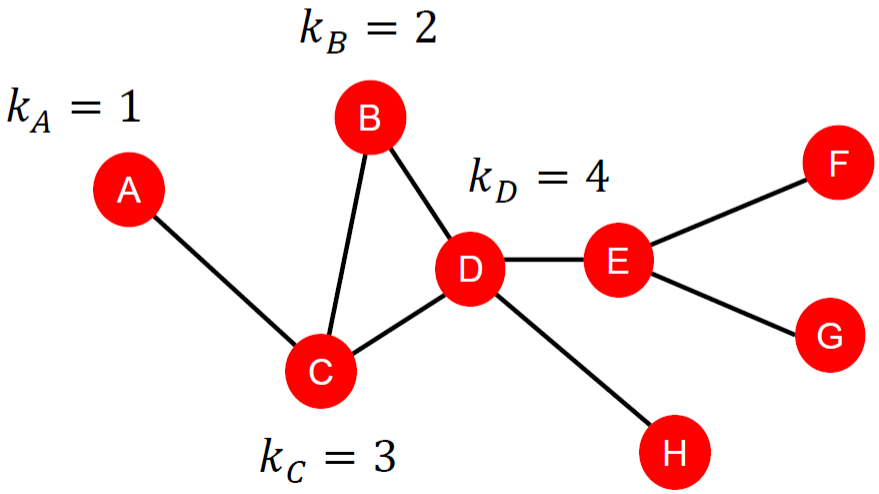
Goal: Characterize the structure and position of a node in the network:

- Node degree
- Node centrality
- Clustering coefficient
- Graphlets



Node Features: Node Degree

- The degree k_v of node v is the number of edges (neighboring nodes) the node has.
- Treats all neighboring nodes equally.



Node Features: Node Centrality

- Node degree counts the neighboring nodes without capturing their importance.
- Node centrality c_v takes the node importance in a graph into account
- **Different ways to model importance:**
  - Eigenvector centrality
  - Betweenness centrality
  - Closeness centrality
  - and many others…

Node Centrality (1)

- **Eigenvector centrality:**
  - A node v is important if surrounded by important neighboring nodes $u \in N(v)$.
  - We model the centrality of node v as the sum of the centrality of neighboring nodes:

$$c_v = \frac{1}{\lambda} \sum_{u \in N(v)} c_u$$

λ is normalization constant (it will turn out to be the largest eigenvalue of A)

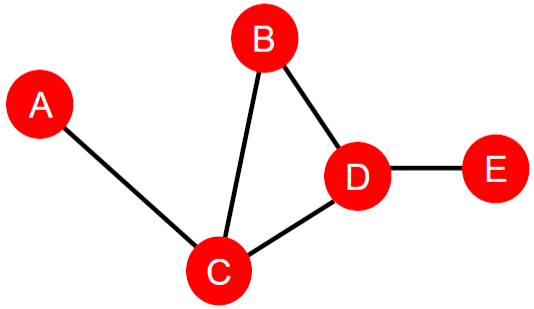
  - Notice that the above equation models centrality in a recursive manner. **How do we solve it?**

Node Centrality (2)

- **Betweenness centrality:**
  - A node is important if it lies on many shortest paths between other nodes.

$$c_v = \sum_{s \neq v \neq t} \frac{\#(\text{shortest paths betwen } s \text{ and } t \text{ that contain } v)}{\#(\text{shortest paths between } s \text{ and } t)}$$

  - **Example:**



$c_A = c_B = c_E = 0$

$c_C = 3$

(A-C-B, A-C-D, A-C-D-E)

$c_D = 3$

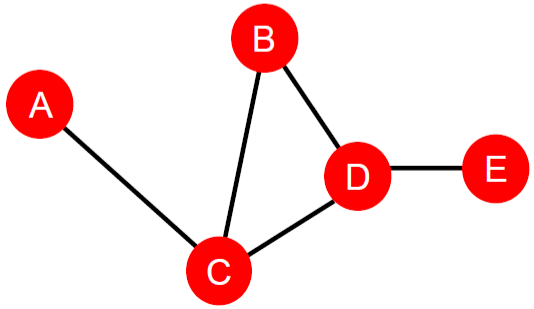
(A-C-D-E, B-D-E, C-D-E)

Node Centrality (3)

- **Closeness centrality:**
  - A node is important if it has small shortest path lengths to all other nodes.

$$c_v = \frac{1}{\sum_{u \neq v} \text{shortest path length between } u \text{ and } v}$$

  - **Example:**



$c_A = 1/(2+1+2+3) = 1/8$
(A-C-B, A-C, A-C-D, A-C-D-E)

$c_D = 1/(2+1+1+1) = 1/5$
(D-C-A, D-B, D-C, D-E)

Node Features: Clustering Coefficient

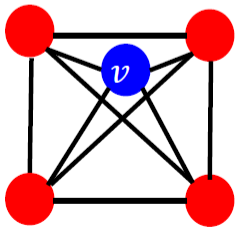
- Measures how connected $v's$ neighboring nodes are:

$$e_v = \frac{\#(\text{edges among neighboring nodes})}{\binom{k_v}{2}} \in [0,1]$$

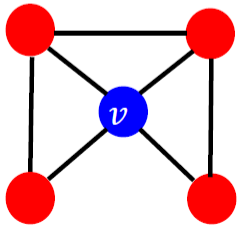
$\#(\text{node pairs among } k_v \text{ neighboring nodes})$

In our examples below the denominator is 6 (4 choose 2).

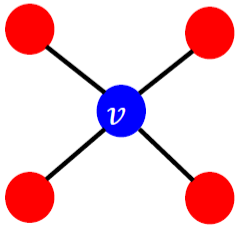
- **Examples:**



$e_v = 1$



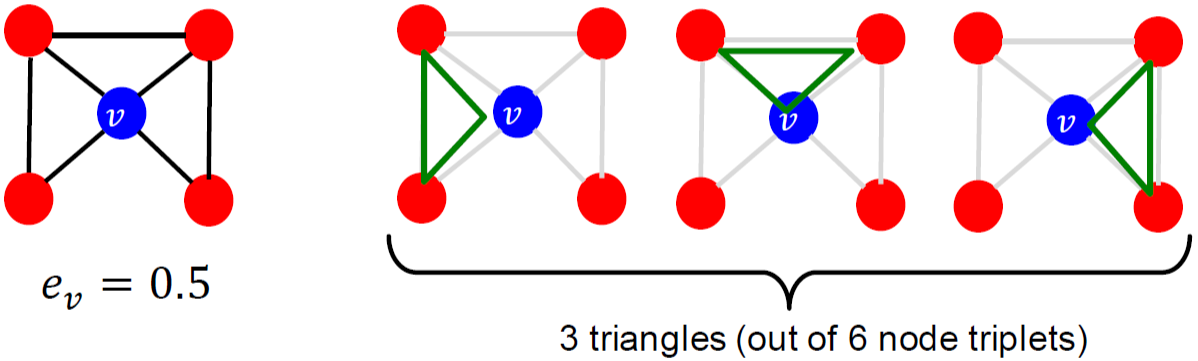
$e_v = 0.5$



$e_v = 0$

Node Features: Graphlets

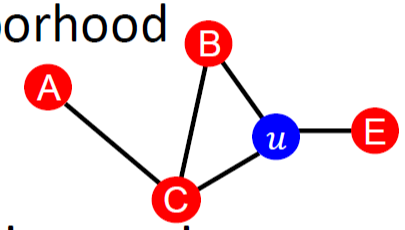
- **Observation:** Clustering coefficient counts the #(triangles) in the ego-network



- We can generalize the above by counting #(pre-specified subgraphs, i.e., **graphlets**).

Node Features: Graphlets

- **Goal:** Describe network structure around node u
  - **Graphlets** are small subgraphs that describe the structure of node u's network neighborhood



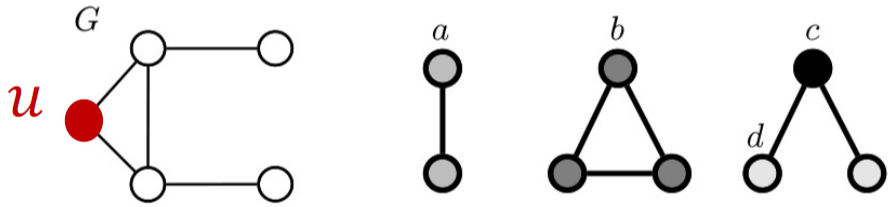
Analogy:

- **Degree** counts **#(edges)** that a node touches
- **Clustering coefficient** counts **#(triangles)** that a node touches.
- **Graphlet Degree Vector (GDV):** Graphlet-base features for nodes
  - **GDV** counts **#(graphlets)** that a node touches

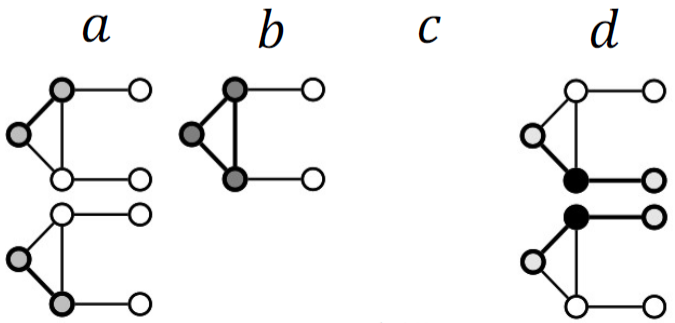
Node's Subgraphs: Graphlets

- **Graphlets:** A count vector of rooted subgraphs at a given node.
- **Example:**

All possible graphlets on up to 3 nodes



Graphlet instances of node u:



Graphlets of node u:
a, b, c, d
[2,1,0,2]

Node-Level Feature: Summary

- We have introduced different ways to obtain node features.
- They can be categorized as:
  - Importance-based features:
    - Node degree
    - Different node centrality measures
  - Structure-based features:
    - Node degree
    - Clustering coefficient
    - Graphlet count vector

Node-Level Feature: Summary

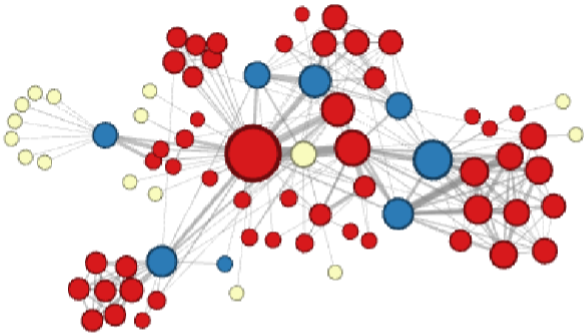
- **Importance-based features**: capture the importance of a node in a graph
  - Node degree:
    - Simply counts the number of neighboring nodes
  - Node centrality:
    - Models importance of neighboring nodes in a graph
    - Different modeling choices: eigenvector centrality, betweenness centrality, closeness centrality
- Useful for predicting influential nodes in a graph
  - **Example:** predicting celebrity users in a social network

Node-Level Feature: Summary

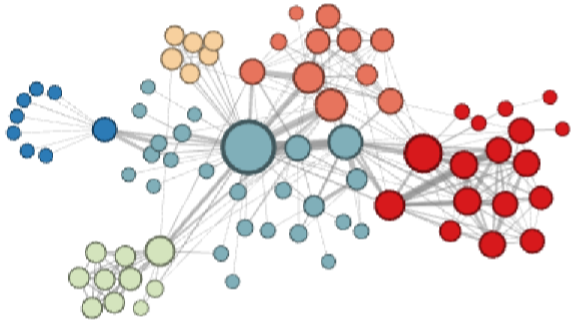
- **Structure-based features**: Capture topological properties of local neighborhood around a node.
  - **Node degree:**
    - Counts the number of neighboring nodes
  - **Clustering coefficient:**
    - Measures how connected neighboring nodes are
  - **Graphlet degree vector:**
    - Counts the occurrences of different graphlets
- **Useful for predicting a particular role a node plays in a graph:**
  - **Example:** Predicting protein functionality in a protein-protein interaction network.

Discussion

Different ways to label nodes of the network:



Node features defined so far would allow to distinguish nodes in the above example

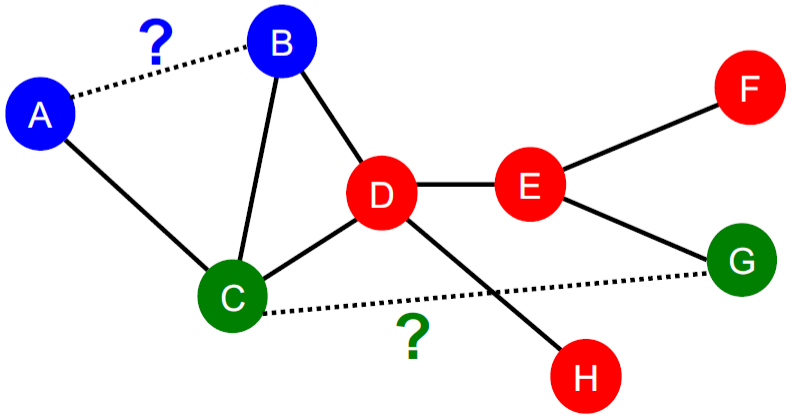


However, the features defines so far would not allow for distinguishing the above node labelling

Link Prediction

Link-Level Prediction Task

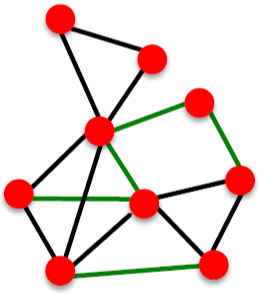
- The task is to predict **new/missing/unknown links** based on the existing links.
- At test time, node pairs (with no existing links) are ranked, and top K node pairs are predicted.
- Task: Make a prediction for **a pair of nodes**.



Link Prediction as a Task

Two formulations of the link prediction task:

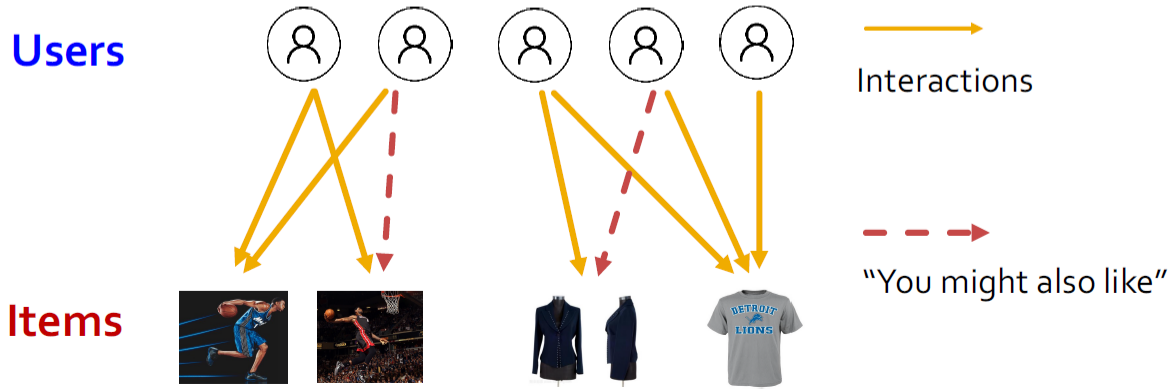
- **1) Links missing at random:**
  - Remove a random set of links and then aim to predict them
- **2) Links over time:**
  - Given $G[t_0, t_0']$ a graph defined by edges up to time t_0', **output a ranked list L** of edges (not in $G[t_0, t_0']$) that are predicted to appear in time $G[t_1, t_1']$
  - **Evaluation:**
    - $n = |E_{new}|$: # new edges that appear during the test period $[t_1, t_1']$
    - Take top n elements of L and count correct edges



$G[t_0, t_0']$
$G[t_1, t_1']$

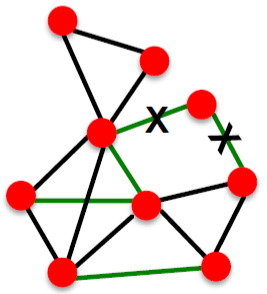
Example (1): Recommender Systems

- **Users interacts with items**
  - Watch movies, buy merchandise, listen to music
  - **Nodes:** Users and items
  - **Edges:** User-item interactions
- **Goal: Recommend items users might like**



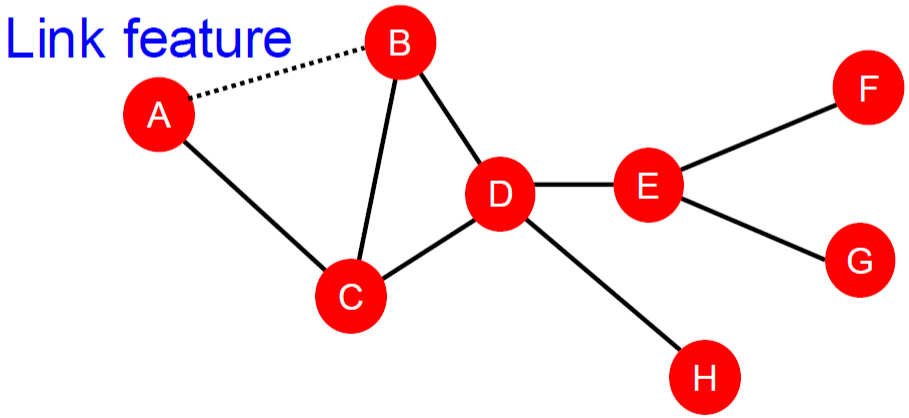
Link Prediction via Proximity

- **Methodology:**
  - For each pair of nodes *(x,y)* compute score *c(x,y)*
    - For example, *c(x,y)* could be the # of common neighbors of *x* and *y*
  - Sort pairs *(x,y)* by the decreasing score *c(x,y)*
  - **Predict top n pairs as new links**
  - **See which of these links actually appear in $G[t_1, t_1']$**



Link-Level Features: Overview

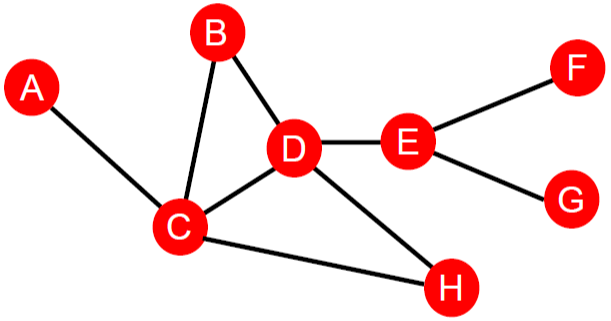
- Distance-based feature
- Local neighborhood overlap
- Global neighborhood overlap



Distance-Based Features

Shortest-path distance between two nodes

- Example:



$$S_{BH} = S_{BE} = S_{AB} = 2$$
$$S_{BG} = S_{BF} = 3$$

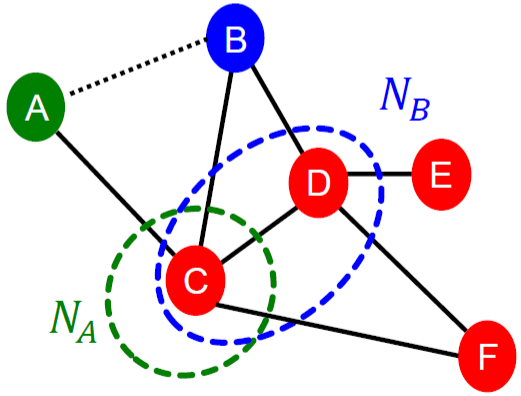
- However, this does not capture the degree of neighborhood overlap:
  - Node pair *(B, H)* has 2 shared neighboring nodes, while pairs *(B, E)* and *(A, B)* only have 1 such node.

Local Neighborhood Overlap

Captures # neighboring nodes shared between two nodes v_1 and v_2:

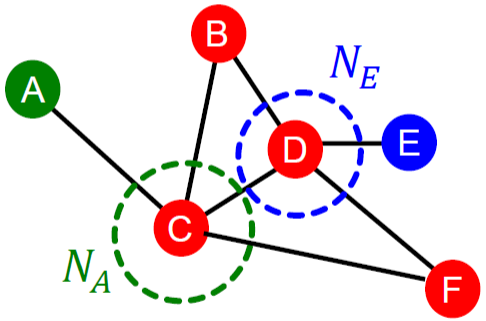
- **Common neighbors:** $|N(v_1) \cap N(v_2)|$
  - Example: $|N(A) \cap N(B)| = |\{C\}| = 1$
- **Jaccard's coefficient:** $\frac{|N(v_1) \cap N(v_2)|}{|N(v_1) \cup N(v_2)|}$
  - Example: $\frac{|N(A) \cap N(B)|}{|N(A) \cup N(B)|} = \frac{|\{C\}|}{|\{C,D\}|} = \frac{1}{2}$
- **Adamic-Adar index:**

  $\sum_{u \in N(v_1) \cap N(v_2)} \frac{1}{\log(k_u)}$
  - Example: $\frac{1}{\log(k_C)} = \frac{1}{\log 4}$



Global Neighborhood Overlap

- **Limitation of local neighborhood features:**
  - Metric is always zero if the two nodes do not have any neighbors in common.



$$N_A \cap N_E = \phi$$
$$|N_A \cap N_E| = 0$$

  - However, the two nodes may still potentially be connected in the future.
- **Global neighborhood overlap** metrics resolve the limitation by considering the entire graph.

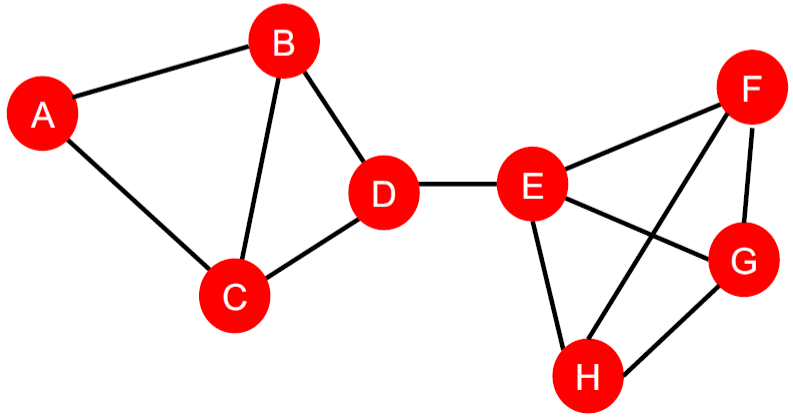
Link-Level Features: Summary

- **Distance-based features:**
  - Uses the shortest path length between two nodes but does not capture how neighborhood overlaps.
- **Local neighborhood overlap:**
  - Captures how many neighboring nodes are shared by two nodes.
  - Becomes zero when no neighbor nodes are shared.
- **Global neighborhood overlap:**
  - Uses global graph structure to score two nodes.
  - Katz index counts #walks of all lengths between two nodes.

Graph-Level Features

- **Goal:** We want make a prediction for an entire graph or a subgraph of the graph.

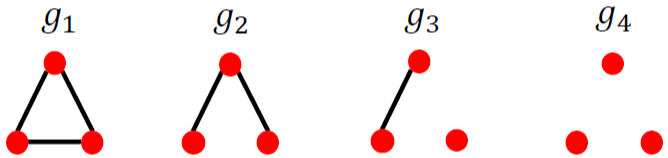
- **For example:**



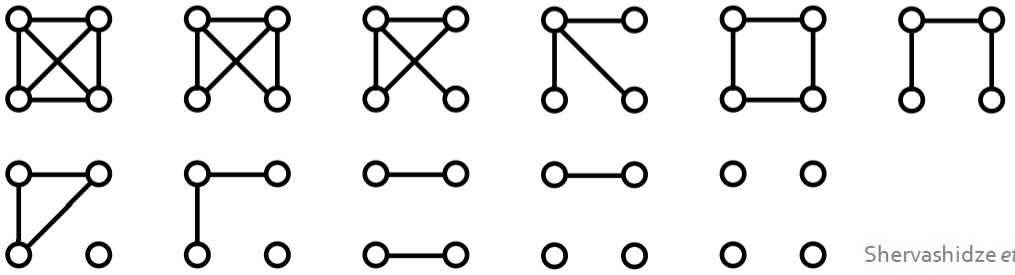
Graph-Level Graphlet Features

Let $G_k = (g_1, g_2, \ldots, g_{n_k})$ be a list of graphlets of size k.

- For $k = 3$, there are 4 graphlets.



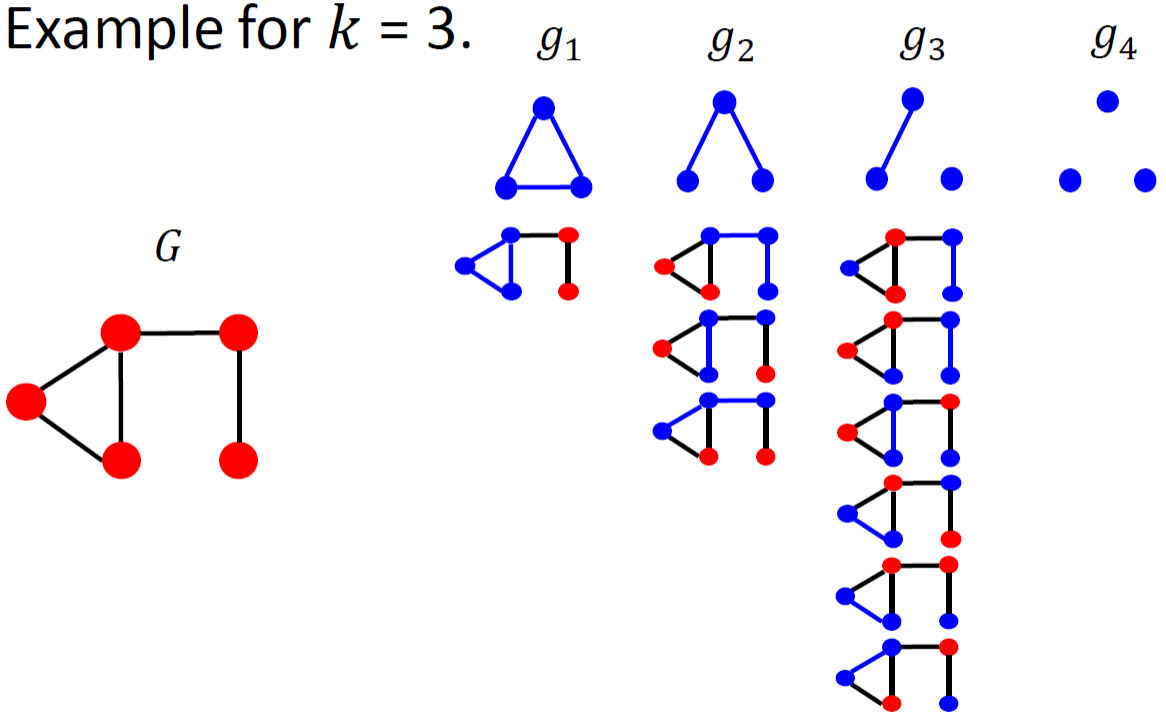
- For $k = 4$, there are 11 graphlets.



Shervashidze *et al.*, AISTATS 2011

Graph-Level Graphlet Features

- Example for $k = 3$.



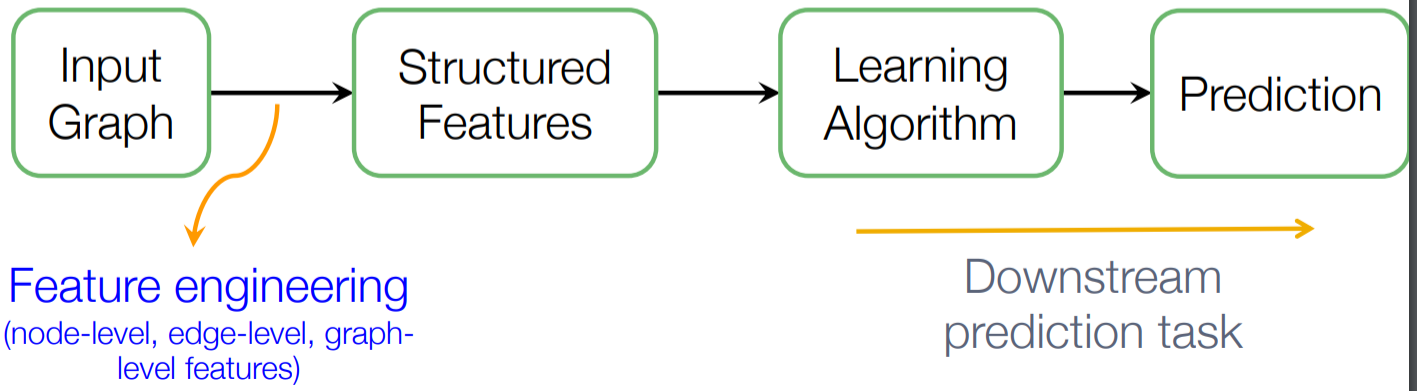
$$f_G = (1, \quad 3, \quad 6, \quad 0)^T$$

Today's Summary

- **Traditional ML Pipeline**
  - Hand-crafted (structural) features + ML model
- **Hand-crafted features for graph data**
  - **Node-level:**
    - Node degree, centrality, clustering coefficient, graphlets
  - **Link-level:**
    - Distance-based feature
    - local/global neighborhood overlap
  - **Graph-level:**
    - Graphlet kernel, WL kernel
- However, we only considered featurizing the graph structure (but not the attribute of nodes and their neighbors)

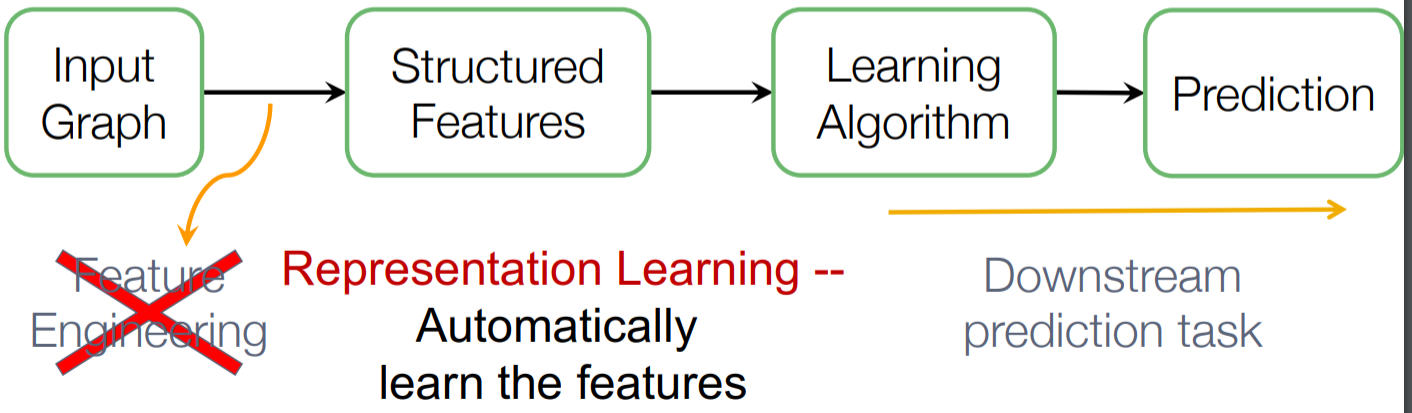
Recap: Traditional ML for Graphs

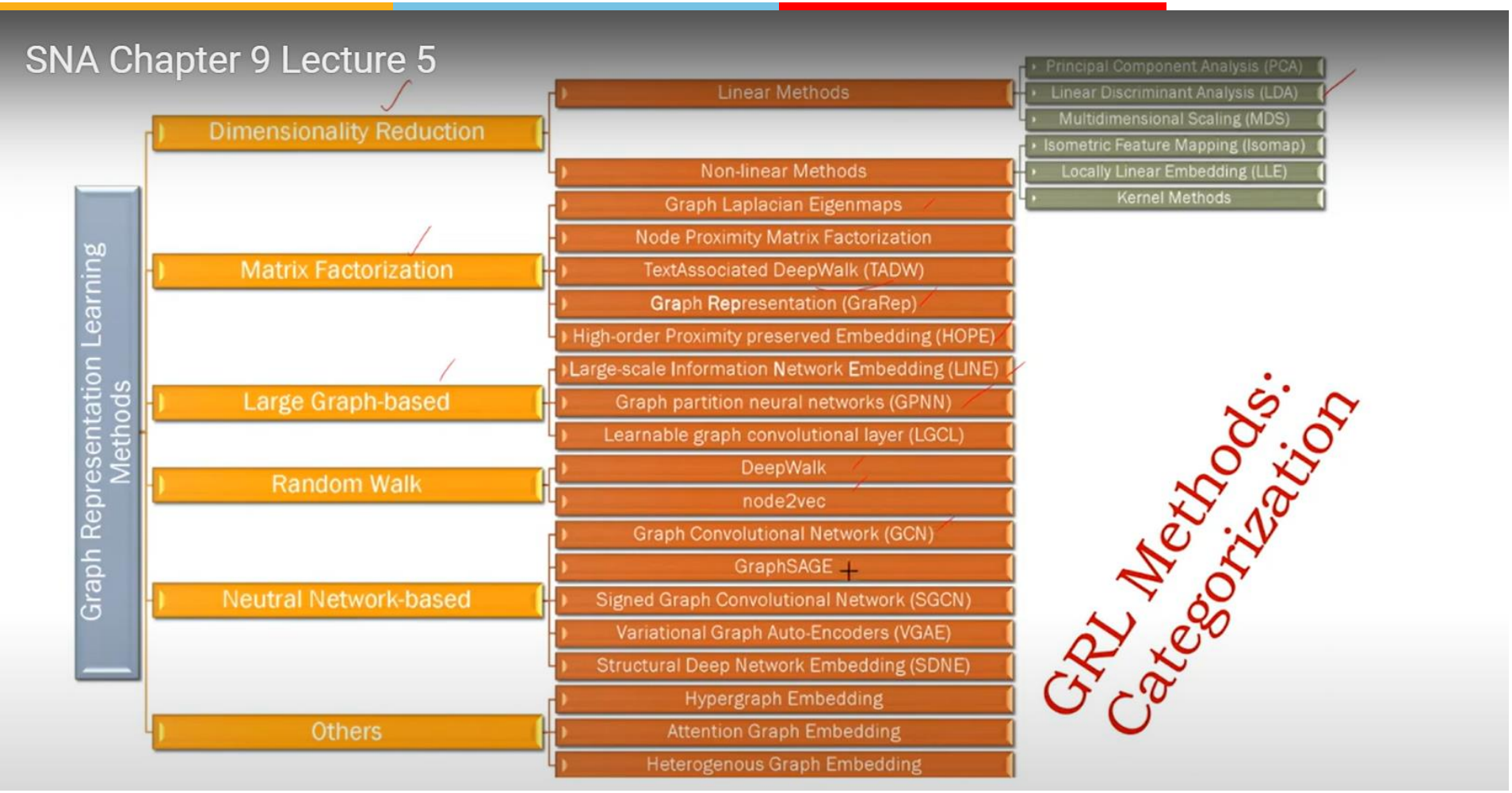
Given an input graph, extract node, link and graph-level features, then learn a model (SVM, neural network, etc.) that maps features to labels.



Graph Representation Learning

Graph Representation Learning alleviates the need to do feature engineering every single time.





Reference: Tanmoy Chakraborty, https://www.youtube.com/watch?v=NKHgftvRZKM

Thank you