Link Analysis: Web Structure and Search
Web Science (VU) (707.000)

Elisabeth Lex
KTI, TU Graz

May 2, 2016
Repetition

- Signed networks (positive, negative, absent)
- Structural balance
- Weak structural balance, Status Theory, Trust/Distrust
Outline

1. Information Networks
2. Paths and Strong Connectivity
3. The Bow-Tie Structure of the Web
4. Web Search
5. Link Analysis and Ranking by Popularity
Information Networks
Our examples so far were mainly social networks
I.e.; nodes are people and links are relationships between people
Now we turn our attention to information networks
I.e., Nodes are chunks of information and links join related chunks
The most prominent example: the Web
There are differences between various kinds of networks but we can use the same mathematical abstractions (graphs) to reason about them
The Web as a graph

- Nodes are Web pages
- Links are hyperlinks ($< a \ href = "..." \ > ... \ < /a >$) that connect Web pages
- Links are directed (point only in one direction)
- A technical side note: why are links directed and not undirected (bidirectional) on the Web?
The Web as a graph

- Nodes are Web pages
- Links are hyperlinks (<a href="...">...</a>) that connect Web pages
- Links are directed (point only in one direction)
- A technical side note: why are links directed and not undirected (bidirectional) on the Web?
- Links consistency does not scale (deleting a target doc would require to update all the source docs)
- Tim Berners-Lee: "Let the links fail to make them scale"
Why node-link metaphor?

- Application of computer-aided authoring style known as hypertext
- Dates back to 1940’ies
- The idea: replace linear structure of text with a graph structure
- Any portion of text can link to any other portion of text in an associative manner
- Tim Berners-Lee created the Web in early 90’ies by simplifying and combining this idea with a distributed networked computer system (Internet)
Historical side note

1945 Vannevar Bush “As We May Think”

Memex: Digital library - knowledge management system, extends human brain

Index device for later retrieval

Recording information with microphone, camera

Would create trails of links connecting sequences of microfilm frames

Figure: Drawing of Bush’s theoretical Memex machine (1945)

1 www.theatlantic.com/magazine/archive/1945/07/as-we-may-think/3881/
The Web as a directed graph

- We will concentrate on the **navigational links**
- There is a huge fraction of Web pages that are stable and connected to each other by links
- Another type of links are transactional links, which are very dynamic and related to a particular transaction taking place on the Web, i.e. online payment
- However, even transactional content is linked together by a navigational “backbone”
The basic distinction between social networks and the Web is the \textit{directed} nature of the Web

Directed graphs are asymmetric: links point from one node to another

Analogy: friendship network versus name-recognition network (link from person A to person B if A has heard of B)

Name-recognition network is asymmetric: e.g. celebrities are recognizable to millions of people, but they do not recognize all of their fans

Facebook - Twitter distinction
The Web as a directed graph

Paths and Strong Connectivity. The connectivity of undirected graphs was defined in terms of paths: two nodes are linked by a path if we can follow a sequence of edges from one to the other; a graph is connected if every pair of nodes is linked by a path; and we can break up a disconnected graph into its connected components. Now that we're dealing with a directed graph, we're going to try following the same general strategy for talking about connectivity; but to do this, we first need to rework the definition of a path to take directions.
Paths and Strong Connectivity
Paths and strong connectivity

For undirected graphs:

- **Path**: a sequence of nodes such that each consecutive pair in the sequence is connected by a link.
- If every node can reach every other node by a path the graph is **connected**.
- Otherwise, a graph is disconnected and then it breaks apart into a set of connected **components**.
- **Component**: a subset of nodes such that
  1. every node in the subset has a path to every other node in that subset (internally connected)
  2. the subset is not a part of some larger connected set (stands in isolation from the rest of the graph).
Directed Paths

- **Path**: a sequence of nodes such that each consecutive pair in the sequence is connected by a link in the forward direction
- E.g. the sequence: Univ of X, Classes, Networks, Networks class blog, Blog about college rankings, USNews college rankings is a directed path
- But no directed path from Company Z’s to USNews college rankings (ignoring the direction of links there is a path between those two nodes)
Strong connectivity

- If every node can reach every other node by a (directed) path, the graph is **strongly connected**.

Is the example graph strongly connected?
Strong connectivity

- If every node can reach every other node by a (directed) path, the graph is **strongly connected**

Is the example graph strongly connected? No! Because there is no directed path from e.g. Company Z’s to USNews college rankings
Reachability

- When a directed graph is not strongly connected we are interested in its \textit{reachability properties}.
- We want to know which nodes are reachable from which other nodes using (directed) paths.
- In an undirected graph: if two nodes are in the same component then they are mutually reachable by a path.
- Otherwise if two nodes are in different components then they cannot reach each other.
Reachability

- Directed networks exhibit the following combinations:
  1. Pairs of nodes for which each can reach each other (Univ. of X and USNews: college rankings)
  2. Pairs for which one can reach the other but not vice versa (USNews college rankings and Company Z’s home page)
  3. Pairs for which neither can reach the other (I’m a student and I’m applying to college)
Strongly connected components

- **Strongly Connected Component (SCC):** a subset of nodes such that
  1. every node in the subset has a (directed) path to every other node in that subset (internally connected)
  2. the subset is not part of some larger set with the property that every node can reach every other

- Second part of definition differs from the undirected case
- SCC must not be completely isolated from the rest of the graph as in undirected case
- SCCs summarize a graph in a form of "super-nodes"
Strongly connected components
Structure of the Web
The Web as a directed graph

- Web growth rate in the 90’s was very quick
- (Broder et al., 1999) set out to build global map of the Web using strongly connected components (SCC) as basic building blocks
- Dataset: index of pages and links from AltaVista (largest search engine at that time)
- Since then, study has been replicated many times, e.g. larger datasets, subsets of the Web such as Wikipedia, etc.
The Map of the Web

- Web **not** fully interconnected network!
- Web has a giant SCC that contains interconnected pages
- Why? Major search engines and “start pages“ (e.g. directories) keep links to core
- Reachability within core is good
- Some pages from core link back to search engines
- Suppose two giant SCCs X and Y: single link from X to Y and vice versa would turn X and Y into one single SCC
IN and OUT on the Web

- Now: Position all other components in relation to giant SCC
- Classify nodes by their ability to reach and be reached from the giant SCC
- IN: nodes that can reach the giant SCC but not vice versa (e.g. new pages that have not yet been linked to)
- OUT: nodes that can be reached from the giant SCC but not vice versa (e.g. as corporate websites containing only internal links)
Example: IN and OUT

- SCC: largest in the middle of the network including e.g. Univ. of X
- I’m a student and I’m applying to college constitute
Example: IN and OUT

- SCC: largest in the middle of the network including e.g. Univ. of X
- I’m a student and I’m applying to college constitute IN
- Blog post about... and the whole component involving Company Z constitute
Example: IN and OUT

- SCC: largest in the middle of the network including e.g. Univ. of X
- I’m a student and I’m applying to college constitute IN
- Blog post about... and the whole component involving Company Z constitute OUT
There are also pages that belong to none of IN, OUT, or the giant SCC

*Tendrils and Tubes*: Connect to either IN or OUT, or both, but not to the core

*Disconnected*: Nodes that are disconnected from the SCC even if we ignore the link direction
Example

Figure 13.8: A directed graph of Web pages.

(b) Name an edge you could add or delete from the graph in Figure 13.8 so as to increase the size of the set \( \text{IN} \).

(c) Name an edge you could add or delete from the graph in Figure 13.8 so as to increase the size of the set \( \text{OUT} \).

3. In Exercise 2, we considered how the constituent parts of the bow-tie structure change as edges are added to or removed from the graph. It's also interesting to ask about the magnitude of these changes.

(a) Describe an example of a graph where removing a single edge can reduce the size of the largest strongly connected component by at least 1000 nodes. (Clearly you shouldn't attempt to draw the full graph; rather, you can describe it in words, and also draw a schematic picture if it's useful.)

(b) Describe an example of a graph where adding a single edge can reduce the size of the set \( \text{OUT} \) by at least 1000 nodes. (Again, you should describe the graph rather than actually drawing it.)
Exercise 1: Which nodes constitute the largest SCC? Which nodes belong to the IN of this SCC? Which to OUT?

(b) Name an edge you could add or delete from the graph in Figure 13.8 so as to increase the size of the set IN.

(c) Name an edge you could add or delete from the graph in Figure 13.8 so as to increase the size of the set OUT.

In Exercise 2, we considered how the constituent parts of the bow-tie structure change as edges are added to or removed from the graph. It's also interesting to ask about the magnitude of these changes.

(a) Describe an example of a graph where removing a single edge can reduce the size of the largest strongly connected component by at least 1000 nodes. (Clearly you shouldn't attempt to draw the full graph; rather, you can describe it in words, and also draw a schematic picture if it's useful.)

(b) Describe an example of a graph where adding a single edge can reduce the size of the set OUT by at least 1000 nodes. (Again, you should describe the graph rather than actually drawing it.)
Exercise 2: Pages can move between different parts of the bow-tie structure as new links are created and old ones are removed.

- Name a link (add or delete) so as to increase the size of the largest SCC.
- Name a link (add or delete) so as to increase the size of IN.
- Name a link (add or delete) so as to increase the size of OUT.
Exercise 3: Describe an example of a graph where removing a single link can reduce the size of the largest SCC by a large number, e.g. 1000 nodes.

Describe an example of a graph where adding a single link can reduce the size of the OUT by a large number, e.g. 1000 nodes.
Web Search
Search

- Search is a hard problem, not only on the Web but also in other settings.
- The field of Information Retrieval (IR) has dealt with this problem for decades now.
- Goal of IR: given an information need ("a query"), find e.g. documents that are relevant and rank them accordingly.
- Ranking algorithms:
  - Based on several aspects (terms, links, etc.).
  - Overall score can be a combination of (i) Content score (TF*IDF) and (ii) Popularity score (PageRank, HITS, etc.).
- Why is search hard (traditionally)?
Search is a hard problem, not only on the Web but also in other settings.

The field of Information Retrieval (IR) has dealt with this problem for decades now.

Goal of IR: given an information need ("a query"), find e.g. documents that are relevant and rank them accordingly.

Ranking algorithms:
- Based on several aspects (terms, links, etc.)
- Overall score can be a combination of (i) Content score (TF*IDF) and (ii) Popularity score (PageRank, HITS, etc.)

Why is search hard (traditionally)?
- Keywords are a very limited way to express complex information needs
- Problem of *synonymy* (multiple ways to say the same thing)
- Problem of *polysemy* (multiple meanings of the same term)
The problem of ranking

Why is Web search even harder?

Diversity of Web content (no controlled vocabulary, various styles, internet slang, emojis, etc.)

Diversity of Web queries (different people use different terms for the same things)

Dynamic and constantly changing nature of Web content

Abundance: queries result in millions of hits

Search engine can find and index millions of documents that are relevant to a one-word query

However: humans perform the search - can look at only a few of the results - Which ones?
The problem of ranking

Why is Web search even harder?

- Diversity of Web content (no controlled vocabulary, various styles, internet slang, emojis, etc.)
- Diversity of Web queries (different people use different terms for the same things)
- Dynamic and constantly changing nature of Web content
- Abundance: queries result in millions of hits
- Search engine can find and index millions of documents that are relevant to a one-word query
- However: humans perform the search - can look at only a few of the results - **Which ones?**
Link Analysis and Ranking by Popularity
In response to word query "Graz" at Google www.tugraz.at is retrieved in the Top 10 results

- What are the clues that suggest that www.tugraz.at is a good answer to query "Graz" (or a much better answer than 55 million of other Web pages about Graz)
- A natural way to address this if we take a “link perspective”
- Not really any way to use features purely internal to www.tugraz.at to solve this problem (all 55 million pages have Graz internally)
- Rather, www.tugraz.at stands out because of links on other Web pages that include Graz
- Very often other pages relevant to Graz point to www.tugraz.at with a link
Voting by in-links

- We can use links to assess the *authority* of a page on a topic
- Implicit endorsements through links of other pages
- Each individual link may have many possible meanings
- E.g. it may be off-topic, it may be a critique, it may be a paid advertisement, or it may be a real endorsement
- We assume that in aggregate if a page receives many links from other relevant pages then it receives a kind of *collective endorsement*
Voting by in-links

This can be operationalized as follows:

- First: collect a large sample of pages relevant to a query, e.g. Graz
- E.g. by means of classical text-based information retrieval
- Then: let pages in this sample to vote through their links
- Then: rank the pages according to the number of votes that they receive
List-finding technique

We can further extend the voting mechanism

- Consider typical example of a one-word query: newspapers
- No single best answer here but a number of them
- E.g. all prominent newspapers on the Web
- Ideal answer: list of these newspapers
Voting by in-links

Let us apply voting mechanism on an example:

- First, collect a sample of pages relevant to the query “newspapers”
- Then: count votes to pages within this sample
- Typically, high scores for a mix of prominent newspapers
- Also, some prominent Web sites (e.g. Facebook) not directly related to newspapers will get a lot of votes
- Can you think of a reason why?
Voting by in-links

Let us apply voting mechanism on an example:

- First, collect a sample of pages relevant to the query “newspapers”
- Then: count votes to pages within this sample
- Typically, high scores for a mix of prominent newspapers
- Also, some prominent Web sites (e.g. Facebook) not directly related to newspapers will get a lot of votes
- Can you think of a reason why?
- Reason: such pages have a lot of in-links no matter what the query is
A List-Finding Technique.

It’s possible to make deeper use of the network structure than just counting in-links, and this brings us to the second part of the argument that links are essential. Consider, as a typical example, the one-word query “newspapers.” Unlike the query “Cornell,” there is not necessarily a single, intuitively “best” answer here; there are a number of prominent newspapers on the Web, and an ideal answer would consist of a list of the most prominent among them. With the query “Cornell,” we discussed collecting a sample of pages relevant to the query and then let them vote using their links. What happens if we try this for the query “newspapers”?

What you will typically observe, if you try this experiment, is that you get high scores for a mix of prominent newspapers (i.e. the results you’d want) along with pages that are going to receive a lot of in-links no matter what the query is — pages like Yahoo!, Facebook, Amazon, and others. In other words, to make up a very simple hyperlink structure for purposes of...
List-finding technique

- Voting by in-links: only a very simple kind of measure
- In addition to most prominent newspapers, also other kinds of useful answers to query available
- E.g., pages that compile lists of resources relevant to the topic (lists of newspapers, etc.)
- Such lists exist for many broad enough queries (e.g. universities, hotels)
List-finding technique

We will discuss a useful technique for finding good lists:

- Among the pages casting votes, only a few of them exist that vote for many of the pages that receive votes
- Such pages have some sense where good answers are
- Thus, should be scored high as lists
Newspapers example

Page’s value as a list: equal to the sum of the votes received by all pages that it voted for
Newspapers example

Page’s value as a list: equal to the sum of the votes received by all pages that it voted for

```
Page's value as a list: equal to the sum of the votes received by all pages that it voted for

Wall St. Journal  2 votes
New York Times    4 votes
USA Today         3 votes
Yahoo!            1 vote
Amazon            3 votes
Facebook          6 votes
SJ Merc News      8 votes

Figure 14.2: Finding good lists for the query "newspapers": each page's value as a list is written as a number inside it.

In this example, we'd see something like Figure 14.1: the unlabeled circles represent our sample of pages relevant to the query "newspapers," and among the four pages receiving the most votes from them, two are newspapers (New York Times and USA Today) and two are not (Yahoo! and Amazon). This example is designed to be small enough to try by hand; in a real setting, of course there would be many plausible newspaper pages and many more off-topic pages.

But votes are only a very simple kind of measure that we can get from the link structure—there is much more to be discovered if we look more closely. To try getting more, we ask a different question. In addition to the newspapers themselves, there is another kind of useful answer to our query: pages that compile lists of resources relevant to the topic. Such pages exist for most broad enough queries: for "newspapers," they would correspond to lists.
The Principle of Repeated Improvement

- Pages scoring well as lists have a better sense for where the good results are
- We should weight their votes more heavily
- Thus, count the votes again
- This time we give each page’s vote a weight equal to its value as a list
The Principle of Repeated Improvement

- Pages scoring well as lists have a better sense for where the good results are
- We should weight their votes more heavily
- Thus, count the votes again
- This time we give each page's vote a weight equal to its value as a list

Intuition from our daily lives: Suppose you hear restaurant recommendations from a lot of people. You discover that certain restaurants get mentioned a lot by a lot of people. Certain *people* had mentioned the highly recommended restaurants when you asked them. They are like the high valued lists on the Web. You’ll take their recommendations more seriously since you trust their judgement.
Newspapers example

- Now the other newspapers (i.e., SJ Merc News, Wall Str. Journal) have surpassed the initially high-scoring Yahoo and Amazon.
- Reason: they were endorsed by pages that were estimated to be good lists.
- Suppose that you want to buy a new sci-fi book and get a lot of recommendations from your friends.
- But there are some friends that you will trust more on this issue because you know that they are e.g. dedicated sci-fi fans (i.e. there are good lists).
Repeated improvement

- Finally, we don’t need to stop after one step of reweighting but could continue
- We use now refined votes to refine list scores, then again refine votes/scores, etc.
- We can repeat this process for as many steps as we want
- *Principle of Repeated Improvement:* Each refinement to one side of the figure enables a further refinement to the other
- In typical case, all numbers will converge and will not change with new refinements
Hubs and Authorities
Let us specify this ranking procedure more precisely:

- We call pages prominent and highly endorsed for a query *authorities*
- The high value lists are called *hubs*
- For each page \( p \) we try to estimate its value as a potential authority \( auth(p) \) and its value as a potential hub \( hub(p) \)
- Each of these is initially 1, as we don’t know the best in either of these categories yet
Voting

- Voting procedure: Here, we use the quality of hubs to refine estimates for the quality of authorities.
- Authority Update Rule: For each page $p$, update $auth(p)$ to be the sum of the $hub(q)$ scores of all pages $q$ that point to it.
List-finding technique

- List-finding procedure: in which we use the quality of authorities to refine estimates for the quality of hubs
- **Hub Update Rule**: For each page $p$, update $hub(p)$ to be the sum of the $auth(q)$ scores of all pages $q$ that it points to
Repeated improvement step in which we start with all hub scores and all authority scores equal to 1

- We choose a number of steps $k$
- We then perform a sequence of $k$ hub-authority updates, where in each update:
  1. First: apply the Authority Update Rule to the current set of scores
  2. Then: apply the Hub Update Rule to the resulting set of scores

- At the end, the hub and authority scores may have very large numbers - therefore we normalize them to a probability distribution\(^2\)

---
\(^2\)divide each authority score by sum of all authority score, same with hub score.
Newspapers example

Result of normalizing the authority scores, e.g. first page: $19/125 = 0.152$
Normalized values converge to limits as $k$ goes infinity

Same limits even if other initial hub/authority vals

Kind of equilibrium: relative sizes remain unchanged if authority update rule or hub update rule is applied

A page’s authority score proportional to hub scores of pages that point to the page and vice versa
Intuition behind hubs and authorities is based on idea that pages play multiple roles in the network.

In particular, pages (hubs) can strongly endorse other pages without themselves being heavily endorsed.

Real-life example: E.g. competing firms will not link to each other.

The only way to pull them together is through a set of hubs that link to all of them at once.
Hubs and authorities: Summary

- The procedure we discussed corresponds to the Hypertext Induced Topic Selection (HITS) algorithm (Kleinberg 1999).
- Used for rating and ranking websites based on the link information when identifying topic areas (search query dependent).
- Mutual reinforcement between Web pages: “A better hub points to many good authorities, and a better authority is pointed to by many good hubs.”
- Authority value: Sum of the scaled hub values that point to that page.
- Hub value: Sum of the scaled authority values of the pages it points to.
PageRank
PageRank

- Endorsement also can be viewed as passing directly from one prominent page to another
- A page is important if it is linked from other important pages
- This forms the basis for PageRank
- Unlike HITS, PageRank is independent of the search query!
- PageRank calculation also starts with simple voting based on in-links and gets then repeatedly improved
The basic definition of PageRank

- Intuitively, we can think of PageRank as a kind of fluid that flows through the network.
- The fluid passes from node to node across links.
- It pools at the nodes that are the most important.
The basic definition of PageRank

- We compute PageRank in the following way:
  1. In a network with \( n \) nodes we assign all nodes with the initial PageRank \( 1/n \)
  2. We choose a number of steps \( k \)
  3. We perform a sequence of \( k \) updates using Basic PageRank Update Rule: each page divides and passes its current PageRank equally across its out-going links. Each page updates its new PageRank to be the sum of the shares it receives.
PageRank Example: First two steps

The basic definition of PageRank.

Intuitively, we can think of PageRank as a kind of “fluid” that circulates through the network, passing from node to node across edges, and pooling at the nodes that are the most important. Specifically, PageRank is computed as follows.

1. In a network with $n$ nodes, we assign all nodes the same initial PageRank, set to be $1/n$.
2. We choose a number of steps $k$.
3. We then perform a sequence of $k$ updates to the PageRank values, using the following rule for each update:

   **Basic PageRank Update Rule:** Each page divides its current PageRank equally across its out-going links, and passes these equal shares to the pages it points to. (If a page has no out-going links, it passes all its current PageRank to itself.) Each page updates its new PageRank to be the sum of the shares it receives.

The diagram shows a network of eight pages, labeled A to H. Page A has the largest PageRank, followed by B and C (which collect endorsements from A).

The table below illustrates the calculation of PageRank for the first two steps:

<table>
<thead>
<tr>
<th>Page</th>
<th>Step 1</th>
<th>Step 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1/2</td>
<td>5/16</td>
</tr>
<tr>
<td>B</td>
<td>1/16</td>
<td>1/4</td>
</tr>
<tr>
<td>C</td>
<td>1/16</td>
<td>1/4</td>
</tr>
<tr>
<td>D</td>
<td>1/16</td>
<td>1/32</td>
</tr>
<tr>
<td>E</td>
<td>1/16</td>
<td>1/32</td>
</tr>
<tr>
<td>F</td>
<td>1/16</td>
<td>1/32</td>
</tr>
<tr>
<td>G</td>
<td>1/16</td>
<td>1/32</td>
</tr>
<tr>
<td>H</td>
<td>1/16</td>
<td>1/16</td>
</tr>
</tbody>
</table>

Careful: error in book, page 408!

Elisabeth Lex (KTI, TU Graz)
All pages start out with a PageRank of $1/8$. PR(A) = $1/2$. It gets all of F, G, H and half of D and E. What about B and C?
PageRank Example: First two steps

All pages start out with a PageRank of 1/8. PR(A) = 1/2. It gets all of F, G, H and half of D and E. What about B and C? PR(B) and PR(C): get half of A’s PR, so only 1/16
PageRank Example: First two steps

All pages start out with a PageRank of 1/8. PR(A) = 1/2. It gets all of F, G, H and half of D and E. What about B and C?

PR(B) and PR(C): get half of A’s PR, so only 1/16

<table>
<thead>
<tr>
<th>Step</th>
<th>Page</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>1/2</td>
<td>1/16</td>
<td>1/16</td>
<td>1/16</td>
<td>1/16</td>
<td>1/16</td>
<td>1/16</td>
<td>1/8</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>5/16</td>
<td>1/4</td>
<td>1/4</td>
<td>1/32</td>
<td>1/32</td>
<td>1/32</td>
<td>1/32</td>
<td>1/16</td>
</tr>
</tbody>
</table>
All pages start out with a PageRank of 1/8. PR(A) = 1/2. It gets all of F, G, H and half of D and E. What about B and C? PR(B) and PR(C): get half of A’s PR, so only 1/16

<table>
<thead>
<tr>
<th>Step</th>
<th>Page</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>1/2</td>
<td>1/16</td>
<td>1/16</td>
<td>1/16</td>
<td>1/16</td>
<td>1/16</td>
<td>1/16</td>
<td>1/8</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>5/16</td>
<td>1/4</td>
<td>1/4</td>
<td>1/32</td>
<td>1/32</td>
<td>1/32</td>
<td>1/32</td>
<td>1/16</td>
</tr>
</tbody>
</table>

Careful: error in book, page 408!
Equilibrium Values of PageRank

- Similarly as with hub-authority computation if we increase the number of iteration steps $k$ the values will become stable and will not change anymore.
- The calculation converges and we reach an equilibrium.
- One can prove this convergence.
- One can also prove that for a strongly connected network the equilibrium values are unique.
A basic problem with PageRank: Example

- Now, F and G point to each other and not to A
- PageRank that flows from C to F and G can never flow back to the network
- Links out of C - “slow leak”, all the PageRank ends up at F and G

If we repeat Basic PageRank Update Rule:

1. Convergence to PageRank of $1/2$ for each of F and G
2. The others? Have PageRank of 0
A basic problem with PageRank: Example

- Now, F and G point to each other and not to A
- PageRank that flows from C to F and G can never flow back to the network
- Links out of C - “slow leak”, all the PageRank ends up at F and G

If we repeat Basic PageRank Update Rule:
Convergence to PageRank of 1/2 for each of F and G
A basic problem with PageRank: Example

- Now, F and G point to each other and not to A
- PageRank that flows from C to F and G can never flow back to the network
- Links out of C - “slow leak”, all the PageRank ends up at F and G

If we repeat Basic PageRank Update Rule:
Convergence to PageRank of 1/2 for each of F and G
And the others?
A basic problem with PageRank: Example

- Now, F and G point to each other and not to A
- PageRank that flows from C to F and G can never flow back to the network
- Links out of C - “slow leak”, all the PageRank ends up at F and G

If we repeat Basic PageRank Update Rule:
Convergence to PageRank of 1/2 for each of F and G
And the others? Have PageRank of 0
A basic problem with PageRank

- Wrong nodes may end up with “all” the PageRank
- If graph is not strongly connected - complete PageRank will leak to nodes in OUT
- Therefore: extend Basic PageRank Update Rule to Scaled PageRank Update Rule
- Apply Basic Update Rule and then scale down all values by a scaling factor $s$ (strictly between 0 and 1)
- Divide the residual $1 - s$ equally over all nodes giving $(1 - s)/n$ to each
- Preserves the total PageRank in the network - based on redistribution
- Can be shown that this rule converges and that no PageRank is leaking
Repeated application of Scaled PageRank Update Rule converges to set of limiting PageRank values as number of updates $k$ goes infinity.

These limiting values form the unique equilibrium: unique set of values that remains unchanged under application of update rule.

Depend on choice of scaling factor $s$.

In practice: scaling factor $s$ usually between 0.8 and 0.9.

Do you think that these values are sufficient for our toy example of 8 nodes?
Repeated application of Scaled PageRank Update Rule converges to set of limiting PageRank values as number of updates $k$ goes infinity.

These limiting values form the unique equilibrium: unique set of values that remains unchanged under application of update rule.

Depend on choice of scaling factor $s$.

In practice: scaling factor $s$ usually between 0.8 and 0.9.

Do you think that these values are sufficient for our toy example of 8 nodes?

No, most of the PageRank will still get to F and G. With only 8 nodes, a “slow leak” isn’t actually so slow. In real-world large networks, redistribution of PageRank works well to assign nodes outside the SCC very small limiting PageRank values.
Consider someone who is randomly browsing a network of Web pages:

- They start by choosing a page at random, picking each page with equal probability.
- Then, they follow links of a sequence of $k$ steps.
- In each step, they pick a random out-going link from their current page and follow it.
- If the page has no out-going links, they stay.
- This is called a **Random Walk** on the network.
Random Walks: An equivalent definition of PageRank

**Definition**

Claim: The probability of being at a page X after k steps of this random walk is precisely the PageRank of X after k applications of the Basic PageRank Update Rule.

- Both formulations of PageRank (repeated improvement & random walks) equivalent
- Random Walk based analysis provides additional intuition for PageRank as measure of importance
- PageRank of page X: limiting probability that random walk across hyperlinks will end up at X
- E.g.: in our earlier example, the probability of the walk reaching F or G converges to 1 and when it reaches F or G, it is stuck at these nodes
- Thus: limiting probabilites of being at F or G converges to 1/2, and 0 for all other nodes
The Scaled PageRank Update Rule can also be formulated in terms of Random Walks.

Walker performs “scaled” version of the walk rather than following a random link in each step:

- With probability $s$, the walker follows a random edge as before.
- With probability $1 - s$, the walker jumps to a random node anywhere in the network, choosing each edge with equal probability.
Summary

We have learned about:

- Information Networks and the (Bow-Tie) Structure of the Web
- Connectivity and Paths in Networks
- Search and Web Search
- Link Analysis: Hubs, Authority, PageRank, Random Walks
Some Practical Examples

- PageRank on protein interaction graphs\(^3\)
- Social Media Analysis\(^4\)
- Altmetrics and Analysis of Readership Data on Mendeley\(^5\)

\(^3\)http://rsos.royalsocietypublishing.org/content/2/4/140252.abstract
\(^5\)http://arxiv.org/abs/1504.07482
Thanks for your attention - Questions?

elisabeth.lex@tugraz.at