CENTRALITY MEASURES: A TOOL TO IDENTIFY KEY ACTORS IN SOCIAL NETWORKS

DR. JAYA TANGIRALA

CENTRALITY MEASURES

- Centrality measures address the question: **‘What is the most important / central node in this network?’**
- There are many answers to this question, depending on what we mean by importance.
- These measures help us understand which nodes are the **most influential, well-connected, or central** in a social network.
- There are several different centrality measures, each focusing on different aspects of centrality

Centrality Measures

- Two types of measures
  - Local: Involves only immediate neighborhood
  - Global: Involves the whole graph

DEGREE CENTRALITY

Degree Centrality: Undirected Networks

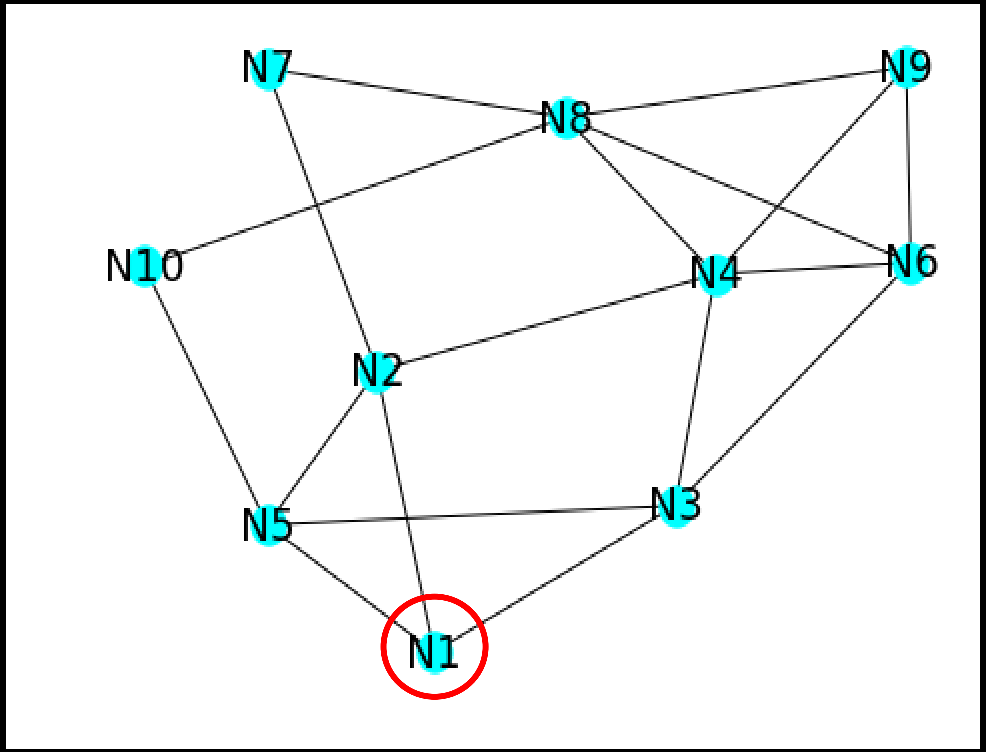
- **Assumption:** Nodes adjacent to many other nodes play an important role in a network.
- **Degree centrality** focuses on individual nodes
- Degree centrality of node v is computed as the fraction of other nodes adjacent to node v out of total possible.

$$C_d(v) = \frac{d_v}{n-1}$$

- Nodes with high degree centrality are called as **Hub** nodes

DEGREE CENTRALITY: EXAMPLE

$$C_d(v) = \frac{d_v}{n-1}$$



Node	d_v	$C_d(v)$
N1	3	3/9 = 0.33
N2	4	4/9
N3	4	4/9
N4	5	5/9
N5	4	4/9
N6	4	4/9
N7	2	2/9
N8	5	5/9
N9	3	3/9
N10	2	2/9

Degree Centrality

- Range:
    - Simple graphs: 0 to 1
    - For multigraphs or graphs with self-loops the maximum degree might be higher than n-1 and values of degree centrality greater than 1 are possible.
- In directed networks
    - In degree centrality
    - Out degree centrality
- Disconnected networks
    - Isolated nodes, it is 0
    - We can limit it to the nodes in the largest connected component
- Weighted Networks
    - Consider weight in place of adjacency

Closeness Centrality

Closeness Centrality: Undirected Networks

- **Assumption**: Node which is close to all nodes is central.
- Closeness centrality is a way of detecting nodes that can spread information very efficiently through the network.
- The closeness centrality of node *v* in a graph is the average shortest path distance to *v* over all *n-1* reachable nodes and then taking a reciprocal of it.

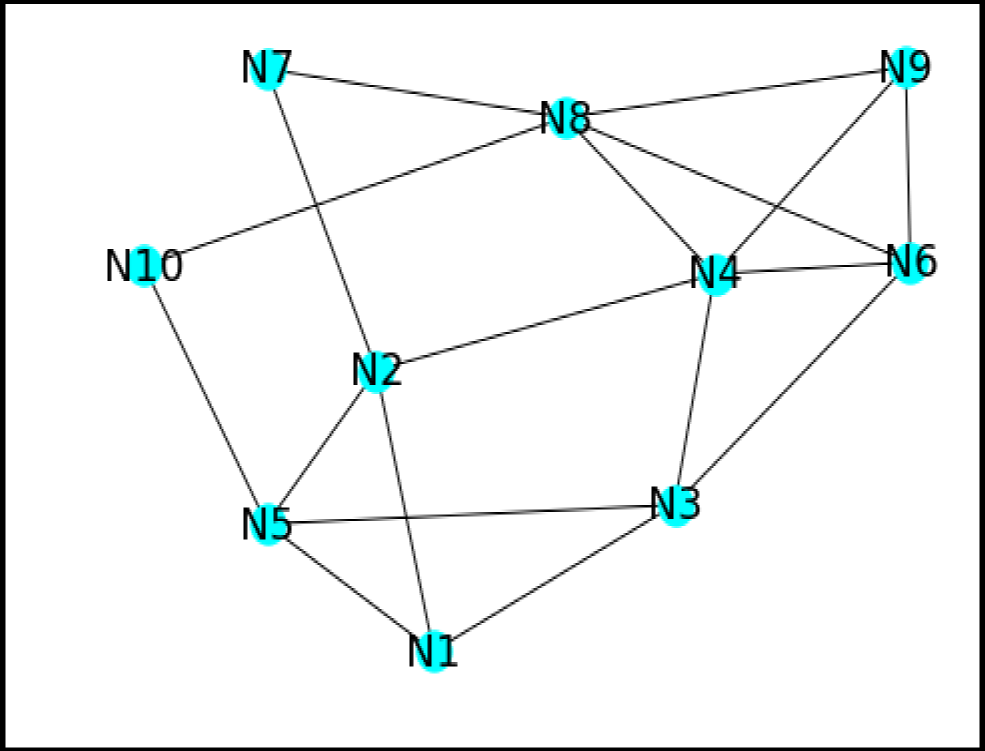
$$C_c(v) = \frac{n-1}{\sum_{u \neq v} d(v,u)}$$

where d(v,u) is the length of shortest path from *v* to *u*.

- The node with the **highest closeness centrality** is the **closest one** to all other nodes.

CLOSENESS CENTRALITY: EXAMPLE

$$C_c(v) = \frac{n-1}{\sum_{u \neq v} d(v,u)}$$



Node	Shortest path originating from N1	Shortest path length
N1	-	0
N2	N1-N2	1
N3	N1-N3	1
N4	N1-N2-N4	2
N5	N1-N5	1
N6	N1-N3-N6	2
N7	N1-N2-N7	2
N8	N1-N2-N4-N8	3
N9	N1-N2-N4-N9	3
N10	N1-N5-N9	2

$\sum_{u \neq v} d(N1,u)$ = 1+1+2+1+2+2+3+3+2 =17

$$C_c(N1) = \frac{n-1}{\sum_{u \neq v} d(N1,u)} = \frac{9}{17} = 0.5294$$

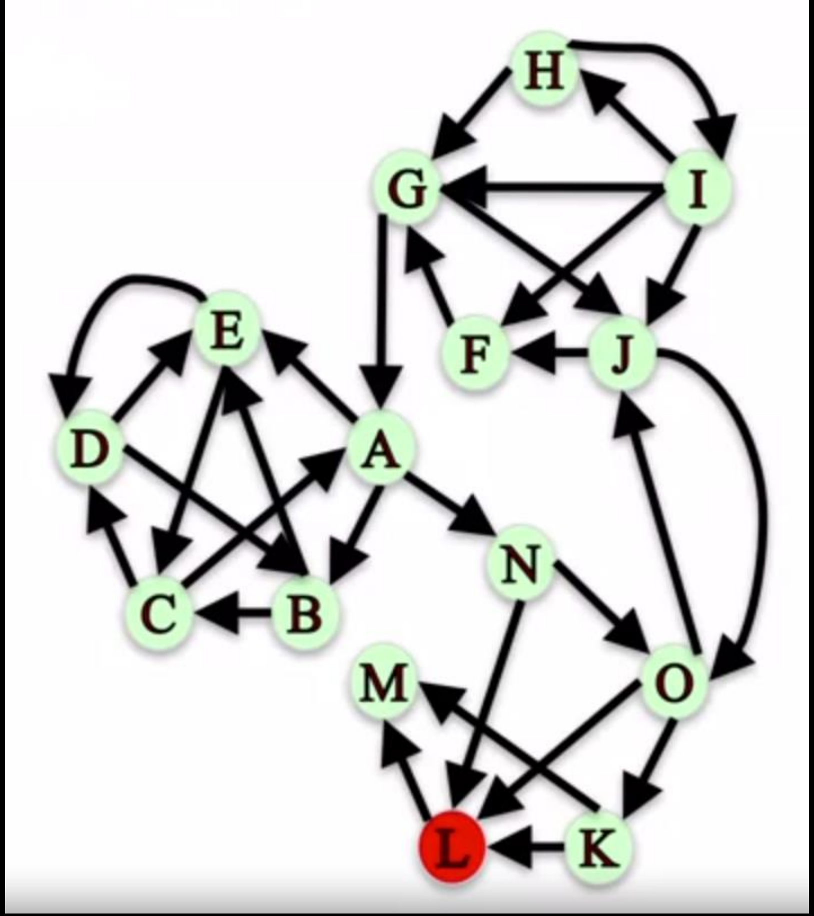
Closeness Centrality: Directed Networks

- Definition is same, considering directed paths

$$C_c(v) = \frac{n-1}{\sum_{u \neq v} d(v,u)}$$

where d(v,u) is the length of shortest path **from *v* to *u*.**

Closeness Centrality: Disconnected Graphs



Two options:

- **Unnormalized**: Consider only the nodes reachable by v

$$C_c(v) = \frac{|R(v)|}{\sum_{u \neq v} d(v,u)}$$

where R(v) is set of nodes reachable by v.
Ex: In graph, $C_c(L)$ = 1/1 =1.

- **Normalized**: Normalize R(L) with the total number of nodes v ‘can’ reach

$$C_c(v) = \frac{|R(v)|}{n-1} * \frac{|R(v)|}{\sum_{u \neq v} d(v,u)}$$

Closeness Centrality: Weighted Graphs

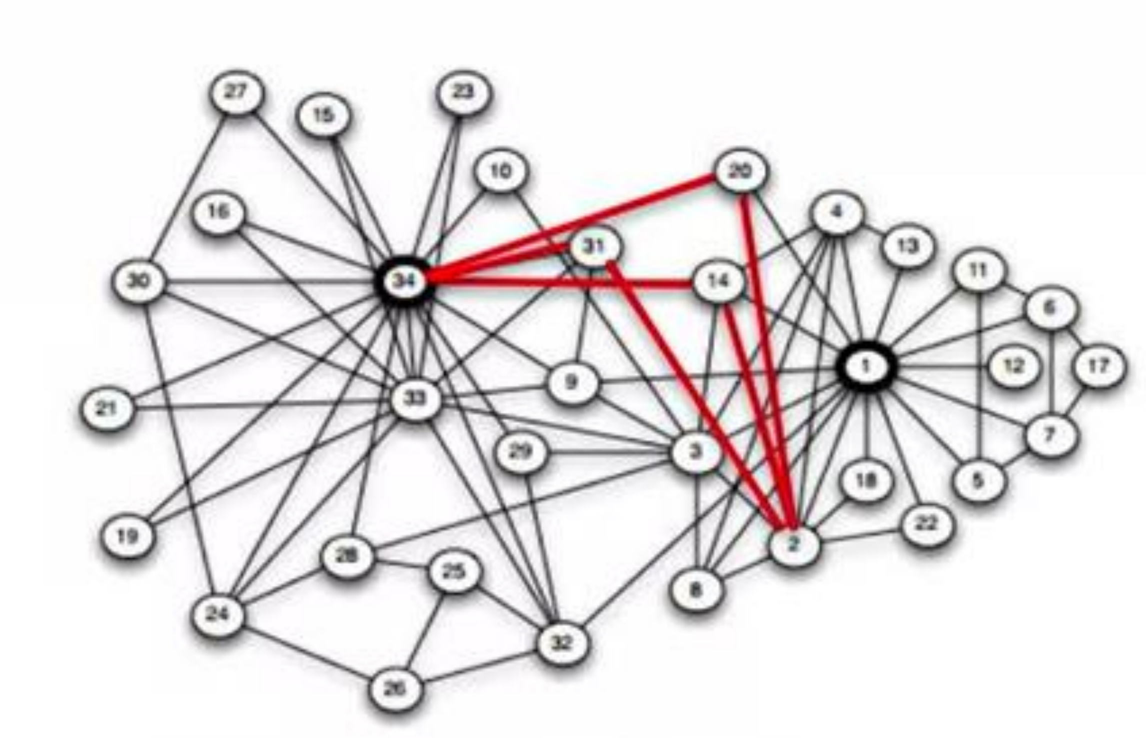
- Use weights to compute distance between nodes instead of path length.
- Ex: Dijkstra’s algorithm

BETWEENNESS CENTRALITY

Betweenness Centrality: Node

- Assumption: Notes that connect other nodes are central to the network

$$C_b(v) = \sum_{s,t \in V} \frac{\sigma_{s,t}(v)}{\sigma_{s,t}}$$

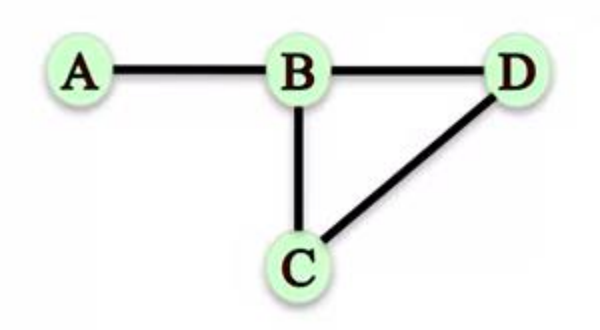


where
$\sigma_{s,t}$ is the number of shortest paths between s and t
$\sigma_{s,t}(v)$ is the number of shortest paths between
s and t passing through node v.

- Options: include /exclude v in computing shortest paths

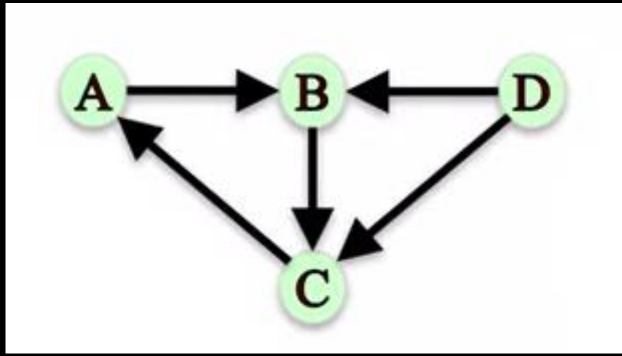
Example

- Excluding v
  - $C_b(B) = 2$
- Including B
  - $C_b(B) = 5$



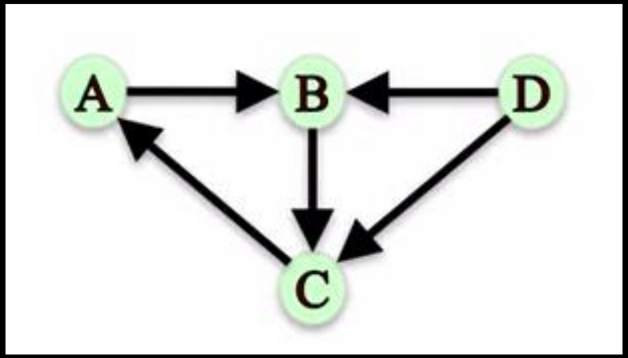
DISCONNECTED NODES

- Ex: Node D in the graph. No node can reach D.
- This implies denominator is 0.
- Therefore, if there is no shortest path between given nodes, Exclude that pair in computation.
- $C_b(C) = 2$ (excluding C)



Normalized Betweenness Centrality

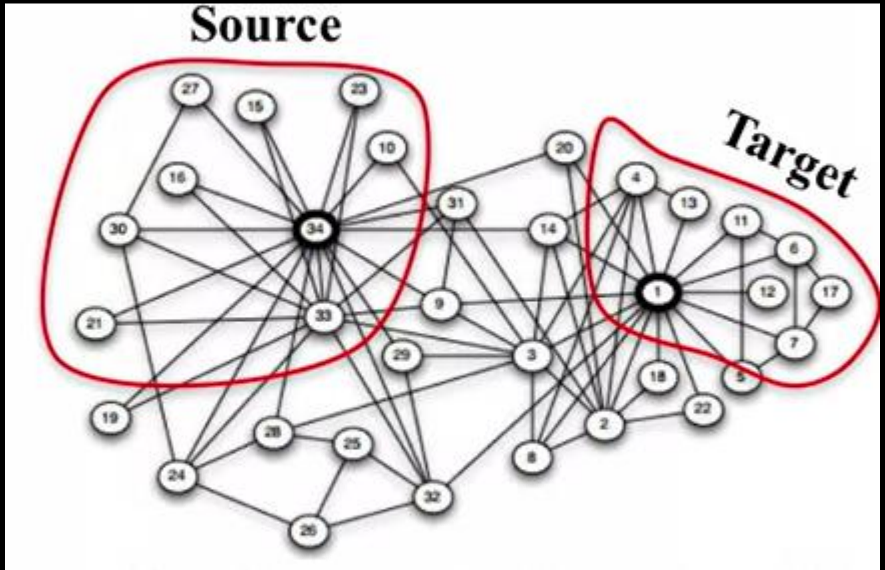
- Normalize by dividing centrality value by number of pairs in the network excluding that node.
  - Undirected network: $\frac{(n-1)(n-2)}{2}$
  - Directed network: $(n-1)(n-2)$
- Limitation: Computationally expensive
- Solution: Approximation
  - Compute C_b using only a sample of nodes rather than considering all pairs



BETWEENNESS CENTRALITY BETWEEN TWO SUBSETS

$$c_B(v) = \sum_{s \in S, t \in T} \frac{\sigma(s,t|v)}{\sigma(s,t)}$$

where S is the set of sources, T is the set of targets, $\sigma(s,t)$ is the number of shortest (s,t)-paths, and $\sigma(s,t|v)$ is the number of those paths passing through some node v other than s, t. If $s = t$, $\sigma(s,t) = 1$, and if $v \in s, t$, $\sigma(s,t|v) = 0$ [2].



Betweenness Centrality: Edge

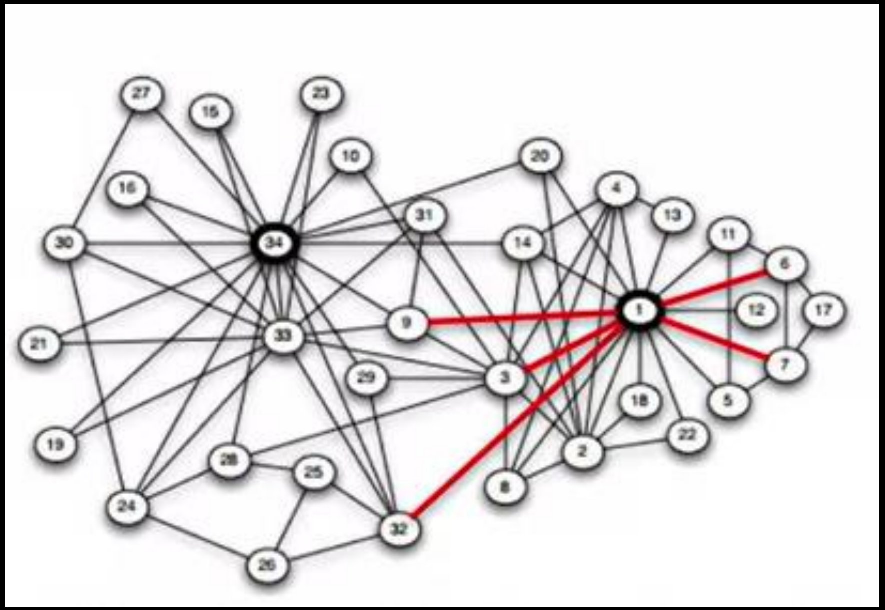
- Considering all edges

$$C_b(e) = \sum_{s,t\,\in V} \frac{\sigma_{s,t}(e)}{\sigma_{s,t}}$$

where

$\sigma_{s,t}$ is the number of shortest paths between s and t

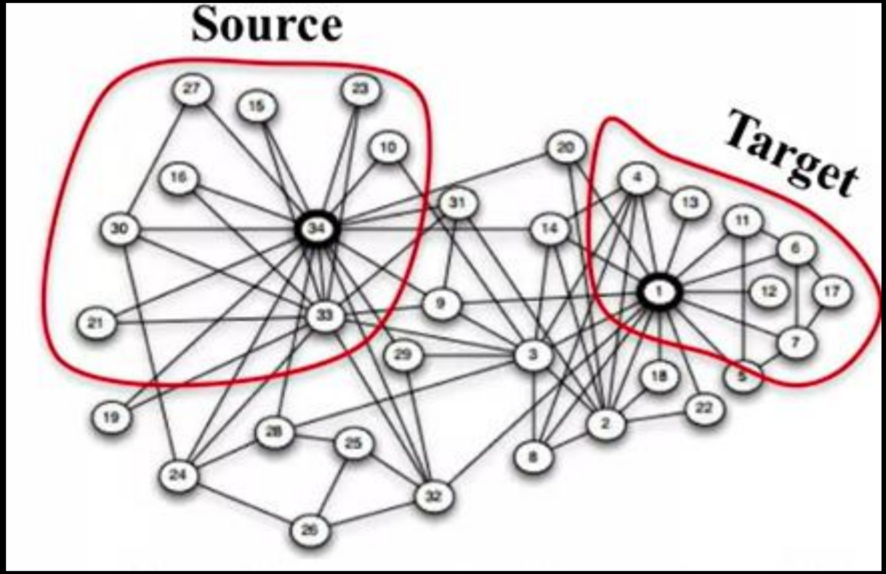
$\sigma_{s,t}(e)$ is the number of shortest paths between s and t passing through edge e.



Betweenness Centrality: Edge-Subset

- Considering between subsets S and T

$$C_b(e) = \sum_{s \in S, t \in T} \frac{\sigma_{s,t}(e)}{\sigma_{s,t}}$$



where

$\sigma_{s,t}$ is the number of shortest paths between s and t

$\sigma_{s,t}(e)$ is the number of shortest paths between s and t passing through edge e.

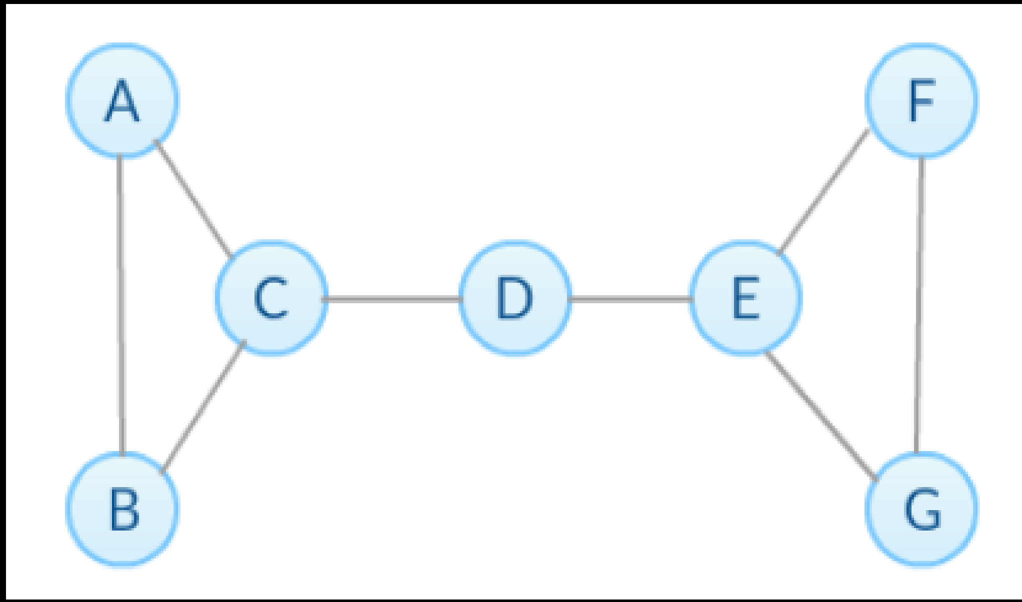
PRACTICE QUESTION

Q1. Which three nodes have highest betweenness centralities in the graph

1. Unnormalized
    1. Considering end points
    2. Not considering end points
2. Normalized not considering end points
3. Considering source nodes as {A,B,C,D} and target nodes as {E,F,G}

Q2. Which three edges have highest betweenness centralities in the graph considering all nodes?

Q3. Which three edges have highest betweenness centralities in the graph considering source nodes as {A,B,C,D} and target nodes as {E,F,G}



EIGENVECTOR CENTRALITY

EIGEN CENTRALITY

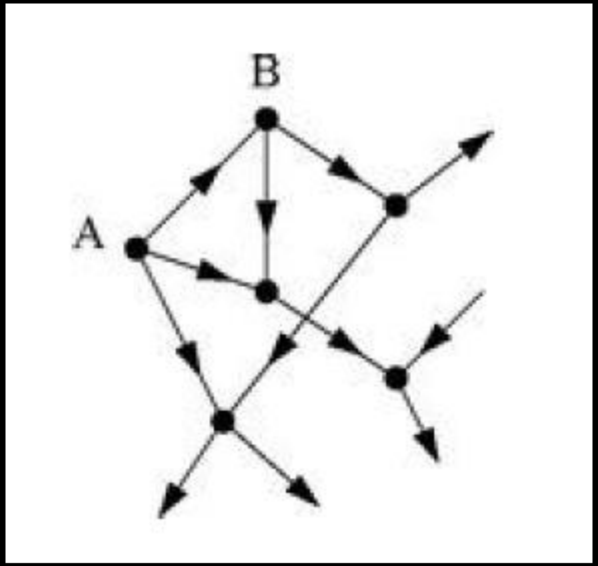
- A simple extension of degree centrality.
- **Assumption:** In many circumstances a vertex's importance in a network is increased by having connections to other vertices that are *themselves important*.
- **Principle:** Eigenvector centrality gives each vertex a score proportional to the sum of the scores of its neighbors.

$$C_e(v) = \sum_{(v,u) \in E} C_e(u)$$

- Iterate this till it gets converges.
- This information exactly can be captured by computing eigen values and eigen vectors of the Adjacency matrix.

Challenge

Nodes with 0 in-degree have 0 eigen value and it can’t further spread info to neighbors.



- Vertex A in this network has only outgoing edges and hence will have eigenvector centrality zero.
- Vertex B has outgoing edges and one ingoing edge, but the ingoing one originates at A, and hence vertex B will also have centrality zero.
- Solution: simply give each vertex a small amount of centrality “for free,” regardless of its position in the network or the centrality of its neighbors.

KATZ CENTRALITY

KATZ

- **Assumption:** Paths weaken when length increases
- **Principle:** Penalize the paths by length with a damping factor.

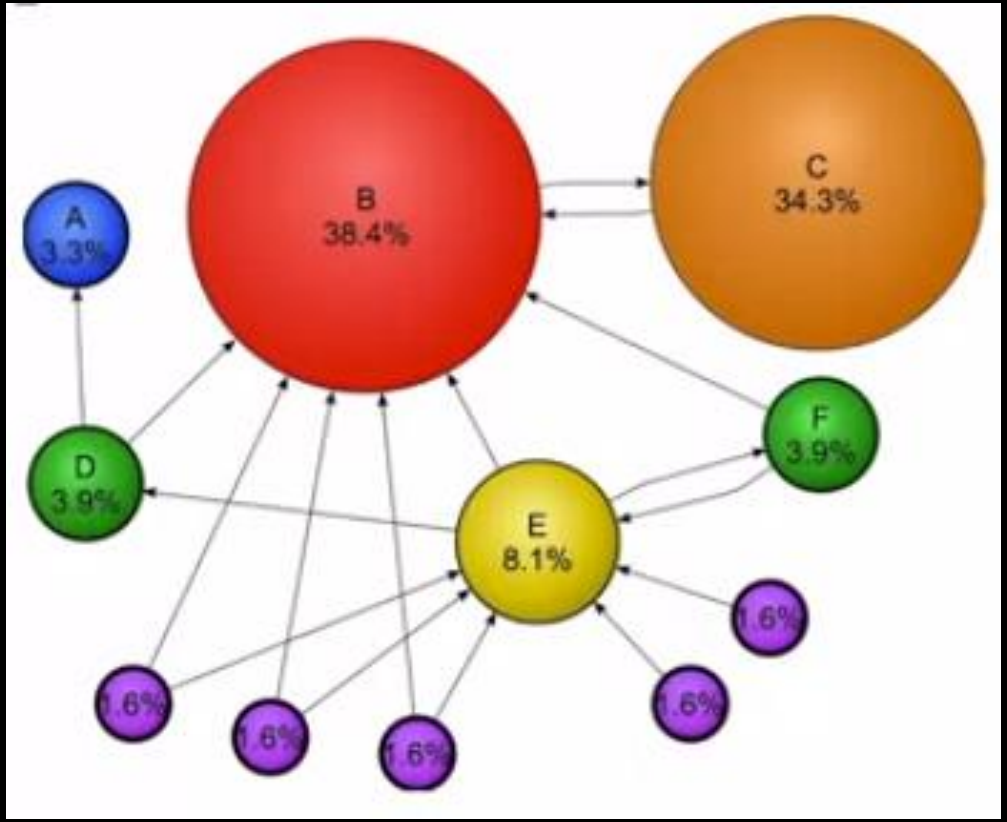
$$C_k(v) = \sum_{l=1}^{\infty} \alpha^l * |path^l(u,v)| + \beta$$

- where α is damping factor between 0 and 1 and
- β is the bias.

PAGE RANK CENTRALITY

PageRank

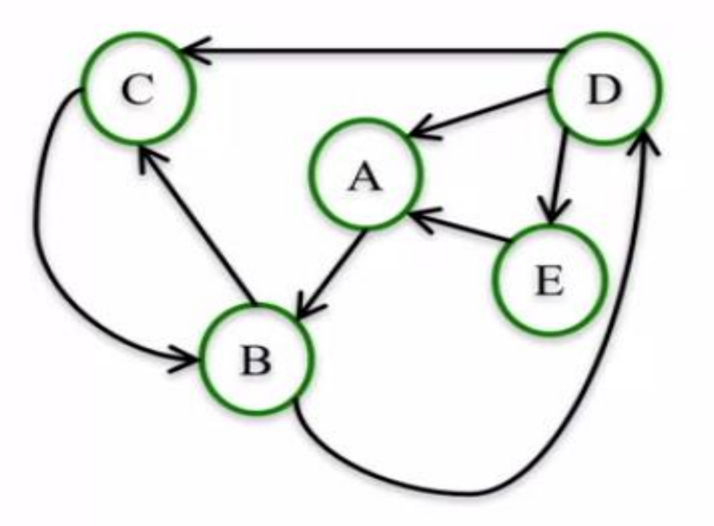
- Developed by Google founder Larry Page to measure importance of webpages in search engine
- Mainly focused on directed networks
- PageRank of a node depends on it’s neighbors.
- Def: PageRank of a node *v* at step *k* is the probability that a random walker reaches node *v* after *k* steps.



BASIC PAGERANK

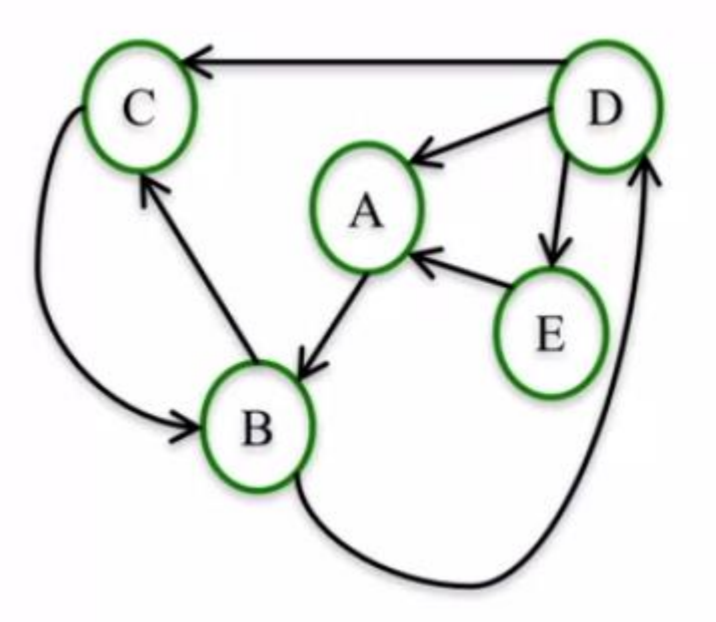
1. All nodes start with PageRank of $1/n$
2. Perform the *Basic PageRank Update Rule k* times:
   - **Basic PageRank Update Rule**: Each node gives an equal share of its current PageRank to all the nodes it links to.
   - The new PageRank of each node is the sum of all the PageRank it received from other nodes.

For most networks, PageRank values converge as k gets larger ($k \to \infty$)



Example

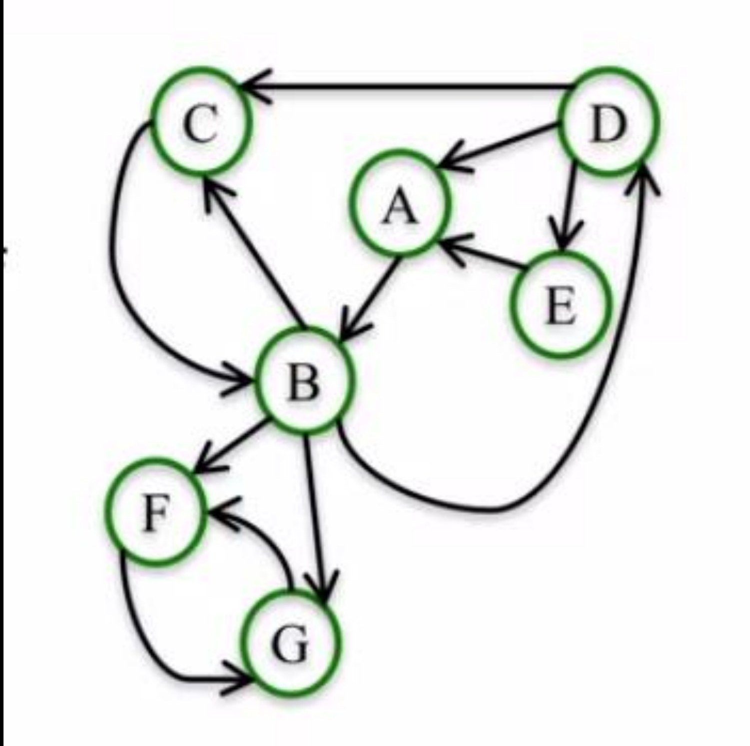
	A	B	C	D	E
Old					
New					



Iteration	A	B	C	D	E
1					
2					
3					

Special Condition

- After certain steps, the random walk structs at F or G.
- To fix this, introduce a parameter called damping factor, α.
  - With probability α, move to one of it's out neighbors uniformly at random
  - With probability 1- α, choose any node in the network and go to it. Note that, there are n-1 other nodes in the network, therefore probability of going to any random node in the network is $\frac{1}{n-1}$



$$C_p(v) = \left[\alpha * \sum_{(v,u)\in E} C_p(u)\right] + \left[(1-\alpha) * \frac{1}{n-1}\right]$$

- A value of α=0.85 is proved to give good convergence.

INSIGHTS

Centrality: Check Your Understanding

- generally different centrality metrics will be positively correlated
- when they are not, there is likely something interesting about the network
- suggest possible topologies and node positions to fit each square

	Low Degree	Low Closeness	Low Betweenness
High Degree		Embedded in cluster that is far from the rest of the network	Ego's connections are redundant - communication bypasses him/her
High Closeness	Key player tied to important/active players		Probably multiple paths in the network, ego is near many people, but so are many others
High Betweenness	Ego's few ties are crucial for network flow	Very rare cell. Would mean that ego monopolizes the ties from a small number of people to many others.	