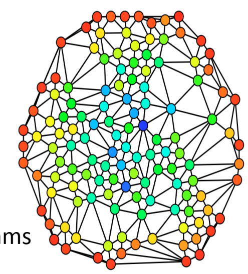
CS544: Graph Algorithms, Social Networks & NLP applications

March 22, 2011

Zornitsa Kozareva
USC/ISI
Marina del Rey, CA
kozareva@isi.edu
www.isi.edu/~kozareva

Graph Theory

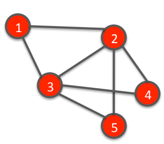
- General introduction (terminology)
- Directed Graphs
- Undirected Graphs
- Refresh some algorithms



1

What is a Graph?

- A *graph* G=(***V***,***E***) is composed of:
  - ***V***: set of ***vertices***
  - ***E***: set of ***edges*** connecting the vertices
- An ***edge*** e=(u,v) is a pair of ***vertices***



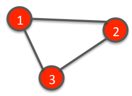
V = {1,2,3,4,5}

E = {(1,2);(1,3);(2,3);(2,4);(2,5);(3,4);(3,5)}

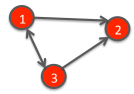
2

Undirected and Directed Graphs

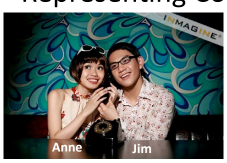
- An ***undirected graph*** is one in which the pair of vertices in an edge is unordered
  - $(v_0,v_1) = (v_1,v_0)$



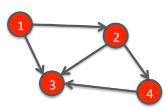
- A ***directed graph*** is one in which each edge is a directed pair of vertices
  - $(v_0,v_1) \;!\!= (v_1,v_0)$



Representing Conversations as Graph



Anne:	Jim, tell the Murrays they're invited
Jim:	Don, you and your dad should come for dinner!
Jim:	Mr. Murray, you should both come for dinner
Anne:	Don, did Jim tell you about the dinner? You must come.
Don:	Dad, we are invited for dinner tonight
Sam:	Anne, we're going, it's settled!

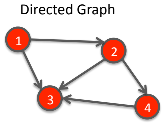


vertex (node)

edge (link)

Data Representation - Directed Graph

Directed Graph



vertex (node)

edge (link)

Edge list

Vertex	Vertex
1	2
1	3
2	3
2	4
3	4

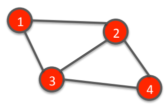
Adjacency matrix

Vertex	1	2	3	4
1	-	1	1	0
2	0	-	1	1
3	0	0	-	0
4	0	0	1	-

Directed graph captures who speaks to whom in the conversation

Data Representation - Undirected Graph

Undirected Graph



Edge list

Vertex	Vertex
1	2
1	3
2	3
2	4
3	4

Adjacency matrix (symmetric)

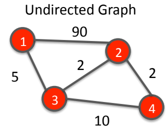
Vertex	1	2	3	4
1	-	1	1	0
2	1	-	1	1
3	1	1	-	1
4	0	1	1	-

Undirected graph captures who knows who in the conversation

6

Adding weights to edges

Undirected Graph



Edge list

Vertex	Vertex	Weight
1	2	90
1	3	5
2	3	2
2	4	2
3	4	10

Adjacency matrix has the weights instead 0 and 1

Vertex	1	2	3	4
1	-	90	5	0
2	90	-	2	2
3	5	2	-	10
4	0	2	10	-

Weights can represent the frequency of interaction

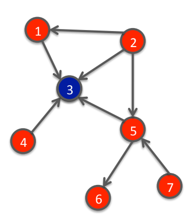
7

Degree Centrality

- The activity of a node can be captured through degrees
- The degree of a node corresponds to the number of direct connections it has
- <u>Degree measures:</u>

  − $inDegree(u) = \sum_{\forall (u,v) \in E} (v,u)$

  is the sum of all incoming edges to *u*

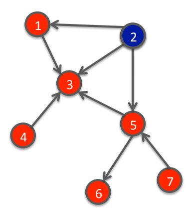


8

Degree Centrality

- The activity of a node can be captured through degrees
- The degree of a node corresponds to the number of direct connections it has
- <u>Degree measures:</u>
  - $outDegree(u) = \sum_{\forall (u,v) \in E} (v,u)$

  is the sum of all outgoing edges to *u*

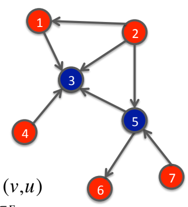


9

Degree Centrality

- The activity of a node can be captured through degrees
- The degree of a node corresponds to the number of direct connections it has
- <u>Degree measures:</u>
  - $totalDegree(u) = \sum_{\forall (u,v) \in E} (u,v) + \sum_{\forall (u,v) \in E} (v,u)$

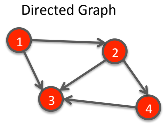
  is the sum of all outgoing and incoming edges to *u*



10

Calculate Centrality with Adjacency Matrix

Directed Graph



Adjacency matrix

Vertex	1	2	3	4
1	-	1	1	0
2	0	-	1	1
3	0	0	-	0
4	0	0	1	-

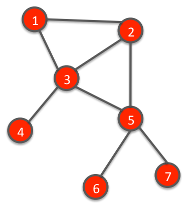
- The row sum is the *outDegree* of a node
- The column sum is the *inDegree* of a node

11

Degree Centrality

- How much are the *inDegree*, *outDegree* and *totalDegree* of a node in the undirected graph?

Answer: They are identical.



12

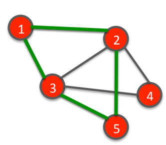
Why using Centrality ?

- Centrality measure captures the connectedness of a node, hence we can measure influence and/or popularity
- Useful in assessing which nodes are central with respect to spreading information and influencing others in their immediate neighborhood
- Analyze your own networks

13

Paths and short path

- A ***path*** between two nodes is any sequence of vertices $v_1, v_2, \ldots v_k$ that connect two nodes
- The ***shortest path*** between two nodes is the path that connects the two nodes with the shortest number of edges



- How much is the shortest path between nodes 1 and 5?

Length 2: {1,3,5} and {1,2,5}

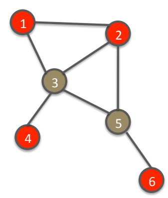
- What are the longer paths between the two nodes?

{1,2,3,5}, {1,2,3,5}, {1,2,4,3,5}, {1,3,2,4,5}

14

Betweenness Centrality

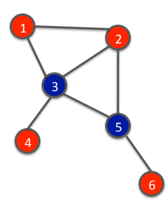
- The ***betweenness*** of node v is $BE(v) = \sum_{\substack{s \neq v \neq t \in V \\ s \neq t}} \frac{\sigma_{st}(v)}{\sigma_{st}}$ the number of shortest paths that pass through a node divided by all shortest paths in the network
- Reflects which nodes are more likely to be in communication paths between other nodes
- What happens if nodes 3 and 5 are removed from the network?



15

Closeness Centrality

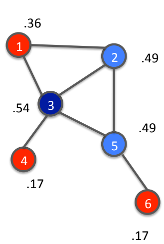
- The ***closeness*** of a node v_i is
$$CL(v_i) = \frac{n-1}{\sum_{j=1}^{n} d(v_i, v_j)}$$
where *n* is the total number of nodes in the graph, $d(v_i,v_j)$ is the shortest path of v_i to all other nodes in the network
(how many hops on average are necessary to reach every other node)
- Measures the ***reach***, i.e. how long will it take to reach other nodes from a given starting node
- Useful when information dissemination is main concern



16

Eigenvector Centrality

- The ***eigenvector centrality*** is the sum of the eigenvector centralities of all nodes directly connected to it
- A node with high eigenvector centrality is connected to other nodes with high eigenvector centralities
- Useful to determine who is connected to the most connected nodes



17

Interpretations in Social Networks

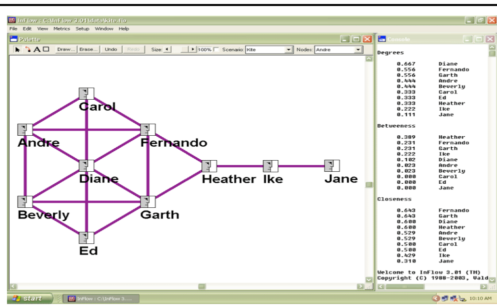
- **Degree** How many people can this person reach directly?
- **Betweenness** How likely is this person to be the most direct route between two people in the network?
- **Closeness** How fast can this person reach everyone in the network?
- **Eigenvector** How well is this person connected to other well-connected people?

18

General Interpretations …

- **Degree** How many people has this person collaborated with?
- **Betweenness** Who is the spy through whom most of the confidential information is likely to flow?
- **Closeness** How fast will a disease spread from a person to the rest of the network?
- **Eigenvector** Who is the author that is most cited by other well-cited authors?

19



- Which node has the highest degree?
- Which node is the most central to the network?

20

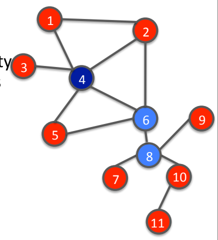
Key Player Problem

- The key player is calculated as

$$KPP(v) = \frac{\sum_{u \in V} \frac{1}{d(u,v)}}{|V| - 1}$$

high values indicate strong connectivity and proximity to the rest of the nodes

- Observations:
  - node 4 is the most central node
  - nodes 6 and 8 reach more nodes
  - if nodes 6 and 8 are removed, the network will become disrupted.
  - nodes 6 and 8 together are more 'key' to this network than node 4



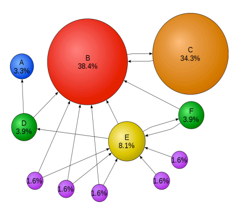
21

Page Rank *(Brin & Page 1998)*

- The page rank of a node v is

$$PR(v) = \frac{(1-\alpha)}{|V|} + \alpha \sum_{u,v \in E} \frac{PR(u)}{\mathrm{outD(u)}}$$

- Imagine a web surfer doing a simple random walk on the entire web for an infinite number of steps.
- Occasionally, the surfer will get bored and instead of following a link pointing outward from the current page will jump to another random page.



- At some point, the percentage of time spent at each page will converge to a fixed value.

22

NLP APPLICATIONS, SEMANTIC CLASS LEARNING

23

FLASHBACK

How are Max Planck, Angela Merkel and Dalai Lama related?

All have doctoral degrees from German universities

24

FLASHBACK

Semantic Class Learning: Objectives

- Given a class and an instance, learn automatically with minimum supervision new instances of that class
- Examples:
  - *class_name*: Nobel prize winners
  - *instances*: Albert Einstein, Max Plank …
  - *class_name*: US states
  - *instances*: Georgia, Alabama, California …

25

FLASHBACK

Bootstrapping

- Start with a pattern, *class_name* and <*seed*>
- Feed the newly learned terms on <*seed*> position
- Conduct a breadth-first search

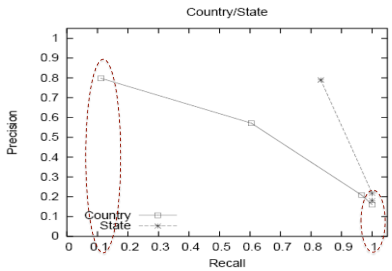
states such as *Texas* *Alabama* *Mississippi* *Arkansas* and *

Florida
Texas
California
Mississippi
…
Arkansas
Tennessee
Nevada

26

FLASHBACK

Performance of Bootstrapping



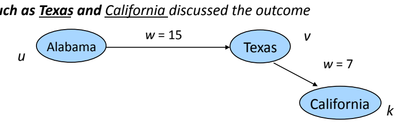
Problem: search needs guidance
Solution: rank the learned instances

27

Hyponym Pattern Linkage Grap

- HPLG=(V,E) where vertex $v \in V$ is an instance, and $e \in E$ is an edge between two instances

*certain **states, such as Alabama and** Texas, should forbid prayers that are led*
***states such as Texas and** California discussed the outcome*

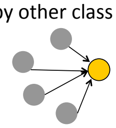


- the weight *w* of an edge is the frequency with which *u* generated *v*

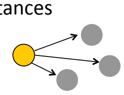
28

Properties of Graph Measures

- Observing two characteristics:
  – *Popularity*, the ability of an instance to be discovered by other class instances



  – *Productivity*, the ability of an instance to discover other class instances



29

Ranking Functions

- Employed graph-based measures
  - inDegree
  - outDegree
  - totalDegree
  - Betweenness
  - Key player
- Use them to rank the learned instance elements

30

Learning US states

number of learned instances →

	States						
	#N	inDegree	outDegree	totalDegree	Betweenness	KeyPlayer	Bootstrap.
	25	1.0	**1.0**	**1.0**	.88	1.0	.45
	50	.98	**1.0**	**1.0**	.86	.98	.10
	64	.77	.78	.78	.77	.78	--

- HPLGs perform better than bootstrapping
- outDegree and totalDegree discover all state instances
- if there are only 50 US states, why does the algorithm keep on learning

31

The Troublesome Fourteen

- Instances after the learned 50 US states:

  Russia, Ukraine, Uzbekistan, Azerbaijan, Moldava, Tajikistan, Armenia, Chicago, Boston, Atlanta, Detroit, Philadelphia, Tampa, Moldavia

“authoritarian **former Soviet states** such as **Georgia** and **Ukraine**”

“Findlay has 20 restaurants in states such as **Florida** and **Chicago**”

32

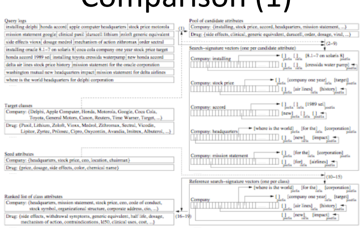
Learning Country Names

Countries		
#N	KeyPlayer	outDegree
10	.90	**1.0**
25	.88	**1.0**
50	.80	**1.0**
75	.69	.93
100	.68	.84
116	.65	.80

Error Analysis

- Type 1: incorrect proper name extraction
  "states such as Georgia and English speaking countries"
- Type 2: instances that formerly belonged to the semantic class
  "Serbia-Montenegro", *"Czechoslovakia"*
- Type 3: spelling variants
  "Kyrgystan" vs *"Kyrgyzhstan"*
- Type 4: sentences with wrong factual assertions
  "industry in countries such as France and North America"
- Type 5: broken expressions
  "issue has been tough for states such as Texas and New"

Comparison (1)



- Contextual vectors from query logs (Pasca,07)*

Learned Country Names	Pasca 07 (precision)	outDegree (precision)
100	95%	100%
150	82%	100%

***Organizing and Searching the World Wide Web of Facts - Step Two: Harnessing the Wisdom of the Crowds (Pasca,07)**

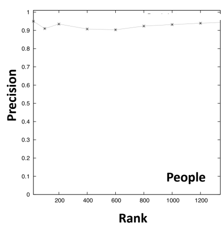
Comparison (2)

- KnowItAll system, details in Lecture #6
  - uses singly-anchored patterns
    "country such as *"
  - ranks with mutual information

Learned Country Names	KnowItAll 1	KnowItAll 2	outDegree
Precision	79%	97%	100%
Recall	89%	58%	77%

36

Evaluation against WordNet



	# harvested	PrWN	PrHUM	NotInWN
People Names	1344	.23	.95	986

37

Lessons Learned so far …

- Graph algorithms can be employed to guide semantic class harvesting systems
- outDegree outperforms complex graph ranking algorithms
- Achieves higher recall and accuracy compared to existing knowledge harvesting algorithms
- Learns information missing from WordNet

38