LECTURE 35: NETWORKS 2

TEACHER: GIANNI A. DI CARO
Certain positions within the network give nodes more **importance / power**

- Directly affect/influence others
- Control the flow of information
- Avoid control of others

Influencers in social networks, vehicles for disease spreading, hubs in road networks, key infrastructures on the Internet, leaders in animal societies, ...
What characterizes an important node?

- **Network Centrality indices** provide answers to the question by defining a mapping that attributes a real-valued number to each node.
- These numbers can be used to determine a *ranking* among the nodes of a network.
**Central Centrality in Networks**

- **Centrality** encodes the relationship between *structure* (topology) and *importance/power* (flows of information and control) in interconnected systems.

  → Certain positions within the network give nodes more power or importance.

- **How do we measure importance?**
  
  - Who can *directly* affect/influence others? [direct information transfer]
    
    • Highest *degree* nodes are “in the thick of it”
  
  - Who controls *information flows*? [relaying information transfer]
    
    • Nodes that fall on shortest paths *between* others can disrupt the flow of information between them.
  
  - Who can *quickly* inform most others? [multicasting, few hops dissemination]
    
    • Nodes who are *close* to other nodes can quickly get/give information to them.
**Degree Centrality**

- **Degree Centrality** of node \( n \) is the number of other nodes \( n \) is connected to

  - A node with high degree has high *potential* communication activity

<table>
<thead>
<tr>
<th>node</th>
<th>In-degree</th>
<th>Out-degree</th>
<th>Total degree</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>1</td>
<td>3</td>
</tr>
</tbody>
</table>
MATHEMATICAL REPRESENTATION USING ADJACENCY MATRIX

Adjacency matrix A

\[
\begin{bmatrix}
0 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 \\
0 & 1 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 1 \\
0 & 1 & 0 & 0 & 0 \\
\end{bmatrix}
\]

Out-degree: row sum \( d_i^{out} = \sum_j A_{ij} \)

In-degree: column sum \( d_i^{in} = \sum_j A_{ji} \)

<table>
<thead>
<tr>
<th>node</th>
<th>In-degree</th>
<th>Out-degree</th>
<th>Total degree</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>1</td>
<td>3</td>
</tr>
</tbody>
</table>
**Betweenness Centrality** of node $n$ is the number of shortest paths (geodesics) connecting all pairs of other nodes that pass through $n$

- Node with highest betweenness can potentially control or distort communication between a large number of nodes

Paths from 1
- $1 \rightarrow 2$
- $1 \rightarrow 2 \rightarrow 3$
- $1 \rightarrow 2 \rightarrow 4$
- $1 \rightarrow 2 \rightarrow 4 \rightarrow 5$

Paths from 2
- $2 \rightarrow 3$
- $2 \rightarrow 4$
- $2 \rightarrow 4 \rightarrow 5$

Paths from 3
- $3 \rightarrow 2$
- $3 \rightarrow 4$
- $3 \rightarrow 5$

Paths from 4
- $4 \rightarrow 5 \rightarrow 2$
- $4 \rightarrow 5 \rightarrow 2 \rightarrow 3$
- $4 \rightarrow 5$

Paths from 5
- $5 \rightarrow 2$
- $5 \rightarrow 2 \rightarrow 3$
- $5 \rightarrow 2 \rightarrow 4$
CLOSENESS CENTRALITY

- Node that is *closer to all other nodes* can reach all other nodes in shortest amount of time (on average)
  - This node should best avoid being controlled by others!

- Closeness centrality is the reciprocal of the sum of geodesic distances from a node to all other nodes

\[
C(n) = \frac{1}{\sum d(m, n)}
\]

\[
C(n) = \frac{N}{\sum d(m, n)}
\]

In not strongly connected graphs, the distance between two nodes that are not connected is set to \(1/\infty=0\)
SELF-CONSISTENT MEASURES OF CENTRALITY

- **Katz (1953): Katz score**
  - “not only on how many others a person is connected to, but *who* he connects to”
  - One’s status is determined by the status of the people s/he is connected to

- **Bonacich (1972): Eigenvector centrality**
  - Node’s centrality is the sum of the centralities of its connections
  - *Relative centrality* can be computed as an average over the centralities of the directly connected nodes
  - We can use the degree centrality as a reference centrality for the $c_j$

\[
c_n = \frac{1}{\lambda} \sum_{j \in N(n)} c_j
\]
Bonacich (1972): Eigenvector centrality

- Node's centrality is the sum of the centralities of its connections

\[ c_i = \frac{1}{\lambda} \sum_{j \in \mathcal{N}(n)} c_j \]

Using the adjacency matrix \( A \) for degree centrality:

\[ c_i = \frac{1}{\lambda} \sum_{j \in \mathcal{G}} A_{ij} c_j \]

Where \( G \) is the entire network graph, \( \mathcal{N} \) is \( n \)'s neighborhood

\[ \lambda c_i = \sum_{j \in \mathcal{G}} A_{ij} c_j \]

This is an eigenvector equation!

\[ \lambda c = Ac \]

\( c \) is the eigenvector of \( A \) associated to the largest eigenvalue \( \lambda \)
(only \( c \geq 0 \) are of interest + Perron-Frobenius theorem)
Eigenvector centrality

\[ \lambda c_i = \sum_{j \in G} A_{ij} c_j \quad \lambda c = Ac \]

o The value of the \( n \)-th component of the eigenvector provides the value of the relative centrality of node \( n \)

o There’s a scale factor, such that only the relative values are meaningful (for ranking the nodes)

o Computational aspects: find the largest eigenvalue, compute the eigenvector (for very large matrices)
**Eigenvector Centrality**

- **Message passing** computation by estimate bootstrapping:

  - Start with an initial guess for the centrality of a node (e.g., number of neighbors), then iterate for each node $i$:
    - Get centrality values from neighbors and update $c_i$ according to
      \[ c_i \leftarrow \frac{1}{\lambda} \sum_{j \in \mathcal{N}} A_{ij} c_j \]
  - Over time, estimates become more and more accurate, converging to the correct estimates in the limit
  - Convergence can be assessed by any $L_1$, $L_2$, $L_\infty$ metric of choice for detecting that the vector of centralities undergo no significant changes
  - Any synchronous or asynchronous scheme can be adopted for selecting the nodes performing the update, examples from algorithms for *asynchronous value iteration* (*dynamic programming*) can be useful to devise computationally-effective schemes
MESSAGE PASSING BY PUSH-PULL GOSSIP

- The message passing algorithm algorithm can be implemented in a fully distributed and decentralized way, including running it on the network itself.

- Message passing computation or estimation of network-level node-local properties can be realized according to a general scheme based on push-pull gossip of information data, which is based on a model of epidemic spreading.

---

**Active thread**

Each node has two running threads

- **do forever**
  - wait(T time units)
  - \( p \leftarrow \text{GETRANDOMPEER}() \)
  - send \( s \) to \( p \)
  - \( s_p \leftarrow \text{receive}(p) \)
  - \( s \leftarrow \text{UPDATESTATE}(s, s_p) \)

**Passive thread**

- **do forever**
  - \( s_p \leftarrow \text{receive}(*) \)
  - send \( s \) to sender(\( s_p \))
  - \( s \leftarrow \text{UPDATESTATE}(s, s_p) \)

- **Active**: runs periodically, selects a neighbor \( p \) (peer), sends (push) its state value \( s \) to \( p \), receives (pull) \( p \)'s state value, use it to update its local state \( s \)

- **Passive**: listen to requests from neighbors, when contacted receives state value from requester, sends its state value, updates its local state

- \( s \) is the state value held by a node, that can be the centrality value, as well as any other state value of interest related to the network.
EIGENVECTOR CENTRALITY: POWER METHOD

- **Power method (Power iteration):** form of centralized numeric implementation of the message passing approach for computing centrality values, that iterates the following equation:

  \[ c_{t+1} = \frac{A c_t}{\|A c_t\|} \]

- This method is a general approach for the **numeric calculation of the largest eigenvalue** of a diagonalizable matrix \( A \) and, accordingly, of the **eigenvector** associated to the largest (dominant) eigenvalue

- **Assumptions (for convergence):** matrix \( A \) is diagonalizable, has a dominant eigenvalue, and the starting vector \( c_0 \) has a non-zero component in the direction of an eigenvector associated to the dominant eigenvalue

- It’s a simple and possibly slow to converge algorithm

- Appropriate for **large and sparse matrices**
**Alpha-Centrality (Bonacich, 1987)**

- **Alpha Centrality:** Similar to eigenvector centrality, but the degree to which a node centrality contributes to the centralities of other nodes depends on a parameter $\alpha$.

- **Mathematical interpretation:**
  
  - $c_i(\alpha)$ is the expected number of paths activated directly or indirectly by node $i$.

  $$c_i(\alpha) = \sum_j (1 + \alpha c_j(\alpha))A_{ij}$$

  $$c_i(\alpha) = \sum_j (A + \alpha A^2 + \alpha^2 A^3 + ...)$$
A CLOSER LOOK AT ALPHA-CENTRALITY

- Alpha-Centrality matrix

- 1\textsuperscript{st} term: number of paths of length 1 (edges) between \(i\) and \(j\)

- Contribution of this term to \(c_i(\alpha)\) is \(\Sigma_j A_{ij}\)

\[
\begin{bmatrix}
0 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 \\
0 & 1 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 1 \\
0 & 1 & 0 & 0 & 0 \\
\end{bmatrix}
\]

\[
c(\alpha) = A + \alpha A^2 + \alpha^2 A^3 + ...
\]
A CLOSER LOOK AT ALPHA-CENTRALITY

- Alpha-Centrality matrix
  \[ c(\alpha) = A + \alpha A^2 + \alpha^2 A^3 + \ldots \]

- 2nd term: number of paths of length 2 between \( i \) and \( j \)

\[
\begin{bmatrix}
0 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 \\
0 & 1 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 1 \\
0 & 1 & 0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
0 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 \\
0 & 1 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 1 \\
0 & 1 & 0 & 0 & 0
\end{bmatrix}
= 
\begin{bmatrix}
0 & 0 & 1 & 1 & 0 \\
0 & 1 & 0 & 1 & 2 \\
0 & 1 & 1 & 1 & 1 \\
0 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 0
\end{bmatrix}
\]
A CLOSER LOOK AT ALPHA-CENTRALITY

- Alpha-Centrality matrix

\[
c(\alpha) = A + \alpha A^2 + \alpha^2 A^3 + \ldots
\]

- 3rd term: number of paths of length 3 between \(i\) and \(j\)

\[
\begin{bmatrix}
0 & 0 & 1 & 1 & 0 \\
0 & 1 & 0 & 1 & 2 \\
0 & 1 & 1 & 1 & 1 \\
0 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 0
\end{bmatrix}
\times
\begin{bmatrix}
0 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 \\
0 & 1 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 1 \\
0 & 1 & 0 & 0 & 0
\end{bmatrix}
= 
\begin{bmatrix}
0 & 1 & 0 & 1 & 2 \\
0 & 2 & 1 & 1 & 1 \\
0 & 2 & 1 & 2 & 2 \\
0 & 0 & 1 & 1 & 0 \\
0 & 1 & 0 & 1 & 2
\end{bmatrix}
\]
A CLOSER LOOK AT ALPHA-CENTRALITY

- Alpha-Centrality matrix

\[ c(\alpha) = A + \alpha A^2 + \alpha^2 A^3 + ... = \sum_{k=0}^{\infty} \alpha^k A^{k+1} \]

- Number of paths of diverges as length of the path k grows

- To keep the infinite sum finite, \( \alpha < 1/\lambda_1 \), where \( \lambda_1 \) is the largest eigenvalue of \( A \) (also called radius of centrality)

- Interpretation: Node’s centrality is the sum of paths of any length connecting it to other nodes, exponentially attenuated by length of the path, so that longer paths contribute less than shorter paths

\[ c(\alpha) = A \sum_{k=0}^{\infty} \alpha^k A^k = (I - \alpha A)^{-1} A \]
Parameter $\alpha$ sets the length scale of communication or interactions.

- For $\alpha = 0$, only local interactions (with neighbors) are considered
  - Only **local** structure is important
  - Centrality is same as **degree centrality**
Parameter $\alpha$ sets the length scale of communication or interactions.

- As $\alpha$ grows, the length of interaction grows
  - **Global** structure becomes more important
  - Centrality depends on node’s position within a larger structure, e.g., a community
Parameter $\alpha$ sets the length scale of communication or interactions.

- As $\alpha \to 1/\lambda_1$, length of interactions becomes infinite
  - **Global** structure is important
  - Centrality is same as *eigenvector centrality*
Alpha-Centrality diverges for $\alpha > 1/\lambda_1$

Solution: Normalized Alpha-Centrality

\[
n(\alpha) = \frac{c(\alpha)}{N \sum_{i,j} c_{ij}(\alpha)}
\]

○ Holds for $0 \leq \alpha \leq 1$
MULTI-SCALE ANALYSIS WITH ALPHA-CENTRALITY

- Parameter $a$ allows for multi-scale analysis of networks
  - Differentiate between local and global structures
- Study how rankings change with $a$
  - Leaders: high influence on group members
    - Nodes with high centrality for small values of $a$
  - Bridges: mediate communication between groups
    - Nodes with low centrality for small values of $a$
    - But high centrality for large values of $a$
  - Peripherals: poorly connected to everyone
    - Nodes with low centrality for any value of $a$
KARATE CLUB NETWORK [ZACHARY, 1977]
Ranking Karate Club Members

Centrality scores of nodes vs. $\alpha$

![Graph showing centrality scores vs. $\alpha$](image)
Florentine families in 15th Century Italy
RANKING FLORENTINE FAMILIES
Network position confers advantages or disadvantages to a node, but how you measure it depends on what you mean by advantage:

- Ability to directly reach many nodes $\rightarrow$ degree centrality
- Ability to control information $\rightarrow$ betweenness centrality
- Ability to avoid control $\rightarrow$ closeness centrality

Self-consistent (bootstrapping) definitions of centrality:

- Node’s centrality depends on centrality of those it is connected to, directly or indirectly, but contribution of distant nodes is attenuated by how far they are:
  - Message passing and Power methods for computing centrality values
  - Attenuation parameter sets the length scale of interactions
  - Can probe structure at different scales by varying this parameter
Key insights

- Analyzes the structure of the web of hyperlinks to determine importance score of web pages
  - A web page is important if it is pointed to by other important pages

- An algorithm with deep mathematical roots
  - Random walks
  - Social network theory
Random Surfer

- Starts at arbitrary page
Random Surfer

- Starts at arbitrary page
- Bounces from page to page by following links randomly
PageRank and the Random Surfer

Random Surfer
- Starts at arbitrary page
- Bounces from page to page by following links randomly
- PageRank score of a web page is the relative number of time it is visited by the Random Surfer
PageRank is a solution to a random walk on a graph.

Adjacency matrix of the graph A:

\[
\begin{bmatrix}
A & B & C & D & E & F & G & H & I & L & M \\
A & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
B & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
C & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
D & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
E & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\
F & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
G & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
H & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
I & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
L & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
M & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
\end{bmatrix}
\]
PageRank is a solution to a random walk on a graph.
• PageRank is a solution to a random walk on a graph
  • $h_{ij}$ is probability to go from node $i$ to node $j$

$$h_{ij} = \frac{1}{d_i} \Rightarrow H = D^{-1}A$$
PageRank of page $j$ is defined recursively as
\[ \pi_j = \sum_i \pi_i h_{ij} \]
Or in matrix form $\pi = \pi H$

What contributes to PageRank score?
- Number of links page $j$ receives
  - $Cf$ B and D
- Number of outgoing links of linking pages
  - $Cf$ E’s effect on F and B’s effect on C
- PageRank scores of linking pages
  - $Cf$ E and B
Random Surfer gets trapped by dangling nodes! (no outlinks)

Solution: matrix $S$
- replace zero rows in $H$ with $u=[0.9,0.9, ..., 0.9]$
- From dangling node, surfer jumps to any other node
STILL PROBLEMS

- Random Surfer gets trapped in buckets
  - Reachable strongly connected component without outlinks
- Solution: teleportation matrix $E$
  - Matrix of $u$
Google matrix

\[ G = \alpha S + (1-\alpha) E \]

- Where \( \alpha \) is the damping factor

Interpretation of \( G \)

- With probability \( \alpha \), Random Surfer follows a hyperlink from a page (selected at random)
- With probability \( 1-\alpha \), Random Surfer jumps to any page (e.g., by entering a new URL in the browser)

PageRank scores are the solution of self-consistent equation

\[ \pi = \pi G \]

\[ = \alpha \pi S + (1-\alpha)u \]
SUMMARY

- Recursive (or self-consistent) nature of PageRank has roots in social network analysis metrics

- PageRank is fundamentally related to random walks on graphs
  - Lots of research to compute it efficiently
  - Huge economic and social impact!