Implementasi *Breadth-First Search: PageRank Algorithm* dan Aplikasinya dalam Riset Kanker

Yulius Nainggolan / 13510090
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
yulius@itb.ac.id

***Abstract*—Tingkat kepentingan dari sebuah halaman web menjadi hal yang penting seiring dengan tumbuhnya jumlah website. Penggunaan web *crawler* yang merupakan salah satu implementasi dari BFS sekarang menjadi sangat penting. PageRank juga berkembang dengan semakin banyaknya website yang ada di seluruh dunia, dengan salah satu yang paling terkenal yaitu Google PageRank. Ternyata pada bidang keilmuan yang lain pun, algoritma pagerank ternyata mampu diterapkan, seperti untuk alat bantu dalam peperangan melawan kanker**

***Index Terms*—BFS, web *crawler*, PageRank, NetRank.**

I. Introduction

BFS (*Breadth-First Search)* sebagai salah satu algoritma yang menjelajah simpul-simpul terdekat, memiliki banyak sekali macam implementasi, baik di budang Informatika maupun di budang lain

I.1 Web *Crawler*

Web *crawler* adalah sebuah *software-agent.* Web *crawler* secara umum tugasnya adalah mencari data yang dibutuhkan dengan cara menjelajah URL. Secara umum, web *crawler* menjelajah dari halaman pertama, yang disebut *seeds.* Ketika web *crawler* menjelajah halaman-halaman tersebut, web *crawler* mengidentifikasi semua *hyperlink* yang terdapat pada halaman tersebut, kemudian menambahkannya pada daftar halaman yang telah dikunjungi.

Web *crawler* berperilaku sesuai dengan bagaimana kebijakan pencarian diterapkan terhadapnya:

1. Kebijakan seleksi; menentukan page seperti apa yang diunduh
2. Kebijjakan *re-visit;* menentukan kapan mengecek kembali halaman yang sebelumnya telah dilihat
3. Kebijakan *politeness;* dibuat agar web *crawler* tersebut tidak membuat halaman *overload*
4. Kebijakan paralelisasi; menentukan bagaimana beberapa web *crawler* bekerja agar tidak saling tumpang tindih.

Semakin besarnya ukuran *World Wide Web* dan beraneka ragamnya web membuat tantangan dari kebijakan tersebut semakin sulit dan kompleks.

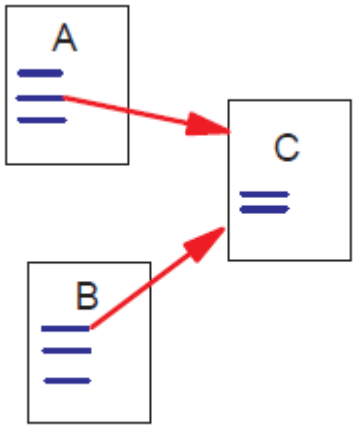


Figure 1Struktur dasar link web. A dan B adalah backlink dari C

I.2 Google PageRank

PageRank adalah sebuah nilai mumerical yang menyatakan seberapa penting sebuah halaman web di internet. Secara sederhana, perhitungan nilai tersebut bertambah bila halaman tersebut muncul sebagai sebuah *hyperlink* di sebuah halaman web lainnya. Semakin besar nilai yang dimiliki, maka semakin penting web tersebut.

PageRank digunakan oleh Google untuk menentukan tingkat kepentingan halaman web. Hal ini penting karena itulah yang akan menjadi faktor penentuan urutan dalam hasil pencarian oleh Google. Memang in bukanlah satu-satunya factor untuk hal tersebut, namun merupakan salah satu yang paling penting.

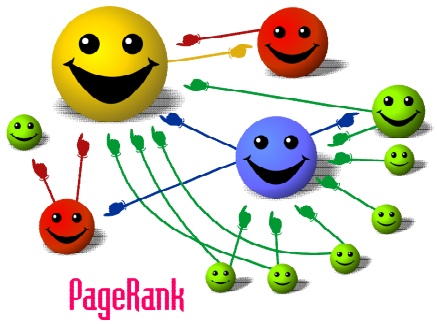


Figure 2Google PageRank

PageRank sendiri merupakan trademark dari Google, dan telah dipatenkan. Namun paten tersebut diberikan bukab pada Google, melainkan Universitas Stanford, Amerika serikat. Google memiliki hak lisensi eksklusif dari Universitas Stanford. Sejak kemunculannya yang pertama dalam paper Larry Page and Sergey Brin di tahun 1998, banyak paper sejenis yang memodifikasi algoritma ini.

Untuk selanjutnya pada paper ini, PageRank hanya akan disebut sebagai PR untuk penyingkatan.

I.3 PR Pada bidang Molekular

Pada awal tahun 2012, seorang Professor Kimia dari *Washington State University*, Aurora Clark beserta beberapa koleganya mengklaim telah membuat moleculaRnetworks, sebuah program yang memakai algoritma PR yang bisa digunakan saintis untuk bentuk dan reaksi kimia molekul tanpa biaya, logistic, dan bahaya dari eksperimen laboratorium. Software yang dibuat berfokus pada ikatan hydrogen pada air, cairan yang paling banyak di bumi sekaligus pelaku penting dalam proses biologi.

Air berfungsi sebagai pembantu protein dalam mengatur dirinya agar tetap dalam kondisi cair. Pada proses-proses yang melibatkan mahkluk hidup, proses yang terhadi sangat rumit dan melibatkan banyak sekali bentuk molekul yang mungkin. Hal inilah yang diibaratkan sebagai web di internet.

Sementara ikatan antar molekul web itulah yang menjadi cabang dalam algoritma PR/web *crawler*.

Temuan ini disambut baik di seluruh dunia, baik dari aspek penelitian kanker, aspek biologi/kimia keseluruhan, maupun di bidang informatika. Hal ini membuktikan betapa luasnya penerapan bidang informatika pada umumnya. Selanjutnya, pada Mei 2012, sebuah paper lagi dikeluarkan untuk menganalisis sejauh mana algoritma PR bisa diterapkan.

Kali ini seorang Christof Winter dari Dresden University of Technology, Glen Kristiansen, Stephan Kersting, dan banyak kolega mereka yang membuat paper sejenis berjudul *"Google Goes Cancer: Improving Outcome Prediction for Cancer Patients by Network-Based Ranking of Marker Gene"*. Sistem buatan mereka bernama NetRank yang mengevaluasi interaksi biologis antar produk gen untuk melihat kemampuan bertahan pasien.

II. ALGORITMA PERHITUNGAN

Semua algoritma yang dipakai di atas berdasarkan dari Breadth-First Search, sebuah algoritma yang menelusuri semua kemungkinan *adjacency* dari sebuah keadaan. BFS biasa dimodelkan dengan bentuk pohon,atau graf dengan sebuah simpul yang memodelkan sebuah keadaan dan cabang yang memodelkan kemungkinan perubahan dari simpul parent-nya.

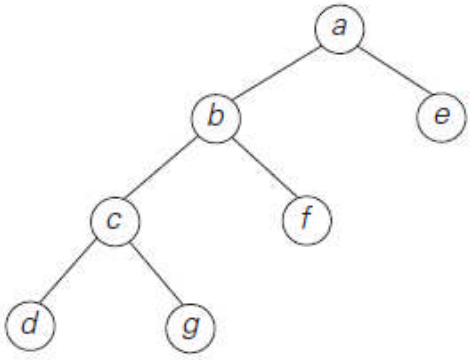


Figure 3 Sebuah contoh pohon

Algoritma BFS secara singkat ditulis dalam 3 poin berikut:

1. Kunjungi sebuah simpul (biasanya simpul akar)
2. Kunjungi semua simpul yang bertetangga dengan simpul yang kita pilih
3. Kunjungi simpul yang belum dikunjungi dan bertetangga dengan simpul yang tadi dikunjungi

Sehingga pergerakan algoritma BFS pada Figure 3 diatas yaitu : a->b->e->c->f->d->g.

II.1 Web *Crawler*

Algoritma Web *Crawler* sangat mudah dibayangkan dengan BFS: sebuah web adalah sebuah simpul, dan sisi simpul menyatakan *link* dari sebuah web ke web lainnya.

Berikut adalah sebuah algoritma *crawler* web, dan sebuah implicit graf untuk memodelkannya.

BFS untuk web *crawler* dijalankan sebagai berikut:

1. Mulai dari sebuah web yang akan dijadikan akar (root) simpul
2. Buat sebuah *queue* websites yang dieksplorasi
3. Buat sebuah *set* untuk web yang telah ditemukan
4. Seiring *crawler* berjalan, *dequeue* web berikutnya dan *enqueue* website yang di-*link*

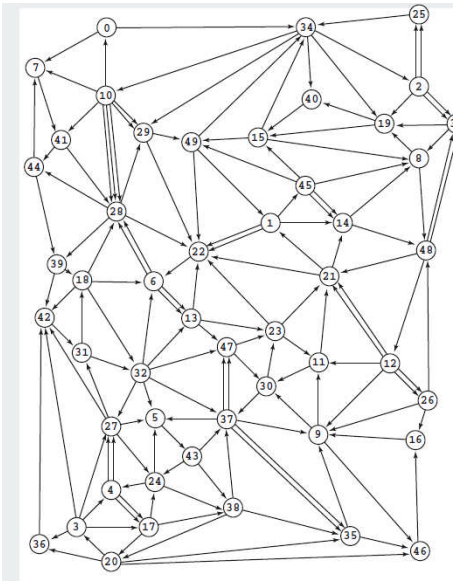


Figure 4 Sebuah Pemodelan web crawler

```
Queue<String> q = new Queue<String>();
SET<String> visited = new SET<String>();
String s = "http://www.itb.ac.id";
q.enqueue(s);
visited.add(s);

while (!q.isEmpty())
{
        String v = q.dequeue();
        System.out.println(v);
        In in new In(v);
        String input = in.readAll();

        String regexp = "http://(\\w+\\.)*(\\w+)";
        Pattern pattern = Pattern.compile(regexp);
        Matcher matcher = pattern.matcher(input);
        while (matcher.find())
       {
                String w = matcher.group();
                if (!visited.contains(w))
                {
                        visited.add(w);
                        q.enqueue(w);
                }
        }
}
```

Implementasi BFS pada web crawler di Java

Pada contoh diatas, root yang dipilih adalah http://www.itb.ac.id . bisa dilihat bahwa bila sebuah *link* web yang terbaca di sebuah web belum terdaftar di set visited, maka web tersebut akan dimasukkan dalam daftar *enqueue*.

Pada implementasi yang lebih lengkap, banyak factor yang perlu dipertimbangkan perihal pergerakan *crawler*, diantaranya:

1. **Halaman yang seperti apa yang harus diunduh?** *Crawler* tidak bisa mengunduh semua page dalam web, untuk itu sangat penting untuk memilih halaman yang "penting" saja, sehingga seluruh halaman yang dikunjungi adalah yang lebih berarti.
2. **Bagaimana cara *crawler* merefresh sebuah halaman?** Halaman web biasanya sering berubah, sehingga bila *crawler* suatu saat sampai di halaman yang sama, sangat penting untuk memutuskan apakah kunjungan ulang itu penting atau tidak
3. **Bagaimana meminimisasi halaman yang dikunjungi?** Membengkaknya jumlah halaman web pada decade-dekade terakhir menjadikan poin ini penting untuk menjaga eksistensi *crawler*.
4. **Bagaimana mem-paralelisasi *web crawler*?** Besarnya web membuat pentingnya melakukan banyak *crawler*an sekaligus, namun akan menambah resiko bila ada web yang terhitung beberapa kali.

II.2 Algoritma PR

Algoritma PR yang dipakai oleh Google dapat dibilang kompleks, mengingat banyaknya halaman web yang di-*cover* Google dan pentingnya pemakaian PR untuk penentuan urutan hasil pencarian.

Pada saat PR pertama kali dipublish, rumus berikut adalah rumus yang dipakai untuk penentuan PR:

PR(A) = (1-d) + d(PR(t1)/C(t1) + ... + PR(tn)/C(tn))

Meski demikian diyakini kalau Google telah melakukan banyak pengembangan dari rumus tersebut.

Pada formula tersebut, t_n adalah halaman yang memiliki *link* ke A, C adalah banyaknya link keluar yang dimiliki halaman tersebut, dan d adalah *damping factor*.

Dapat dilihat rumus PR sebagai berikut:

PR(A) = 0.15+0.85*("share" PR dari semua page)

damping factor biasa diset dengan 0.85.

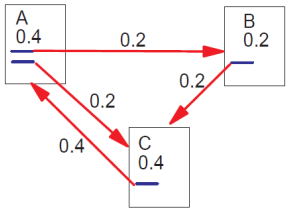
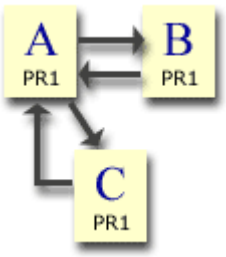


Figure 5 Kalkulasi Sederhana PR

Perhatikan contoh berikut:

Ketiga halaman web berikut ini memiliki PR1 pada awalnya. Kemudian, skema PR dijalankan kembali.



Pada iterasi pertama, perhitungan PR masing-masing adalah:

Halaman A : 1.88
Halaman B: 0.575
Halaman C : 0.575
Dan untuk 100 iterasi, hasilnya adalah:
Halaman A : 1.45
Halaman B : 0.77
Halaman C : 0.77

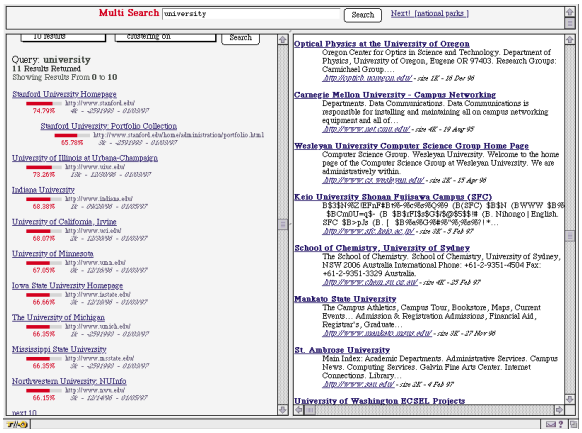


Figure 6 Perbandingan terhadap query "University"

Algoritma PR yang original merefleksikan suatu yang disebut random surfer model, yang berarti PR sebuah halaman ditentukan lewat probabilitas mengunjungi halaman tersebut ketika *link* yang diklik ditentukan secara random. Meski demikian, ada kemungkinan bahwa *surfer* akan mengalami sebuah keaddan dimana web yang dijelajah menjadi loop, sehingga *surfer* akan menjelajah web yang sama sampai selamanya. Biasanya untuk mencegah ini terjadi, ditambahkan sebuah factor lagi sehingga bila *surfer* "bosan", maka akan lompat ke halaman acak

- Mulai dari halaman acak
- Dengan probability 0.85, kunjungi sebuah *Hyperlink* secara acak untuk mengecek halaman berikutnya
- dengan probability 0.15, pilih halaman apapun secara acak
- PR = proporsi waktu yang dihabiskan random surfer pada tiap halaman

Web Page	PageRank (average is 1.0)
Download Netscape Software	11589.00
http://www.w3.org/	10717.70
Welcome to Netscape	8673.51
Point: It's What You're Searching For	7930.92
Web-Counter Home Page	7254.97
The Blue Ribbon Campaign for Online Free Speech	7010.39
CERN Welcome	6562.49
Yahoo!	6561.80
Welcome to Netscape	6203.47
Wusage 4.1: A Usage Statistics System For Web Servers	5963.27
The World Wide Web Consortium (W3C)	5672.21
Lycos, Inc. Home Page	4683.31
Starting Point	4501.98
Welcome to Magellan!	3866.82
Oracle Corporation	3587.63

Figure 7 Hasil PR pada tahun 1996

III. PENELITIAN DALAM RISET KANKER

Pembahasan berikut akan lebih mengarah ke *NetRank*, bukan *moleculaRnetwork*.

Pertanyaan berikut bisa mengantarkan kita pada pengertian bagaimana PR bisa diaplikasikan pada riset kanker: Ada terdapat 20.000 jenis protein yang terlibat dalam pembentukan kanker pancreas; bagaimana cara kita menentukan protein mana yang paling berpengaruh atas munculnya kanker tersebut?

Ketika meneliti kanker, salah satu hal penting untuk diperhatika adalah biomarkers, sebuah substansi kimiawi yang dibuat oleh sel kanker. Idenya adalah jika kita tahu senyawa dari biomarker tersebut, maka kita bisa mencarinya dan menemuka sel-sel yang membuatnya. Hal ini bisa membantu para dokter untuk mencegah penyebaran sebuah kanker/tumor sebelum mereka tumbuh, lebih baik daripada kita mendapatkan seluruh tumor namun terlebih dahulu menunggu sampai dapat dideteksi.

Dengan mengenali sifat protein yang memegang peran besar dalam pembentukan kanker, maka para ilmuwan/dokter dapat menerapkan PR langsung ke dalam metodenya: semakin banyak sebuah protein (biomarker) tersambung dengan macam-macam protein lainnya dalam tubuh, semakin tinggi ranking protein tersebut. Hal ini persis seperti sebuah halaman web dalam perhitungan Google PR.

Setelah kita tahu representasi dari node, kemudian kita harus mencari tahu apa yang akan direpresentasikan oleh sisi-sisi node tersebut. Christof Winter *et al.* menggunakan 3 macam representasi:

1. Hubungan transkripsi factor-target
2. Interaksi protein-protein (antar protein)
3. *Gene co-expression*

Para ilmuwan mengamati ekspresi gen dari 30 pasien yang menjalani operasi dari tahun 1996-2007. Mereka dibagi menjadi 2 grup, yaitu grup dengan prognosa baik bila mereka telah melewati waktu median waktu *survival*, yaitu 17.5 bulan, dan grup dengan prognosa yang tidak baik bila mereka belum mencapai 17.5 bulan. 8000 jenis gen dari mereka kemudian diamati perbedaannya.

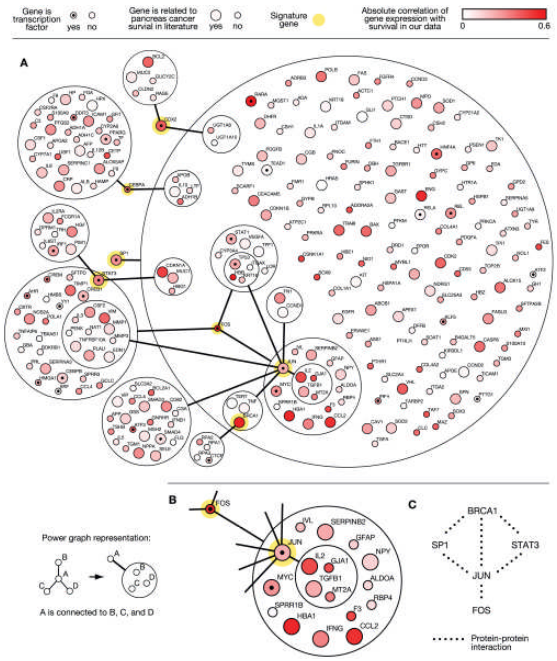


Figure 7Regulatory network around signature genes[1]

Salah satu yang diketahui dari percobaan tersebut yaitu bahwa Hubungan transkripsi factor-target gen adalah yang paling akurat dari ketiga representasi.

Menggunakan *NetRank*,d idapatkan pula tujuh protein yang bisa memprediksi *survival* pada sample mereka. Hal ini juga telah divalidasi dengan analisis gen terhadap 412 pasien tambahan yang juga menjalani operasi kanker pancreas pada durasi 1991-2008. Teknik ini menentukan ekspresi gen pada tiap tumor pada pasien, dan teknik yang mereka gunakan adalah immunohistochemical, yaitu proses pendeteksian antigen pada sel jaringan dengan prinsip dasar bahwa antibody akan terikat secara spesifik pada sebuah jaringan.

IV. KESIMPULAN

Pada bab-bab diatas kita melihat bagaimana algoritma sederhana BFS bila dikembangkan labih jauh untuk tujuan tertentu dapat memcahkan banyak masalah. *Web crawler*, misalnya, dapat diterapkan untuk mengawasi traffic web. Google PR yang dilindungi paten memiliki banyak potensi dan mungkin akan menjadi factor terpenting dalam pencarian query pada mesin pencari apapun. Lebih jauh, penelitian riset kanker menggunakan algoritma PR merupakan terobsan yang masih sangat baru, sehingga belum matang dan memiliki banyak kelemahan. Namun hal tersebut member contoh bahwa lingkup kreativitas dalam berkarya dan membuat terobosan masih sangat besar, termasuk dalam ilmu informatika.

REFERENCES

[1] http://www.ploscompbiol.org/article/info%3Adoi%2F10.1371%2Fjournal.pcbi.1002511&imageURI=info%3Adoi%2F10.1371%2Fjournal.pcbi.1002511.g003, tanggal akses 21 Desember 2012 pkl. 06.00

[2] Castillo, Carlos (2004). *Effective Web Crawling* (Ph.D. thesis). University of Chile. Retrieved 2010-08-03..

[3] Winter, Christof Glen Kristiansen, Stephan Kersting, et al. *Google Goes Cancer: Improving Outcome Prediction for Cancer Patients by Network-Based Ranking of Marker Genes (2012)*

[4] Clark, Aurora and Barbara Logan Mooney dan L. Rene Corrales. *MoleculaRnetworks: An integrated graph theoretic and data mining tool to explore solvent organization in molecular simulation*. (2012)

[5] http://www.biotechniques.com/news/biotechniquesNews/biotechniques-330743.html?utm_source=BioTechniques+Newsletters+%2526+e-Alerts&utm_campaign=617f8b1c2f-Daily&utm_medium=email, tanggal akses 21 Desember 2012 pkl. 03.30..

[6] http://www.google.com/patents/US6285999, tanggal akses 21 Desember 2012 pkl. 03.20.

[7] http://www.webworkshop.net/pagerank.html, waktu akses 21 Desember 2012 pkl. 03.00

[8] Page, Lawrence dan Sergey Brin dan Rajeev Motwani dan Terry Winograd *The PageRank Citation Ranking: Bringing Order to the Web*(1999)..

[9] Cho, Junghoo. Crawling The Web: Discovery and Maintenance of Large Scale Web Data. Dissertation 2003.

[10] Ramos-Vara, JA. "Technical Aspects of Immunohistochemistry" (2005)

[11] www.cs.princeton.edu/~rs/AlgsDS07/13**DirectedGraphs**.pdf, tanggal akses 21 Desember 2012 pkl 05.00

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 29 April 2010

Yulius Nainggolan
13510090