CS 112  Introduction to Programming  
(Spring 2012)

Lecture #15: Random Web Surfer
Zhong Shao
Department of Computer Science
Yale University
Office: 314 Watson
http://flint.cs.yale.edu/cs112

Acknowledgements: Some slides used in this class are taken directly or adapted from those accompanying the textbook: Introduction to Programming in Java. An Interdisciplinary Approach by Robert Sedgewick and Kevin Wayne (Copyright 2002-2010).

Memex  
[Vannevar Bush, 1936] Theoretical hypertext computer system; pioneering concept for world wide web.
- Follow links from book or film to another.
- Tool for establishing links.

World Wide Web  
[Tim Berners-Lee, CERN 1980] Project based on hypertext for sharing and updating information among researchers.

Web Browser  
Killer application of the 1990s.
Library of Babel

La biblioteca de Babel. [Jorge Luis Borges, 1941]

When it was proclaimed that the Library contained all books, the first impression was one of extravagant happiness… There was no personal or world problem whose eloquent solution did not exist in some hexagon.

This inordinate hope was followed by an excessive depression. The certitude that some shelf in some hexagon held precious books and that these precious books were inaccessible seemed almost intolerable.

Web Search

Web search. Killer application of the 2000s.

Relevance. Is the document similar to the query term?
Importance. Is the document useful to a variety of users?

Search engine approaches.
- Paid advertisers.
- Manually created classification.
- Feature detection, based on title, text, anchors, …
- “Popularity.”
**PageRank**

*Google’s PageRank™ algorithm.* [Sergey Brin and Larry Page, 1998]
- Measure popularity of pages based on hyperlink structure of Web.
- Revolutionized access to world’s information.

**90-10 Rule**

**Model.** Web surfer chooses next page:
- 90% of the time surfer clicks random hyperlink.
- 10% of the time surfer types a random page.

**Caveat.** Crude, but useful, web surfing model.
- No one chooses links with equal probability.
- No real potential to surf directly to each page on the web.
- The 90-10 breakdown is just a guess.
- It does not take the back button or bookmarks into account.
- We can only afford to work with a small sample of the web.
- ...

**Web Graph Input Format**

**Input format.**
- N pages numbered 0 through N-1.
- Represent each hyperlink with a pair of integers.

**Transition Matrix**

*Transition matrix.* \( p(i|j) = \text{prob. that surfer moves from page } i \text{ to } j. \)

- Transition matrix example:
  - input graph:
    - 5 pages, 12 links
  - Transition matrix:
    - \( 0.1 \) to \( 1 \)
    - \( 0.5 \) to \( 2 \)
    - \( 0.1 \) to \( 3 \)
    - \( 0.1 \) to \( 4 \)
    - \( 0.1 \) to \( 1 \)

- Probabilities of moving to next page:
  - \( 0.9 \) to \( 1 \)
  - \( 0.36 \) to \( 3 \)
  - \( 0.9 \) to \( 4 \)
  - \( 0.45 \) to \( 3 \)

- Probabilities of remaining on current page:
  - \( 0.9 \) to \( 1 \)
  - \( 0.5 \) to \( 2 \)
  - \( 0.1 \) to \( 3 \)
  - \( 0.1 \) to \( 4 \)
Web Graph to Transition Matrix

```java
public class Transition {
    public static void main(String[] args) {
        int N = StdIn.readInt(); // number of pages
        int[][] counts = new int[N][N]; // links from page i to j
        int[] outDegree = new int[N]; // links from page

        // accumulate link counts
        while (!StdIn.isEmpty()) {
            int i = StdIn.readInt();
            int j = StdIn.readInt();
            outDegree[i]++;
            counts[i][j]++;
        }

        // print transition matrix
        StdOut.println(N + " \ N");
        for (int i = 0; i < N; i++) {
            for (int j = 0; j < N; j++) {
                double p = .90 * counts[i][j] / outDegree[i] + .10 / N;
                StdOut.printf("%7.5f ", p);
            }
            StdOut.println();
        }
    }
}
```

Monte Carlo Simulation

Monte Carlo simulation.
- Surfer starts on page 0.
- Repeatedly choose next page, according to transition matrix.
- Calculate how often surfer visits each page.
Random Surfer

Random move. Surfer is on page $a_{ij}$. How to choose next page $j$?

- Row $p_{ij}$ of transition matrix gives probabilities.
- Compute cumulative probabilities for row $p_{ij}$.
- Generate random number $r$ between 0.0 and 1.0.
- Choose page $j$ corresponding to interval where $r$ lies.

```java
double r = Math.random();
double sum = 0.0;
for (int j = 0; j < N; j++) {
    sum += p[j][j];
    if (r < sum) { page = j; break; }
}
```

Mathematical Context

Convergence. For the random surfer model, the fraction of time the surfer spends on each page converges to a unique distribution, independent of the starting page.

- "Page rank" of Markov chain
- "Stationary distribution" of Markov chain
- "Principal eigenvector" of transition matrix
Mixing a Markov Chain

The Power Method

Q. If the surfer starts on page 0, what is the probability that surfer ends up on page 1 after one step?

A. First row of transition matrix.

The Power Method

Q. If the surfer starts on page 0, what is the probability that surfer ends up on page 1 after two steps?

A. Matrix-vector multiplication.

The Power Method

Power method. Repeat until page ranks converge.
Mathematical Context

Convergence. For the random surfer model, the power method iterates converge to a unique distribution, independent of the starting page.

Page rank
Stationary distribution of Markov chain
Principal eigenvector of transition matrix

Random Surfer: Scientific Challenges

Google’s PageRank™ algorithm. [Sergey Brin and Larry Page, 1998]

- Rank importance of pages based on hyperlink structure of web, using 90-10 rule.
- Revolutionized access to world’s information.

Scientific challenges. Cope with 4 billion-by-4 billion matrix!
- Need data structures to enable computation.
- Need linear algebra to fully understand computation.
Random Surfer and Matrix Multiplication

Q. What is the probability that a surfer moves from page 1 to page 3 in two steps?
A. \( p^2 = p \times p \). [Matrix multiplication]

\[
P = \begin{pmatrix} .02 & .92 & .02 & .02 \\ .02 & .38 & .38 & .20 \\ .02 & .02 & .02 & .02 \\ .47 & .03 & .47 & .02 \end{pmatrix}
\]

Random Surfer: Mathematical Context

Q. What is the probability that a surfer moves from page 1 to page 3 in the limit?
A. \( \lim_{k \to \infty} P^k = P \times P \times \ldots \times P \).

**Mixing theorem.** \( P^k \) converges. Moreover, all rows are equal.

\[
P^k = \begin{pmatrix} .05 & .04 & .36 & .37 & .19 \\ .45 & .04 & .32 & .37 & .02 \\ .86 & .04 & .05 & .02 & .02 \\ .05 & .85 & .04 & .05 & .02 \\ .05 & .44 & .04 & .43 & .02 \end{pmatrix}
\]

Fraction of time surfer spends on page 3 is independent of starting point.