Untangling the Web

Untangling the Web

24

IN A TIME long ago, in the dark days before Google, searching the Web was an exercise in frustration. The sites suggested by the older search engines were too often irrelevant, while the ones you really wanted were either buried way down in the list of results or missing altogether.

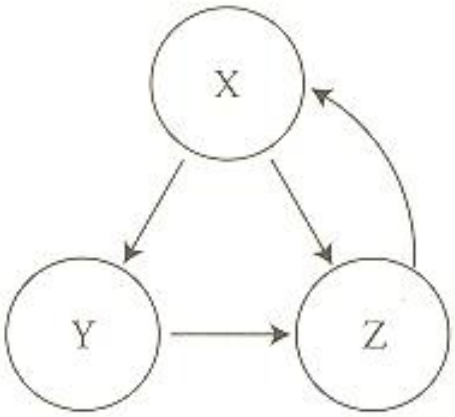
Algorithms based on link analysis solved the problem with an insight as paradoxical as a Zen koan: A Web search should return the best pages. And what, grasshopper, makes a page good? A page is good if other good pages link to it.

That sounds like circular reasoning. It is . . . which is why it's so deep. By grappling with this circle and turning it to advantage, link analysis yields a jujitsu solution to searching the Web.

The approach builds on ideas from linear algebra, the study of vectors and matrices. Whether you want to detect patterns in large data sets or perform gigantic computations involving millions of variables, linear algebra has the tools you need. Along with underpinning Google's PageRank algorithm, it has helped scientists classify human faces, analyze the voting patterns of Supreme Court justices, and win the million-dollar Netflix Prize (awarded to the person or team who could improve by

more than 10 percent Netflix's system for recommending movies to its customers).

For a case study of linear algebra in action, let's look at how PageRank works. And to bring out its essence with a minimum of fuss, let's imagine a toy Web that has just three pages, all connected like this:



The arrows indicate that page X contains a link to page Y, but Y does not return the favor. Instead, Y links to Z. Meanwhile X and Z link to each other in a frenzy of digital back-scratching.

In this little Web, which page is the most important, and which is the least? You might think there's not enough information to say because nothing is known about the pages' content. But that's old-school thinking. Worrying about content turned out to be an impractical way to rank webpages. Computers weren't good at it, and human judges couldn't keep up with the deluge of thousands of pages added each day.

The approach taken by Larry Page and Sergey Brin, the grad students who cofounded Google, was to let webpages rank themselves by voting with their feet—or, rather, with their links. In the example above, pages X and Y both link to Z,

which makes Z the only page with two incoming links. So it's the most popular page in the universe. That should count for something. However, if those links come from pages of dubious quality, that should count against them. Popularity means nothing on its own. What matters is having links from *good* pages.

Which brings us back to the riddle of the circle: A page is good if good pages link to it, but who decides which pages are good in the first place?

The network does. And here's how. (Actually, I'm skipping some details; see the notes on pages 292–293 for a more complete story.)

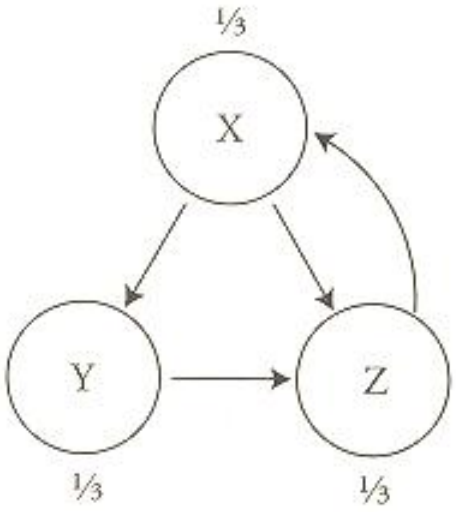
Google's algorithm assigns a fractional score between 0 and 1 to each page. That score is called its PageRank; it measures how important that page is relative to the others by computing the proportion of time that a hypothetical Web surfer would spend there. Whenever there is more than one outgoing link to choose from, the surfer selects one at random, with equal probability. Under this interpretation, pages are regarded as more important if they're visited more frequently (by this idealized surfer, not by actual Web traffic).

And because the PageRanks are defined as proportions, they have to add up to 1 when summed over the whole network. This conservation law suggests another, perhaps more palpable, way to visualize PageRank. Picture it as a fluid, a watery substance that flows through the network, draining away from bad pages and pooling at good ones. The algorithm seeks to determine how this fluid distributes itself across the network in the long run.

The answer emerges from a clever iterative process. The algorithm starts with a guess, then updates all the PageRanks by apportioning the fluid in equal shares to the outgoing links,

and it keeps doing that in a series of rounds until everything settles down and all the pages get their rightful shares.

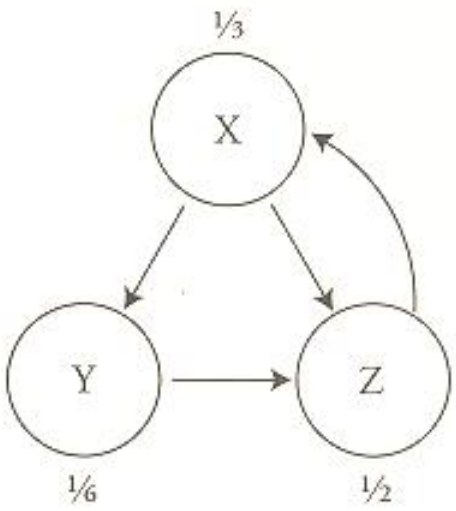
Initially the algorithm takes an egalitarian stance. It gives every page an equal portion of PageRank. Since there are three pages in the example we're considering, each page begins with a score of 1/3.



Initial values of PageRank

Next, these scores are updated to better reflect each page's true importance. The rule is that each page takes its PageRank from the last round and parcels it out equally to all the pages it links to. Thus, after one round, the updated value of X would still equal 1/3, because that's how much PageRank it receives from Z, the only page that links to it. But Y's score drops to a measly 1/6, since it gets only half of X's PageRank from the previous round. The other half goes to Z, which makes Z the big winner at this stage, since along with the 1/6 it receives

from X, it also gets the full 1/3 from Y, for a total of 1/2. So after one round, the PageRank values are those shown below:



Values of PageRank
after one update

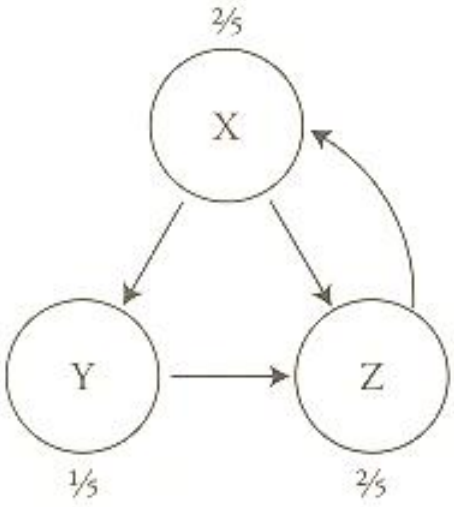
In the rounds to come, the update rule stays the same. If we write (x, y, z) for the current scores of pages X, Y, and Z, then the updated scores will be

$$x' = z$$
$$y' = \frac{1}{2}x$$
$$z' = \frac{1}{2}x + y$$

where the prime symbol in the superscript signifies that an update has occurred. This kind of iterative calculation is easy to do in a spreadsheet (or even by hand, for a network as small as the one we're studying).

After ten iterations, one finds that the numbers don't change much from one round to the next. By then, X has a 40.6 percent share of the total PageRank, Y has 19.8 percent, and Z has 39.6 percent. Those numbers look suspiciously close to 40 percent, 20 percent, and 40 percent, suggesting that the algorithm is converging to those values.

In fact, that's correct. Those limiting values are what Google's algorithm would define as *the* PageRanks for the network.



Limiting values of PageRank

The implication is that X and Z are equally important pages, even though Z has twice as many links coming in. That makes sense: X is just as important as Z because it gets the full endorsement of Z but reciprocates with only half its own endorsement. The other half it sends to Y. This also explains why Y fares only half as well as X and Z.

Remarkably, these scores can be obtained directly, without

going through the iteration. Just think about the conditions that define the steady state. If nothing changes after an update is performed, we must have $x' = x$, $y' = y$, and $z' = z$. So replace the primed variables in the update equations with their unprimed counterparts. Then we get

$$x = z$$
$$y = \tfrac{1}{2}x$$
$$z = \tfrac{1}{2}x + y$$

and this system of equations can be solved simultaneously to obtain $x = 2y = z$. Finally, since these scores must sum to 1, we conclude $x = 2/5$, $y = 1/5$, and $z = 2/5$, in agreement with the percentages found above.

Let's step back for a moment to look at how all this fits into the larger context of linear algebra. The steady-state equations above, as well as the earlier update equations with the primes in them, are typical examples of linear equations. They're called linear because they're related to lines. The variables x, y, z in them appear to the first power only, just as they do in the familiar equation for a straight line, $y = mx + b$, a staple of high-school algebra courses.

Linear equations, as opposed to those containing nonlinear terms like x^2 or yz or $\sin x$, are comparatively easy to solve. The challenge comes when there are enormous numbers of variables involved, as there are in the real Web. One of the central tasks of linear algebra, therefore, is the development of faster and faster algorithms for solving such huge sets of equations. Even slight improvements have ramifications for everything from airline scheduling to image compression.

But the greatest triumph of linear algebra, from the standpoint of real-world impact, is surely its solution to the Zen rid-

dle of ranking webpages. "A page is good insofar as good pages link to it." Translated into symbols, that criterion becomes the PageRank equations.

Google got where it is today by solving the same equations as we did here—just with a few billion more variables . . . and profits to match.
