WEB SEARCH

Pengantar Temu-Kembali Informasi

Kuliah #13

12 Maret 2009

Classic Information Retrieval

- Korpus: koleksi dokumen yang sudah baku dan tetap
- Tujuan: menemukan dokumen yang relevan dengan kebutuhan user (diimplementasikan dalam bentuk query)

Sejarah Search Engine

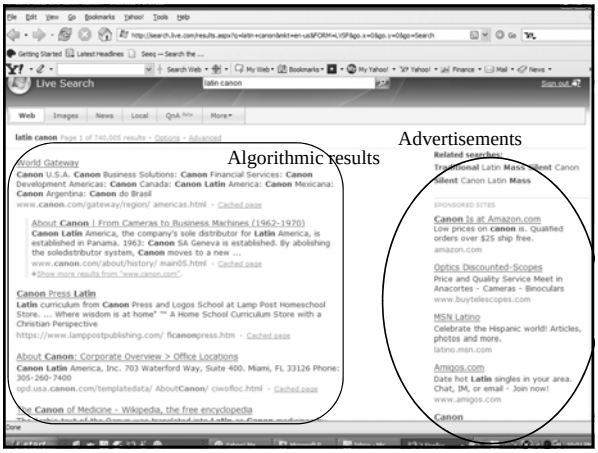
- Di akhir 1980's banyak berkas yang tersedia melalui anonymous FTP.
- Pada tahun 1990, Alan Emtage dari McGill University mengembangkan Archie (kependekan dari "archives")
  - Mengumpulkan daftar dari berkas2 yang tersedia pada server FTP.
  - Menggunakan regex untuk melacak nama-nama berkas ini.
- Pada tahun 1993, Veronica dan Jughead dikembangkan untuk melacak nama-nama berkas teks yang tersedia melalui server Gopher.

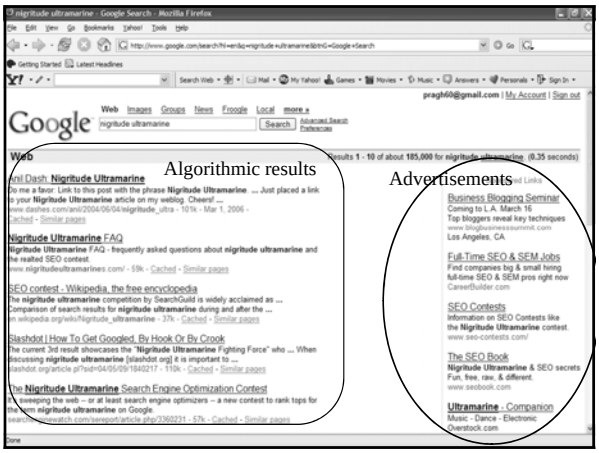
Sejarah Web Search

- Pada 1993, web robots (spiders) pemula dibuat untuk mengumpulkan URL's:
  - Wanderer
  - ALIWEB (Pengindeks WEB seperti Archie)
  - WWW Worm (mengindeks URL dan judul2 agar bisa dilacak dengan regex)
- Pada 1994, mahasiswa pasca dari Stanford (David Filo dan Jerry Yang) mulai mengumpulkan web sites yang digemari secara manual menjadi suatu topical hierarchy yang disebut Yahoo.

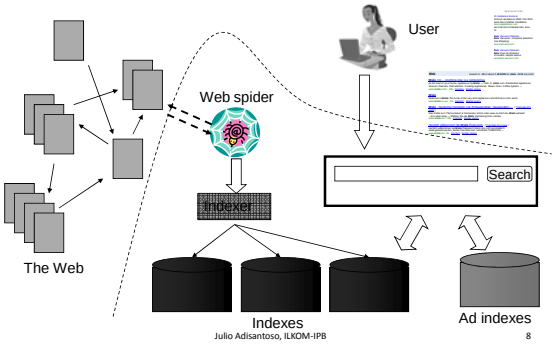
Sejarah Web Search

- Pada awal 1994, Brian Pinkerton mengembangkan WebCrawler sebagai proyek kelas di University of Wash (kemudian menjadi bagian dari Excite dan AOL).
- Beberapa bulan kemudian, Fuzzy Maudlin, mahasiswa pasca mengembangkan Lycos, yang pertama menggunakan sistem IR standar seperti yang dikembangkan untuk proyek DARPA Tipster, dan yang pertama mengindeks sejumlah besar wepages.
- Di akhir 1995, DEC mengembangkan Altavista. Menggunakan mesin Alpha yang secara cepat memproses sejumlah besar query. Bisa memproses operator boolean, frase, "reverse pointer" queries.
- Pada 1998, Larry Page dan Sergey Brin, mahasiswa di Stanford University, memulai Google.





Dasar Web Search



Julio Adisantoso, ILKOM-IPB 8

Komponen Web Search

- Spider (crawler/robot) – membangun korpus
- Indexer – membuat inverted index
- Query processor – menyajikan hasil query
  - Front end – query reformulation, word stemming, etc.
  - Back end – finds matching documents and ranks them

Julio Adisantoso, ILKOM-IPB 9

Spider (crawler/robot)

- Membangun korpus dari URL di seluruh jaringan Internet.
- Mengumpulkan halaman web secara rekursif
  - Untuk tiap URL, ambil halaman, parsing dan ekstrak
  - Dilakukan berulang sesuai periode yang ditetapkan

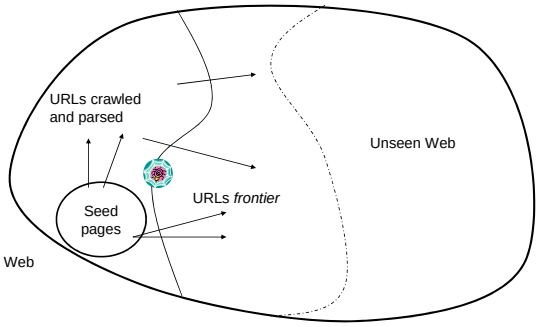
Julio Adisantoso, ILKOM-IPB 10

Spider (crawler/robot)

- Mulai dengan halaman seed yang diketahui.
- Fetch and parse
  - Ekstrak URL
  - Tempatkan URL yang diekstrak tersebut ke dalam antrian
- Fetch tiap URL dalam antrian dan ulangi

Julio Adisantoso, ILKOM-IPB 11

Crawling

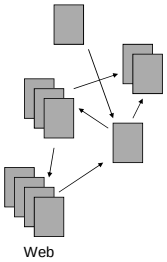


Julio Adisantoso, ILKOM-IPB 12

Masalah pada Crawler

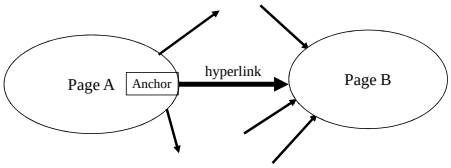
- Masalah pada crawler
  - Banyak pekerjaan yangg hanya dilakukan sekali atau diulangi
  - Kemana harus pergi?
  - Harus adil terhadap web-pages
- Pemecahan
  - Distributed crawling
  - Pengurutan untuk crawling
  - Teknik re-visiting

Web



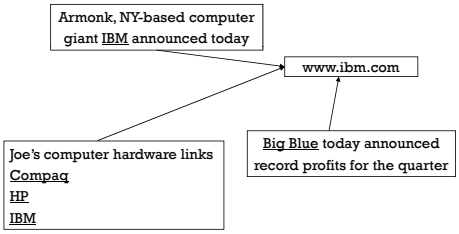
- Data yang terdistribusi: Dokumen tersebar pada lebih dari sejuta web servers yang berbeda.
- Data yang mudah berubah: Banyak dokumen berubah atau hilang dengan cepat.
- Volume yang besar: Dokumen berbeda dalam jumlah milyaran.
- Data yang tidak terstruktur dan terulang: Tidak ada struktur yang sama, kesalahan pada HTML.
- Kualitas Data: Tidak ada pengeditan, informasi yang salah, penulisan yang buruk, salah tulis, dsb.
- Data yang heterogen: Jenis media yang bervariasi (gambar, video), bahasa, set karakter, dsb.

Web sebagai Directed Graph



Indexing anchor text

- Ketika meng-indeks dokumen D, sertakan anchor text dari link ke D

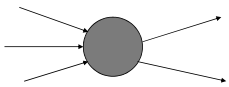


Indexing anchor text

- Ambil anchor text (antara <a> dan) dari tiap link yang mengikutinya.
- Anchor text biasanya adalah deskripsi dari dokumen yang ditunjuknya.
- Tambahkan anchor text pada isi dari halaman tujuan untuk memberikan tambahan kata indeks yang relevan.
- Contoh:
  - Evil Empire
  - IBM

Query-independent ordering

- Generasi pertama : menggunakan banyaknya link sebagai ukuran popularitas paling sederhana.
- Dua ukuran dasar:
  - Undirected popularity : tiap halaman diberi skor = banyaknya in-link ditambah out-link (5=3+2).
  - Directed popularity : skor halaman = banyaknya in-link (3)

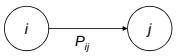


Query processing

- Pertama, ambil semua halaman yang memiliki teks pada query.
- Urutkan halaman berdasarkan ukuran popularitas.
- Cara lain : Algoritma Page Rank dari Google

Rantai Markov (Markov Chain)

- Rantai Markov terdiri dari n states, dan nxn matrik peluang transisi (P).
- Pada setiap tahap, ada tepat satu state.
- Untuk $1 \le l,j \le n$, P_{ij} adalah peluang state *j* terjadi setelah state *i*.



Matrik Peluang Transisi

- Contoh kasus : 1→2, 3→2, 2→1, 2→3
- Buat matrik A, dimana A_{ij}=1 jika ada link *i* ke *j*, dan A_{ij}=0 jika tidak ada link *i* ke *j*.

$$A = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

- Lakukan proses:
  - Bagi setiap nilai 1 dengan banyaknya nilai 1 pada suatu baris.
  - Kalikan hasil matrik dengan 1-α (dampening factor), supaya tidak ada nilai 0.
  - Tambahkan α/N ke setiap elemen matrik untuk memperoleh matrik P

Matrik Peluang Transisi

- Misalkan α=0.5, maka diperoleh

$$P = \begin{pmatrix} 1/6 & 2/3 & 1/6 \\ 5/12 & 1/6 & 5/12 \\ 1/6 & 2/3 & 1/6 \end{pmatrix}$$

- Lakukan proses iterasi sampai nilai vektor konvergen.

Matrik Peluang Transisi

- Misalkan dimulai dari state 1 dengan vektor inisial x_0=(1 0 0).
- Maka setelah satu tahap iterasi, diperoleh:

  $x_0P = (1/6 \quad 2/3 \quad 1/6) = x_1$

- Setelah dua tahap iterasi diperoleh:

$$x_1P = \begin{pmatrix} 1/6 & 2/3 & 1/6 \end{pmatrix} \begin{pmatrix} 1/6 & 2/3 & 1/6 \\ 5/12 & 1/6 & 5/12 \\ 1/6 & 2/3 & 1/6 \end{pmatrix}$$

$$= \begin{pmatrix} 1/3 & 1/3 & 1/3 \end{pmatrix} = x_2$$

Page Rank

- Proses iterasi sampai konvergen
- Contoh sebelumnya:
  - x_0=(1 0 0).
  - x_1=(1/6 2/3 1/6).
  - x_2=(1/3 1/3 1/3).
  - dst.
  - x=(5/18 4/9 5/18) → nilai Page Rank

Proses Query dengan Page Rank

- Preprocessing:
  - Berdasarkan link setiap halaman, buat matrik **P**.
  - Lakukan iterasi untuk mendapatkan Page Rank dari setiap halaman *i*.
- Query processing:
  - Retrieve halaman sesuai dengan query.
  - Urutkan halaman berdasarkan nilai Page Rank.
  - Urutan tersebut merupakan query-*independent*

Persiapan Presentasi Tugas

Ketentuan

- Presentasi dilakukan oleh satu mahasiswa dari setiap kelompok.
- Waktu presentasi hanya 10 menit (termasuk demo program) dengan jumlah slide paling banyak 4 halaman:
  - Slide 1 : sekilas tentang topik tsb.
  - Slide 2 : ruang lingkup dan batasan
  - Slide 3 : arsitektur sistem yang dibuat
  - Slide 4 : pembagian tugas setiap anggota
- Semua mahasiswa wajib hadir dalam presentasi

Ketentuan

- Urutan kelompok akan ditentukan pada saat acara. Tidak boleh ada penundaan dengan sebab apa pun.
- Seluruh sistem yang akan didemokan dan file powerpoint telah disimpan di dalam salah satu komputer yang ada di lab SeIns (silakan dikoordinasikan oleh Komti).
- Waktu presentasi akan diumumkan pada saat UAS nanti.