L25: PageRank

How Google came to rule the web!

11.26.2025

Data Mining

Jeff M. Phillips

Search Engines on the Web.

AltaVista

Q: apple computer

→ page 1
→ page 2
→ page 3 } no scrolling

⋮

→ page 10. } 1 page

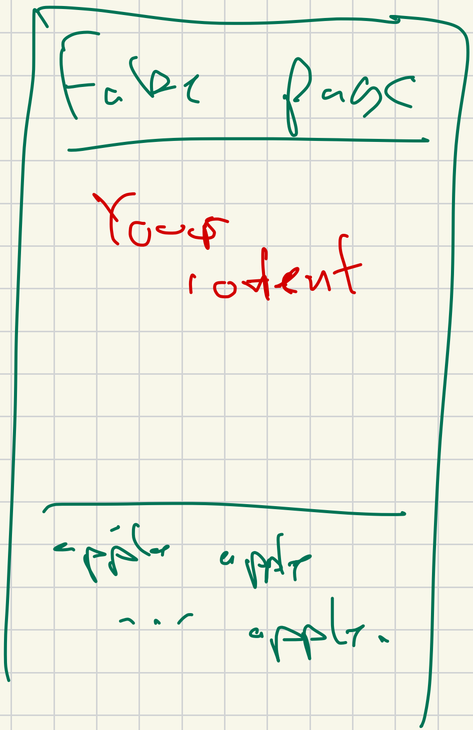
(see more)

Which pages relevant?

- most occurrences of the word "apple"
  - ↳ similarity
    - cosine
    - bag-of-word model
      - page → $v \in \mathbb{R}^{100,000}$
    - Jaccard
      - k-grams
      - past → minhashing

Problems w/ keyword only search?

- apple apple ... just repeat keyword.



→ search engines find pos. btw (apple, c > 0.1)

↳ apple computer

query (ipod, c, apple, computer)

Web browsers

inverted index

keyword 1 → page1, page2, ...

keyword 2 → page1, page7, ...

keyword 3

Crawler

bot, browser web

1. go to page
2. record info → adds in index
3. go to random link

Alternatives to Search Engines

LookSmart, Yahoo!

Built Index: Rolodex, Yellow Pages

- Page useful links
- categories

Most visited Webpages

1. Google
2. Youtube
3. Facebook
4. Instagram
5. Chat GPT
6. Twitter
7. Reddit
8. whatApp
9. Wikipedia
10. Bing
11. Yahoo Japan
12. Tiktok
13. Yahoo!

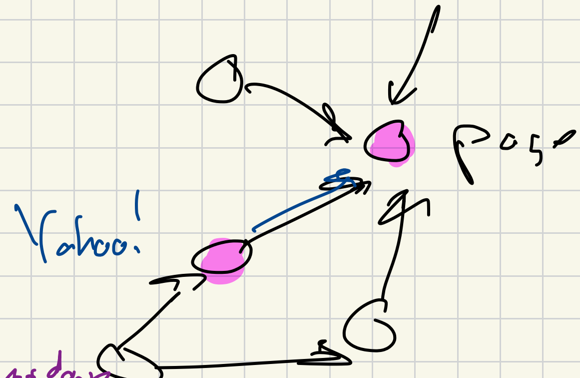
PageRank 1998

Idea: Interesting pages, links to from other interesting pages!

Indirectly which pages visited

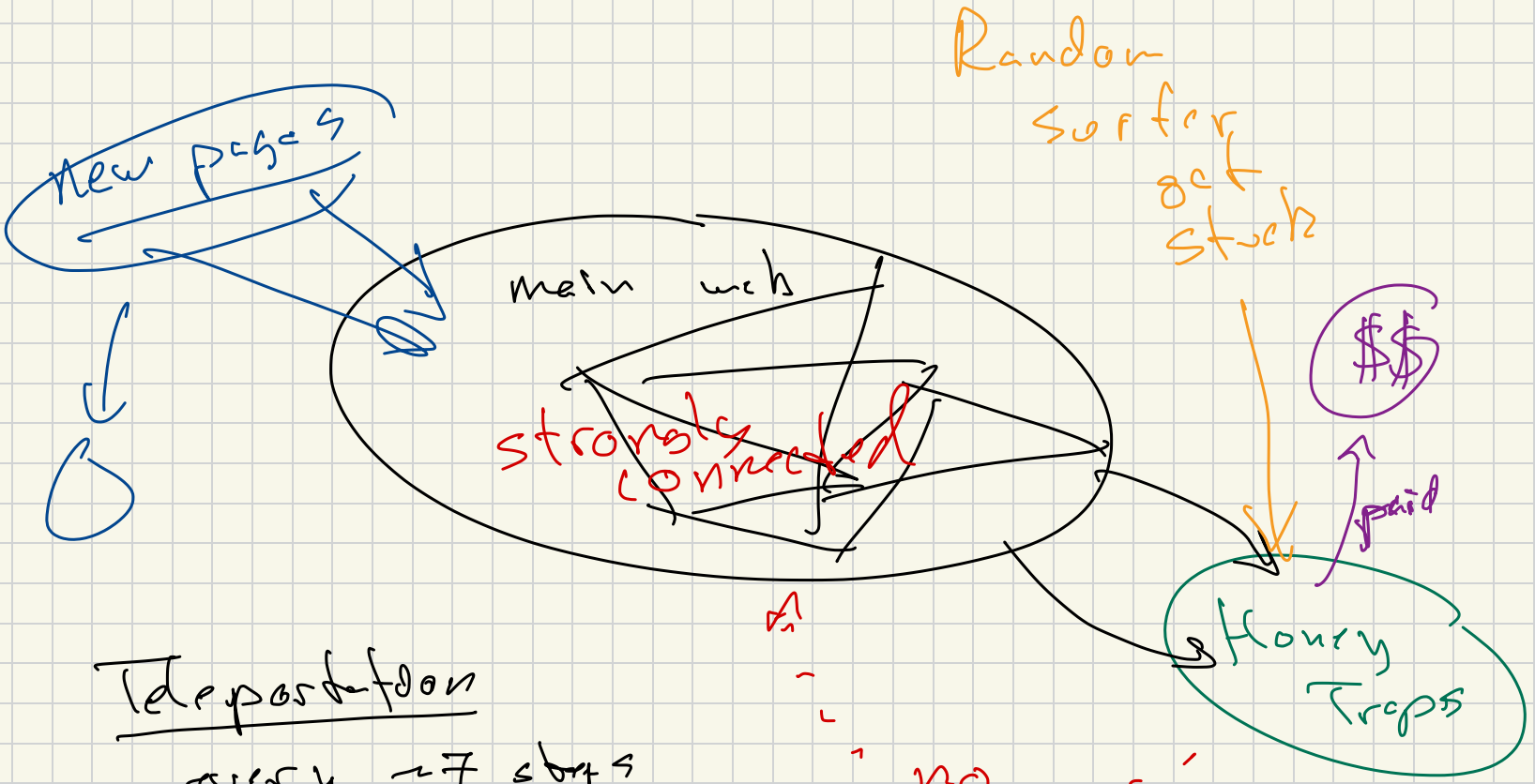
!! random surfer?!

↳ crawler



!! Built web graph!

↳ top eigenvector = prob random surfer on page.


Random surfer at start
New pages
main web
strongly connected
$$
paid
Honey Traps
Teleportation
every ~7 steps go to random page.
no
uniform random

- Web graph adjacency matrix $A \in \mathbb{R}^{n \times n}$
- L_1-normalize $A \rightarrow L \in \mathbb{R}^{n \times n}$
- Teleportation matrix $Q \in \mathbb{R}^{n \times n}$

$$Q = \frac{1}{n}\begin{bmatrix} 1 & \cdots & 1 \\ \vdots & & \vdots \\ 1 & \cdots & 1 \end{bmatrix} = \frac{1}{n}\,\text{ones}(n,n)$$

- Pagerank matrix

$$P = L(1-\beta) + \beta Q$$

0.15

L: not dense; Q: dense

$p^* = p^m q_0$ as $m \rightarrow \infty$

$p^*(j)$ = prob. random surfer on page j.

How to compute p^*?

1.- Crawler statistics = random walk on P
or on L w/ restarts

2. Matrix multiplication $P^n = \underbrace{P \cdot P \cdot P \ldots P}_{n = 20 \; 7 \text{ times}}$ (dense = out of memory)

3. eigs(P) (needed supercomputer)
↳ MapReduce

4. Power Method
repeat
$q_i = P q_{i-1} = (L(1-\beta) + \beta Q) q_{i-1}$
sparse $= L(1-\beta) q_{i-1} + \beta \boxed{1}$ ← const.