Lecture 3 — Page Rank

36-350, Data Mining

31 August 2009

The combination of the bag-of-words representation, cosine distance, and inverse document frequency weighting forms the core of lots of information retrieval systems, because it works pretty well. However, there *is* more information in and about many documents than just this, and that too can be exploited in search. Today's lecture is about one of the most successful of these, which is to use links among documents, famously pioneered by Google (Page *et al.*, 1999; Brin and Page, 1998).

Pages on the Web, of course, don't just contain words, but also links to other pages. These reflect judgments that the linked-to pages are relevant to some matter. Sometimes, of course, people link to pages because they think they want the world to know just how awful they are, but even that is a kind of relevance, and for the most part links are more or less explicit endorsements: this is good, this is worth your time. Page rank aims to exploit this.

1 Calculating Page Rank

Start with a random web-page, say i. Suppose this page has out-going links, to pages $j_1, j_2, \ldots j_{i_n}$. A simple random walk would choose each of those links with equal probability:

$$P_{ij} = \begin{cases} \frac{1}{i_n} & \text{if } j \in \{j_1, j_2, \ldots j_{i_n}\} \\ 0 & \text{otherwise} \end{cases}$$

If starting page i has no out-going links, then $P_{ij} = 1/n$, where n = the total number of pages, for all j. That is, when the walk comes to a dead end, it re-starts to a random location.[1]

(Notice that here we are representing a document entirely by what other documents it links to.)

Let X_t be the page the random walk is visiting at time t, and $N(i, n)$ be the number of $t \leq n$ where $X_t = i$, the number of times X_t visits i. The **page rank** of a page i is how often it is visited in the course of a very long random walk:

$$\rho(i) = \lim_{n \to \infty} \frac{N(i, n)}{n}$$

[1] There is also a variant where there is a small probability of jumping to a random page at *any* stage, not just at a dead end. See the original paper, at `http://ilpubs.stanford.edu:8090/422/1/1999-66.pdf`.

How do we know this is well-defined? Maybe the ratio doesn't converge at all, or it converges to something which depends on the page we started with.

Well, we know that the random walk is a Markov chain: the state of the chain is the page being visited. (Why?) We also see that there is some probability that the chain will go from any page to any other page eventually (if only by eventually hitting a dead-end page and then randomly re-starting). So the state-space of the Markov chain is **strongly connected**. The number of pages n is finite. And remember from probability models that a finite Markov chain whose state-space is strongly connected obeys the **ergodic theorem**, which says, precisely, that the fraction of time the chain spends in any one state goes to a well-defined limit, which doesn't depend on the starting state.

So one way to calculate the page-rank is just to simulate, i.e., to do a random walk in the way I described. But this is slow, and there is another way.

Suppose that ν is a probability vector on the states, i.e., it's an n-dimensional vector whose entries are non-negative and sum to one. Then, again from probability models, if the distribution at time t is ν_t, the distribution one time-step later is

$$\nu_{t+1} = \nu_t P = \nu_0 P^t$$

with P the transition matrix we defined earlier. It's another result from probability that the ν_t keep getting closer and closer to each other, so that

$$\lim_{t\to\infty} \nu_0 P^t = \rho$$

where ρ is a special probability distribution satisying the equation

$$\rho = \rho P$$

That is, ρ is an **eigenvector** of P with **eigenvalue** 1. (There is only one such ρ if the chain is strongly connected.)[2] In fact, this ρ is the same as the ρ we get from the ergodic theorem. So rather than doing the simulation, we could just calculate the eigenvectors of P, which is often faster and more accurate than the simulation.

Unpacking the last equation, it says

$$\rho(i) = \sum_j \rho(j) P_{ij}$$

which means that pages with high page-rank are ones which are reached, with high probability, from other pages of high page-rank. This sounds circular ("a celebrity is someone who's famous for being famous"), but, as we've seen, it isn't. In fact, one way to compute it is to start with ν_0 being the uniform distribution, i.e., $\nu_0(i) = 1/n$ for all i, and then calculate $\nu_1, \nu_2, \ldots$ until the change from ν_t to ν_{t+1} is small enough to tolerate. That is, initially every page has equal page-rank, but then it shifts towards those reached by many strong links (ν_1), and then to those with many strong links from pages reached by many strong links (ν_2), and so forth.

[2] A detailed discussion here would involve the Frobenius-Perron (or Perron-Frobenius) theorem from linear algebra, which is a fascinating and central result, but that will have to wait for another class.

1.1 Shrinkage Page Rank

There is also a variant of where at *every* page, not just the dead ends, there is some (small) probability of going to a completely random page. There are some technical reason for doing this, but it can also be understood as follows. We wrote the transition matrix above as P; let's say Q is the transition matrix where every page has an equal probability of jumping to every other page (except itself). Then what we're talking about is using the transition matrix $(1 - \lambda)P + \lambda Q$, where λ is the take-a-random-jump-regardless probability. If $\lambda = 1$, then we spend an equal fraction of time on every page, and all page-ranks are equal. If $\lambda = 0$, then we get the invariant distribution ρ above. Intermediate values of ρ thus reduce differences in page-rank. This is an example of what is called **shrinkage**, where estimates are "shrunk" towards a common value. This is typically done to reduce variance in estimates, at the cost of (possibly) introducing bias.

2 Page rank in Web search

There is a very simple way to use page-rank to do search:

- Calculate ρ once. (This needs the link representation for each document.)
- Given a query Q, find all the pages containing all the terms in Q. (This needs the bag of words for each document.)
- Return the matching page i where $\rho(i)$ is highest (or the k pages with the highest page-rank, etc.)

However, this is *too* simple — it presumes that a highly-linked-to page is always good, no matter how tangential it might be to the topic at hand. From the beginning, Google has used a combination of page-rank, similarity scores, and many other things (most of them properietary) to determine its search results.

(One reason to not just rely on page rank is that spammers know Google uses it, so they make "link farms", large collections of meaningless pages full of keywords and ads, which all link to each other and not to anything outside. Once the random walk enters these sub-networks, it will stay there for a very long time, so they get high page rank. Of course there has to be a way in to the link farm from the rest of the web, which these days is typically provided by comment spam. Designing a system which can *automatically* tell the difference between a link farm and something like Wikipedia, or LiveJournal, or TV Tropes, is not easy; cf. Fig. 1.)

Adding page rank to web search does indeed make it work a lot better. Human beings are a lot better, still, at dealing with meanings than computers are, and in effect we're using lots of people to do relevance judgments for us *for free*. In the early days, this lead to a lot of talk about how Google was radically democratizing the public sphere, subverting established media/cultural

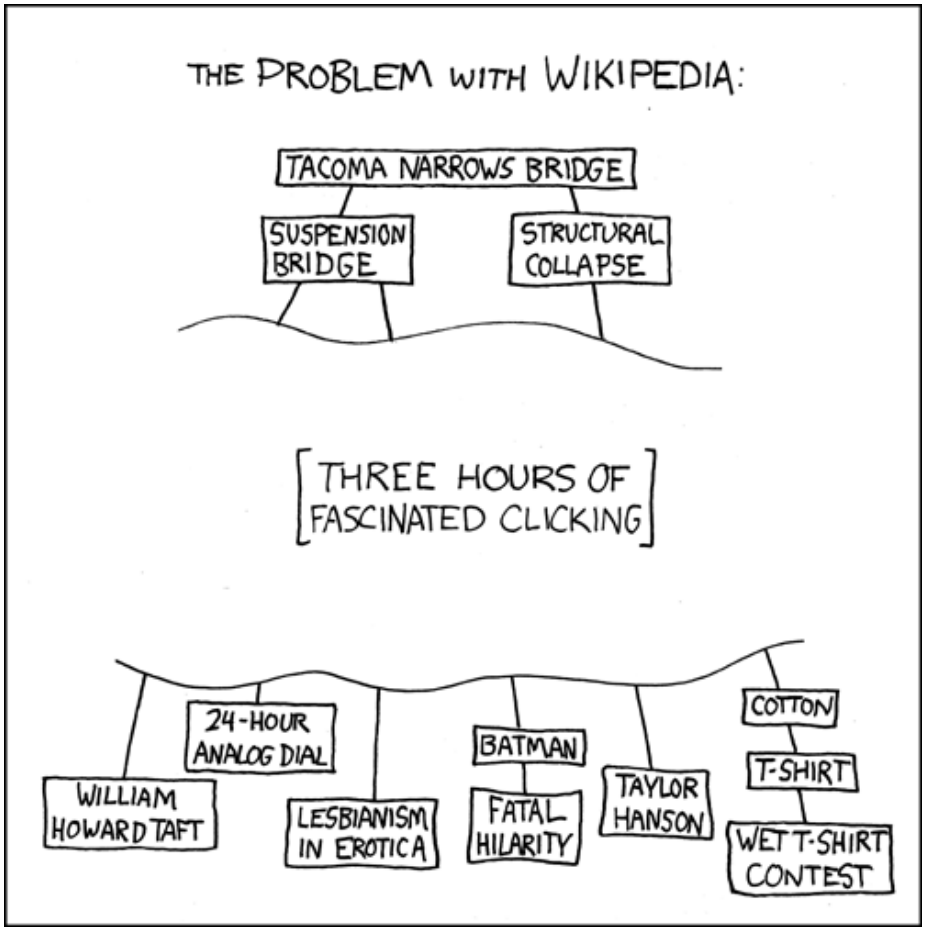

Figure 1: How do we distinguish this, automatically, from a link farm? (By Randall Munroe, `http://xkcd.com/214/`.)

hierarchies, etc. Much was made of the fact that if you searched for "Barbie" in 2005, your top hits were:[3]

barbie.com
Barbie Collecter
AdiosBarbie.com
Barbie Bazaar
If You Were a Barbie, Which Messed Up Version would you be?
Visible Barbie project
Barbie: The Image of us all (1995 undergraduate paper)
Andigraph.free.fre (Barbie and Ken sex animation)
Suicide bomber Barbie
Barbies (dressed and painted as countercultural images)

Now, of course, if you do the same search, the top hits are:

Barbie.com - Activities and Games for Girls Online! (together with eight other links to My Scene, Evertythingggirl, Polly Pocket, Kellyclub, and so on).
Barbie.com - Fun and Games
Barbie - Wikipedia, the free encyclopedia
News results for barbie (with several other links)
Barbie Collector - (The official Mattel site for Barbie Collector)
Barbie.co.uk - Activities and Games for Girls Online!
Barbie.ca
Barbie Girls - and a sublink
Celebrate 50 Years of Barbie
Video results for barbie - with two links to Aqua's Barbie Girl video

This should *not* be surprising. If your search process *succeeds* in aggregating what large numbers of people think, it will mostly reproduce established, mainstream cultural hierarchies, *by definition*. I leave you to draw your own conclusions about the Utopian potential of page-rank.

3 Other Uses and Competitors

Computationally, all that matters is that there is a set of nodes with links between them; the same algorithm could be applied to any sort of graph or network. EigenFactor (`eigenfactor.org`) is a site which ranks academic journals by using the citations in the papers they publish. You could of course do the same thing with social networks, either as disclosed through something like Facebook, or through actually observing who people have various sorts of contact with.

A natural idea is to introduce some bias into the random walk, because not all links are equally valuable. Richardson and Domingos (2002) showed that you

[3]This example is stolen from `http://whimsley.typepad.com/whimsley/2009/07/googling-barbie-again.html`, which you should read.

could do much better than raw page-rank by biasing the random walker towards pages which are extra likely to be relevant, based on textual cues. Whether this is currently being used by any search engine, I don't know.

There are also other ways of using link structure to rank web-pages — Jon Kleinberg's "hubs and authorities" system distinguishes between the value of pages as *authorities* about a particular topic, and *hubs* that aggregate information about many topics (see `http://www.cs.cornell.edu/home/kleinber/auth.pdf`), and a version of this is, apparently, incorporated into Ask.com.

4 Other Tweaks to Search Engines

There are *lots* of ways to add other sorts of information into a search engine. One of the most obvious ones, which keeps getting rediscovered, is to keep track of users, and tailor the results to the user. In particular, the **search history** of a user seems like it ought to provide lots of information that could help narrow down what the user is looking for. This is a huge subject, even leaving aside the hopefully-obvious privacy concerns. One of the important aspects of this, though, is that users *change* what they are looking for, both within a session and between sessions — and of course it's not obvious what constitutes a "session". See Jones and Klinkner (2008), who tackle this problem using methods of the kind we will cover later under classification.

Outside Reading

Manning *et al.* (2008) is a new but good textbook on IR, on its way to becoming a standard. While I said that trying to get the computer to process documents the way people do is not (currently) a good approach to search, this doesn't mean that human psychology is unimportant to designing search engines, any more than human anatomy is unimportant to designing bicycles. Belew (2000) is a fascinating look at the intersection between cognitive science and information retrieval, especially Web search. I should also mention the very deep book by Harris (1988), about, among other things, the way meaning emerges in language from statistical patterns of meaningless symbols. Parts of it will probably be hard to follow until after we've seen information theory later in the course.

References

Belew, Richard K. (2000). *Finding Out About: A Cognitive Perspective on Search Engine Technology and the WWW*. Cambridge, England: Cambridge University Press.

Brin, Sergey and Lawrence Page (1998). "The anatomy of a large-scale hypertextual Web search engine." *Computer Networks and ISDN Systems*, **30**: 107–

117. URL http://infolab.stanford.edu/~backrub/google.html. Proceedings of the Seventh International World Wide Web Conference.

Harris, Zellig (1988). *Language and Information*. New York: Columbia University Press.

Jones, Rosie and Kristina Lisa Klinkner (2008). "Beyond the session timeout: automatic hierarchical segmentation of search topics in query logs." In *Proceeding of the 17th ACM Conference on Information and Knowledge Management*, pp. 699–708. New York: ACM. URL http://www.cs.cmu.edu/~rosie/papers/jonesKlinknerCIKM2008.pdf.

Manning, Christopher D., Prabhakar Raghavan and Hinrich Schütze (2008). *Introduction to Information Retireval*. Cambridge, England: Cambridge University Press.

Page, Lawrence, Sergey Brin, Rajeev Motwani and Terry Winograd (1999). *The PageRank Citation Ranking: Bringing Order to the Web*. Technical Report 1999-66, Stanford University InfoLab. URL http://ilpubs.stanford.edu:8090/422/. Previous number = SIDL-WP-1999-0120.

Richardson, Matthew and Pedro Domingos (2002). "The Intelligent Surfer: Probabilistic Combination of Link and Content Information in PageRank." In *Advances in Neural Information Processing Systems 14 (NIPS 2001)* (Thomas G. Dietterich and Suzanna Becker and Zoubin Ghahramani, eds.), pp. 1441–1448. Cambridge, Massachusetts: MIT Press. URL http://books.nips.cc/papers/files/nips14/AP17.pdf.