MU5IN075
Network Analysis and Mining
3. Advanced concepts

Esteban Bautista-Ruiz, Lionel Tabourier

LIP6 – CNRS and Sorbonne Université

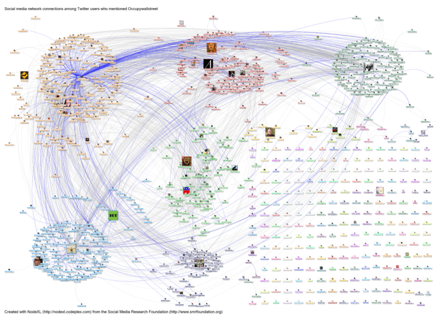
first_name.last_name@lip6.fr

September 28, 2021

Outline

Need for more elaborate representations

Consider cases such as the web, OSN like Twitter, emails, . . .
asymetric interactions



Need for an adapted representation → **directed graphs**

fr: *graphes orientés*

Basic definitions on directed graphs

directed graph (or digraph)

- V set of *vertices* (or *nodes*)
- $A \subseteq (V \times V)$ set of *arcs* ***fr:*** *arcs* and $(a, b) \neq (b, a)$

degree

In-degree *(degré entrant)*: $d^+(v)$ or $d_I(v)$
Out-degree *(degré sortant)*: $d^-(v)$ or $d_O(v)$

directed path

(chemin orienté)
from u to v : sequence of arcs $(u, v_1), (v_1, v_2), \ldots, (v_{k-1}, v)$

Connectedness of directed graphs

Directed path *(chemin orienté)* from u to v : sequence of arcs $(u, v_1), (v_1, v_2), \ldots, (v_{k-1}, v)$

Strongly connected component *(composante fortement connexe)*: maximal set of nodes such that $\exists$ a directed path from any node to any other of the set.

Weakly connected component *(composante faiblement connexe)*: maximal set of nodes such that $\exists$ a path between any pair of nodes in the graph where directed links are replaced by undirected links.

Connectedness of directed graphs

Supposing there is a giant weakly connected component
how can we subdivide it?

- subset of nodes which are all connected by directed paths: a **core**, largest SCC
- can lead to the core but cannot be reached from the core: **upstream**, in-component
- can be reached from the core but cannot lead to it: **downstream**, out-component
- other subsets: tendrils, tubes, ...

Connectedness of directed graphs

Supposing there is a giant weakly connected component
how can we subdivide it?

- subset of nodes which are all connected by directed paths: a **core**, largest SCC
- can lead to the core but cannot be reached from the core: **upstream**, in-component
- can be reached from the core but cannot lead to it: **downstream**, out-component
- other subsets: tendrils, tubes, ...

Connectedness of directed graphs

Supposing there is a giant weakly connected component
how can we subdivide it?

- subset of nodes which are all connected by directed paths: a **core**, largest SCC
- can lead to the core but cannot be reached from the core: **upstream**, in-component
- can be reached from the core but cannot lead to it: **downstream**, out-component
- other subsets: tendrils, tubes, ...

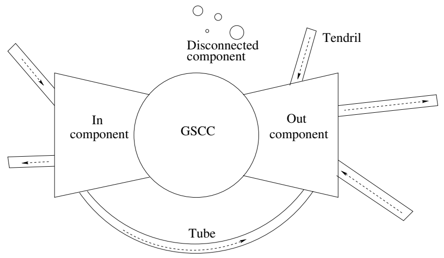
Connectedness of directed graphs

Supposing there is a giant weakly connected component
how can we subdivide it?

- subset of nodes which are all connected by directed paths: a **core**, largest SCC
- can lead to the core but cannot be reached from the core: **upstream**, in-component
- can be reached from the core but cannot lead to it: **downstream**, out-component
- other subsets: tendrils, tubes, . . .

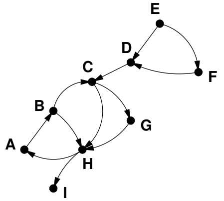
Connectedness of directed graphs

- subset of nodes which are all connected by directed paths: a **core**, largest SCC
- can lead to the core but cannot be reached from the core: **upstream**, in-component
- can be reached from the core but cannot lead to it: **downstream**, out-component
- other subsets: tendrils, tubes, . . .



Practicing with directed graphs

Consider the following graph G_d



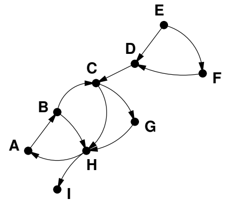
Graph representation

Represent G_d in the following formats:

- list of arcs
- adjacency matrix
- adjacency lists

Practicing with directed graphs

Consider the following graph G_d

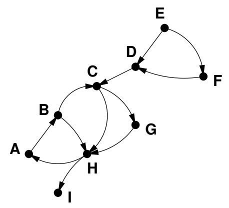


Degree

Find its in- and out-degree distributions.

Practicing with directed graphs

Consider the following graph G_d



Connectedness

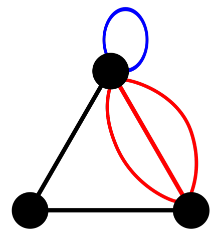
Find its largest strongly connected component.

Multigraphs, weighted graphs

Multigraphs

Can have several edges between nodes and possibly loops

- collaboration networks → undirected, multi
- email exchange networks → directed, multi

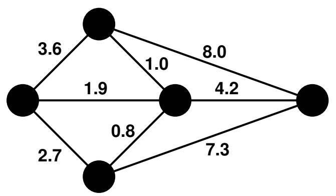


Multigraphs, weighted graphs

Weighted graphs

Generalization to weights which are real numbers (not integers)

- trade networks → directed, weighted
- phonecall networks → (un)directed, weighted

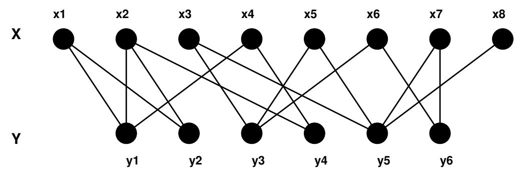


Bipartite networks

Bipartite graph

Two distinct types of nodes U and V, links between U and V

- users watching videos, clients purchasing items, ...
- in general, user/item selection networks

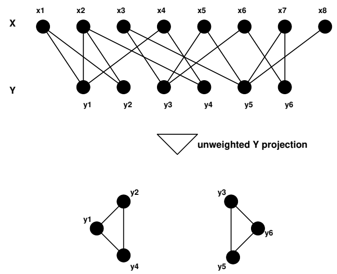


Bipartite networks

Projection of a bipartite graph

If bipartite data not available, if metrics not adapted. . .
$\rightarrow$ projection of bipartite data

Projection on U: if u_1 and u_2 connected to v in bipartite
$\Rightarrow u_1$ and u_2 are connected in the projection
A projection can be unweighted or weighted



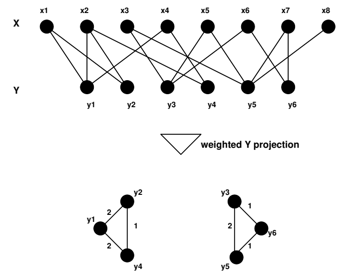
Different types of graphs
Centrality notions

Bipartite networks

Projection of a bipartite graph

If bipartite data not available, if metrics not adapted. . .
$\rightarrow$ projection of bipartite data

Projection on U: if u_1 and u_2 connected to v in bipartite
$\Rightarrow u_1$ and u_2 are connected in the projection
A projection can be unweighted or weighted



Different types of graphs
Centrality notions

Underlying bipartite nature of data

Data often have an underlying bipartite nature
$\rightarrow$ only the projection data is available

Example of contact networks:

- interaction occur during events or inside groups
  ex: disease spreading networks
- but we only get the projection information
  ex: data collection = potential contamination link
- importance of recovering the information
  ex: contact tracing, modeling

Understanding the structure of a network may involve going back to the structure of the underlying bipartite network
($\rightarrow$ course on models)

Different types of graphs
Centrality notions

Underlying bipartite nature of data

Data often have an underlying bipartite nature
$\rightarrow$ only the projection data is available

Example of contact networks:

- interaction occur during events or inside groups
  ex: disease spreading networks
- but we only get the projection information
  ex: data collection = potential contamination link
- importance of recovering the information
  ex: contact tracing, modeling

Understanding the structure of a network may involve going back to the structure of the underlying bipartite network
($\rightarrow$ course on models)

Underlying bipartite nature of data

Data often have an underlying bipartite nature
→ only the projection data is available

Example of contact networks:

- interaction occur during events or inside groups
  ex: disease spreading networks
- but we only get the projection information
  ex: data collection = potential contamination link
- importance of recovering the information
  ex: contact tracing, modeling

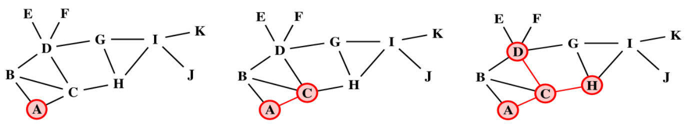
Understanding the structure of a network may involve going back to the structure of the underlying bipartite network
(→ course on models)

Outline

1. Different types of graphs
2. Centrality notions

What is a centrality?

context 1: spreading phenomena

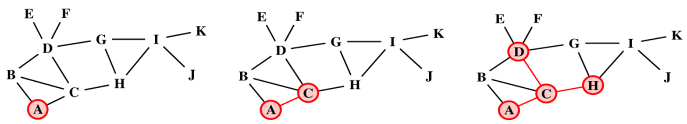


- goal: spread an information in a network fast (*ex: ads*)
- constraint: you can only select a few source nodes
- which selection strategy?

→ select the best "spreaders" first

What is a centrality?

context 1: spreading phenomena



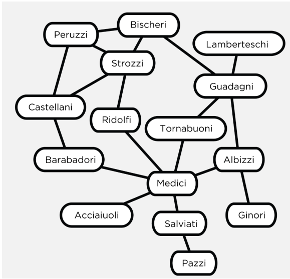
- goal: spread an information in a network fast (*ex: ads*)
- constraint: you can only select a few source nodes
- which selection strategy?

→ select the best "spreaders" first

What is a centrality?

context 2: in human interaction networks, centrality relates to organization of a group, influence and leadership

ex: rise of the Medici in 1400-1434 Padgett and Ansell - *1993*



credits image: V.Gauthier

$\rightarrow$ analyzed through centrality measures Borgatti - *2005*

`https://graal.hypotheses.org/758` in french

Different types of graphs
Centrality notions

What is a centrality?

Centrality measures the **relative importance** of nodes in G

As there is no universal meaning for importance, there are many ways to measure centrality.

Different types of graphs
Centrality notions

Some ideas for an importance measure?

Degree

- simple to measure (in $\mathcal{O}(M)$)
- but not necessarily meaningful

Other idea

An important node should be close to every other nodes
$\rightarrow$ closeness centrality Bavelas - *1950*

$$C_C(x) = \frac{1}{\sum_{y \neq x} d(x, y)}$$

centralité de proximité

Different types of graphs
Centrality notions

Some ideas for an importance measure?

Degree

- simple to measure (in $\mathcal{O}(M)$)
- but not necessarily meaningful

Other idea

An important node should be close to every other nodes
$\rightarrow$ closeness centrality Bavelas - *1950*

$$C_C(x) = \frac{1}{\sum_{y \neq x} d(x, y)}$$

centralité de proximité

Some ideas for an importance measure?

Degree

- simple to measure (in $\mathcal{O}(M)$)
- but not necessarily meaningful

Other idea

An important node should be close to every other nodes
$\rightarrow$ closeness centrality Bavelas - *1950*

$$C_C(x) = \frac{1}{\sum_{y \neq x} d(x,y)}$$

centralité de proximité

Measuring closeness centrality

For any node x, compute its distance to any other node in the connected component, deduce $C_C(x)$

Measuring closeness centrality

For any node x, compute its distance to any other node in the connected component, deduce $C_C(x)$

Algorithm 1: Modified BFS for distance computation from s

```
F ← CreateFIFO()
F.add((s, 0))
Mark(s)
while F not empty do
    (u, d_u) ← F.pop()
    output u, d_u
    for v neighbor of u in G do
        if Unmarked(v) then
            d_v ← d_u + 1
            F.add((v, d_v))
            Mark(v)
```

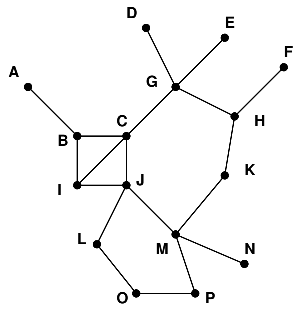

Measuring closeness centrality

For any node x, compute its distance to any other node in the connected component, deduce $C_C(x)$

Complexity

- for one node $\mathcal{O}(M)$
- $\Rightarrow$ for all nodes $\mathcal{O}(NM)$

Exercise



Compute the closeness centrality of J, of F

Limitations

Closeness is only one way to define importance though . . .

Some limitations:

- if several connected components
- do not give specific importance to bridge nodes

Betweenness centrality

Introduced by Freeman (circa 1977) for social networks
$\rightarrow$ evaluate the capacity to control information flows

Betweenness centrality definition

centralité d'intermédiarité

$$C_B(x) = \sum_{i \neq x \neq j} \frac{\sigma_{ij}(x)}{\sigma_{ij}}$$

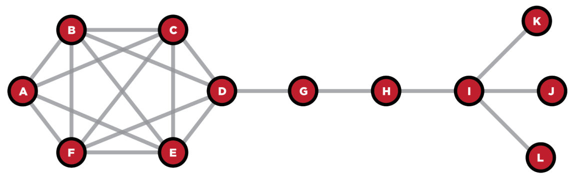
where

- σ_{ij}: number of shortest paths from i to j
- $\sigma_{ij}(x)$: number of shortest paths from i to j going through x

Quiz

Find a node which have:

- high betweenness but low degree
- high degree but low betweenness

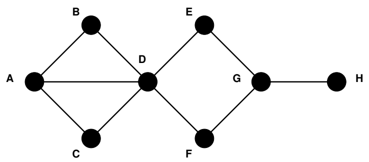


credits image: V.Gauthier

Algorithm to measure betweenness centrality

Listing shortest path expensive in memory
$\Rightarrow$ how not to list explicitly shortest paths?

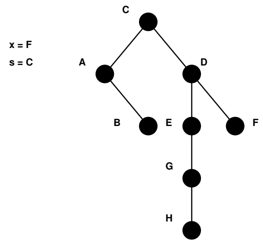
Example:



Algorithm to measure betweenness centrality

To compute the contribution of pair (s,t) to $C_B(x)$, i.e. $\frac{\sigma_{st}(x)}{\sigma_{st}}$

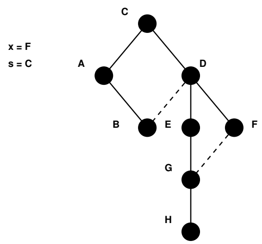
1. Modify BFS to detect all shortest paths from s to any other node $\Rightarrow$ graph G_s
2. Enumerate number of paths $nb_{G_s}(s,v)$ going through any v
   = sum number of paths going through predecessors of v
3. By definition $\sigma_{st} = nb_{G_s}(s,t)$, what about $\sigma_{st}(x)$?
   in G_s, the number of shortest paths from t to v is $nb_{G_s}(t,v)$
   then $\sigma_{st}(x) = nb_{G_s}(s,x) \times nb_{G_s}(t,x)$



Algorithm to measure betweenness centrality

To compute the contribution of pair (s,t) to $C_B(x)$, i.e. $\frac{\sigma_{st}(x)}{\sigma_{st}}$

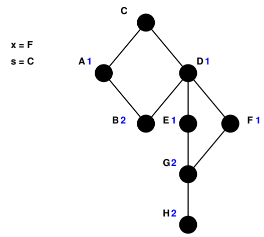
1. Modify BFS to detect all shortest paths from s to any other node $\Rightarrow$ graph G_s
2. Enumerate number of paths $nb_{G_s}(s,v)$ going through any v
   = sum number of paths going through predecessors of v
3. By definition $\sigma_{st} = nb_{G_s}(s,t)$, what about $\sigma_{st}(x)$?
   in G_s, the number of shortest paths from t to v is $nb_{G_s}(t,v)$
   then $\sigma_{st}(x) = nb_{G_s}(s,x) \times nb_{G_s}(t,x)$



Algorithm to measure betweenness centrality

To compute the contribution of pair (s,t) to $C_B(x)$, i.e. $\frac{\sigma_{st}(x)}{\sigma_{st}}$

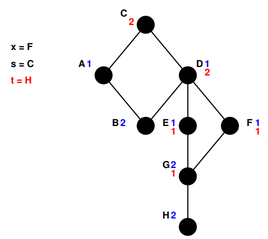
1. Modify BFS to detect all shortest paths from s to any other node $\Rightarrow$ graph G_s
2. Enumerate number of paths $nb_{G_s}(s,v)$ going through any v
   = sum number of paths going through predecessors of v
3. By definition $\sigma_{st} = nb_{G_s}(s,t)$, what about $\sigma_{st}(x)$?
   in G_s, the number of shortest paths from t to v is $nb_{G_s}(t,v)$
   then $\sigma_{st}(x) = nb_{G_s}(s,x) \times nb_{G_s}(t,x)$



Algorithm to measure betweenness centrality

To compute the contribution of pair (s, t) **to** $C_B(x)$**, i.e.** $\frac{\sigma_{st}(x)}{\sigma_{st}}$

1. Modify BFS to detect all shortest paths from s to any other node $\Rightarrow$ graph G_s
2. Enumerate number of paths $nb_{G_s}(s, v)$ going through any v
   = sum number of paths going through predecessors of v
3. By definition $\sigma_{st} = nb_{G_s}(s, t)$, what about $\sigma_{st}(x)$?
   in G_s, the number of shortest paths from t to v is $nb_{G_s}(t, v)$
   then $\sigma_{st}(x) = nb_{G_s}(s, x) \times nb_{G_s}(t, x)$



Different types of graphs
Centrality notions

Algorithm to measure betweenness centrality

To compute the contribution of pair (s, t) **to** $C_B(x)$**, i.e.** $\frac{\sigma_{st}(x)}{\sigma_{st}}$

1. Modify BFS to detect all shortest paths from s to any other node $\Rightarrow$ graph G_s
2. Enumerate number of paths $nb_{G_s}(s, v)$ going through any v
   = sum number of paths going through predecessors of v
3. By definition $\sigma_{st} = nb_{G_s}(s, t)$, what about $\sigma_{st}(x)$?
   in G_s, the number of shortest paths from t to v is $nb_{G_s}(t, v)$
   then $\sigma_{st}(x) = nb_{G_s}(s, x) \times nb_{G_s}(t, x)$

$\rightarrow$ do this for all $s \neq x$ and, when s is fixed: all $t \neq x$ and $t \neq s$
complexity in $\mathcal{O}(N.(M + NM)) = \mathcal{O}(N^2M)$

Different types of graphs
Centrality notions

Algorithm to measure betweenness centrality

To compute the contribution of pair (s, t) **to** $C_B(x)$**, i.e.** $\frac{\sigma_{st}(x)}{\sigma_{st}}$

1. Modify BFS to detect all shortest paths from s to any other node $\Rightarrow$ graph G_s
2. Enumerate number of paths $nb_{G_s}(s, v)$ going through any v
   = sum number of paths going through predecessors of v
3. By definition $\sigma_{st} = nb_{G_s}(s, t)$, what about $\sigma_{st}(x)$?
   in G_s, the number of shortest paths from t to v is $nb_{G_s}(t, v)$
   then $\sigma_{st}(x) = nb_{G_s}(s, x) \times nb_{G_s}(t, x)$

$\rightarrow$ do this for all $s \neq x$ and, when s is fixed: all $t \neq x$ and $t \neq s$
complexity in $\mathcal{O}(N.(M + NM)) = \mathcal{O}(N^2M)$

Different types of graphs
Centrality notions

Another example of centrality measures uses

Hypothesis of the strength of weak tie Granovetter - *1973*

A “weak tie” is a link in a social network which represents a relation which is not frequently maintained

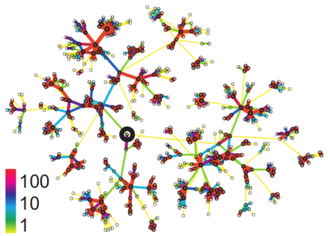
It is argued that weak ties play an essential role as they **ensure connections between groups**

Another example of centrality measures uses

Experimental validation on phonecall network Onnela et al. - *2007*

- strength of a relationship = cumulative duration of calls
- how to measure the fact that a link is between groups?

weight (color) = cumulative duration of calls:

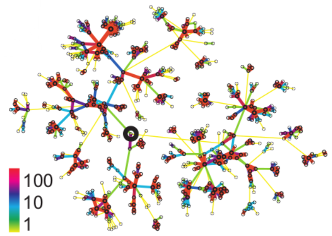


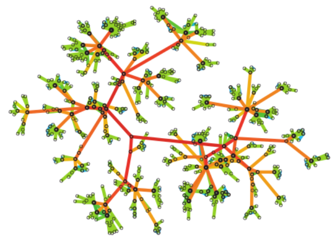
Another example of centrality measures uses

Experimental validation on phonecall network Onnela et al. - *2007*

Link betweenness definition similar to node betweenness

weight (color) = link betweenness:





Another example of centrality measures uses

Experimental validation on phonecall network Onnela et al. - *2007*

Quiz: what would you plot to check the correlation ?

Centrality measurements

- Degree
  - often not very relevant
  - low computational cost
- Closeness centrality
  - take into account distance from the node to others
  - quadratic computational cost
- Betweenness centrality
  - take into account relaying position of the node
  - computationally expensive (but can be improved. . .)
- Harmonic centrality
  - an alternative to closeness centrality (highly correlated)
  - same computational cost as closeness
- Katz centrality
  - take into account number of paths from the node to others
  - computationally expensive
- and others. . . Boldi and Vigna - *2013*

What about directed networks?

WWW: billion of web pages → **which are the most relevant?**

- need for fast computation
- direction is important

⇒ in-degree? no

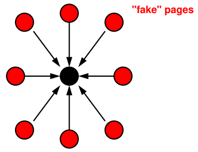
→ see course on search engines

What about directed networks?

WWW: billion of web pages → **which are the most relevant?**

- need for fast computation
- direction is important

⇒ in-degree? no



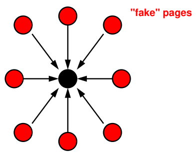
→ see course on search engines

What about directed networks?

WWW: billion of web pages → **which are the most relevant?**

- need for fast computation
- direction is important

⇒ in-degree? no



→ see course on search engines