

# Penerapan Teori Graf Dalam Pembuatan *Webgraph*

Kevin Yauris/13513036

Program Studi Teknik Informatik

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia

13513036@std.stei.itb.ac.id

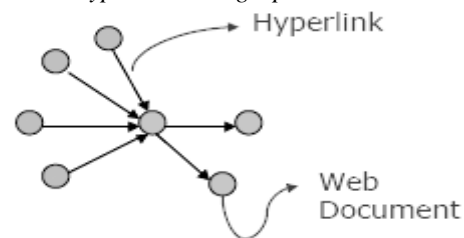
*Abstrak-Di dunia sekarang ini telah tersedia banyak sekali informasi, sebagian besar informasi disimpan dalam bentuk data. Data perlu disimpan dalam suatu memori, dan kadangkala data-data yang berisi informasi yang sama perlu disimpan dalam memori yang berbeda-beda karena diperlukan oleh beberapa orang atau untuk beberapa keperluan yang berbeda, sedangkan kita tahu memori merupakan sesuatu yang bersifat terbatas. Dalam suatu wadah yang besar seperti internet, data dalam berbagai bentuk tersedia dan dapat diakses dari mana saja, padahal data tersebut tersimpan hanya dalam satu memori. Hal ini dapat dilakukan karena ada suatu fasilitas yang bernama hyperlink, yaitu suatu referensi data yang apabila diklik oleh pengguna akan mengantarkan pengguna ke data yang ditautkan. Hal ini sangat berguna karena data tidak perlu disimpan berkali-kali, cukup satu kali saja. Ternyata pembuatan hyperlink ini menggunakan teori graf. Dalam tulisan ini, penulis akan menitikberatkan kepada bagaimana kaitan teori graf dengan hyperlink, dan akan dibahas juga mengenai apa yang disebut webgraph dan bagaimana cara kerja dan bagaimana webgraph mengaplikasikan teori-teori graf.*

**Keywords**—graf, hyperlink, webgraph, struktur data, crawler.

## I. PENDAHULUAN

Dalam suatu struktur data, penggunaan memori yang seminimal mungkin merupakan suatu keharusan, oleh karena itu penyimpanan data yang berisi informasi yang sama pada memori yang berbeda-beda perlu dihindari. Penggunaan memori yang seminimal mungkin menjadi perhatian karena akan berpengaruh pada berbagai hal yang berhubungan dengan struktur data tersebut, contohnya adalah penggunaan algoritma pencarian dalam struktur data, apabila penggunaan memori kecil maka kompleksitas waktu maksimum dari program pencarian pun akan kecil. Akan tetapi, memang kadangkala suatu data diperlukan oleh orang yang berbeda-beda dan dengan keperluan yang berbeda-beda sehingga harus ada cara untuk membuat banyak orang dapat mengakses data yang sama tanpa perlu menyimpan data itu berulang-ulang kali. Hal inilah yang melatar belakangi terciptanya *hyperlink*. Dalam sebuah graf terdapat simpul dan sisi. Sederhananya didalam jaringan internet sebuah halaman, situs atau data merupakan simpul sedangkan *hyperlink* merupakan sisinya. Dengan begitu setiap halaman dapat saling berkaitan satu sama lain lewat *hyperlink* sehingga data yang disimpan dalam suatu halaman pun bisa diminimiliasi. Hal ini sangat menguntungkan karena selain menghemat penggunaan memori, *search engine* yang pun menjadi lebih mudah mengalokasikan sebuah data dan mengirimkan alamatnya ke pengguna. Hal tersebut yang menyebabkan terciptanya nama *hyperlink*, karena dengan *hyperlink*, penggunaan dapat berpindah dari suatu

halaman ke halaman yang lain dengan sangat cepat (*hyperspeed*). Suatu struktur yang menampilkan kumpulan halaman-halaman (*pages*) serta koneksinya dengan halaman lain lewat *hyperlink* dikenal dengan nama *hyperlink graph* atau *webgraph*..



## Web Graph Structure

Gambar 1: contoh sederhana *webgraph*

Diambil dari <http://www.expertsupdates.com/seo-articles/web-mining-12.aspx>

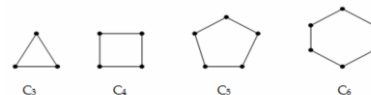
## II. LANDASAN TEORI

### A. Teori Graf

Graf adalah himpunan suatu data atau benda yang saling berhubungan. Benda, data atau objek dalam suatu graf digambarkan sebagai simpul (*vertex*). Sedangkan hubungan antara simpul disebut busur atau sisi (*edge*).

Secara matematis sebuah graf dapat ditulis sebagai berikut  $G = (V, E)$ , dengan  $V$  adalah himpunan tidak kosong dari simpul (*vertices*) =  $\{V_1, V_2, \dots, V_n\}$  dan  $E$  adalah himpunan sisi (*edges*) yang menghubungkan dua simpul tertentu =  $\{e_1, e_2, \dots, e_n\}$ .

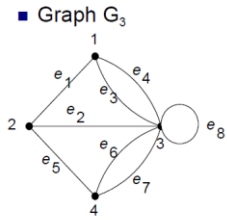
Terdapat beberapa jenis graf, tetapi terbagi atas dua bagian besar berdasarkan ada tidaknya sisi ganda atau sisi gelang, yaitu graf sederhana (tidak mengandung sisi ganda dan sisi gelang) dan graf tak sederhana (memiliki sisi ganda atau sisi gelang). Sisi ganda adalah adanya lebih dari satu sisi yang menghubungkan dua simpul, dan sisi gelang adalah sebuah sisi yang terhubung pada satu simpul (disebut cincin karena biasanya berbentuk cincin).



### Gambar 2: Graf sederhana

Diambil dari [krisna.students-blog.undip.ac.id](http://krisna.students-blog.undip.ac.id)

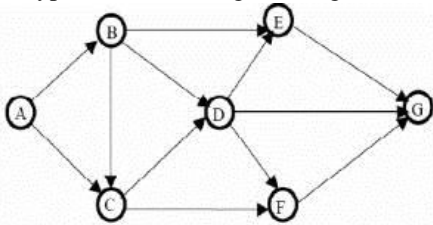
# Graph



Pada  $G_3$ , sisi  $e_8 = (3, 3)$  dinamakan **gelang** atau **kalang (loop)** karena ia berawal dan berakhir pada simpul yang sama.

**Gambar 3 : Graf tak sederhana**  
diambil dari [darkrabbitblog.blogspot.com](http://darkrabbitblog.blogspot.com)

Sedangkan berdasarkan orientasinya graf terdiri atas 2 yaitu graf berarah dan graf tidak berarah. Dalam hyperlink sendiri digunakan graf berarah.



Gambar 2.6 Graf ABCDEFG

Gambar 4 Graf berarah  
Diambil dari [t4urusboy08.blogspot.com](http://t4urusboy08.blogspot.com)

Beberapa terminology graf antara lain :

### 1. Ketetanggaan (adjacent)

Dua buah simpul pada graf dikatakan bertetangga apabila keduanya terhubung langsung oleh sebuah sisi. Pada graf  $G_1$  dalam Gambar 1, simpul D bertetangga dengan simpul C, E, dan F, namun tidak bertetangga dengan simpul A dan B.

### 2. Bersisian (Incidency)

Untuk sembarang sisi  $e = (V_j, V_k)$  dikatakan bersisian dengan simpul  $V_j$ , atau  $e$  bersisian dengan simpul  $V_k$ . Pada graf  $G_1$  dalam Gambar 1, sisi (A,B) bersisian dengan simpul A dan simpul B, sisi (B,C) bersisian dengan simpul B dan simpul C, tapi sisi (A,B) tidak bersisian dengan simpul E.

### 3. Derajat (Degree)

Derajat suatu simpul adalah jumlah sisi yang bersisian dengan simpul tersebut. Notasi :  $d(v)$ . Pada graf  $G_1$  dalam Gambar 1, dapat dilihat bahwa

$$d(A) = 1,$$

$$d(B) = d(D) = d(F) = 3,$$

$$d(C) = d(E) = 2.$$

### 4. Lintasan (Path)

Lintasan (path) yang panjangnya  $n$  dari simpul awal  $V_0$  ke simpul tujuan  $V_n$  di dalam graf  $G$  ialah barisan berselang-seling simpul-simpul dan sisi-sisi yang berbentuk  $V_0, e_1, V_1, V_2, \dots, V_{n-1}, e_n, V_n$  sehingga  $e_1$

$(V_0, V_1)$ ,  $e_n (V_{n-1}, V_n)$  adalah sisi-sisi dari graf  $G$ . Panjang lintasan adalah jumlah sisi dalam lintasan tersebut. Lintasan A-B-C-D-E-F-G pada graf  $G_1$  dalam Gambar 4 mempunyai panjang lintasan 7.

### 5. Siklus (cycle)

Siklus (cycle) merupakan lintasan yang berawal dan berakhir pada simpul yang sama. Misalnya pada graf  $G_1$  dalam Gambar 4, B-C-D-E-B adalah sebuah sirkuit. Panjang sirkuit adalah jumlah sisi dalam sirkuit tersebut. Sirkuit B-C-D-E-B pada  $G_1$  memiliki panjang 4.

### 6. Terhubung (Connected)

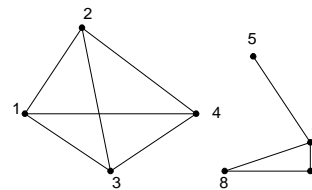
Dua buah simpul  $V_1$  dan  $V_2$  disebut terhubung jika terdapat lintasan (sisi) di antara kedua simpul tersebut. Graf  $G$  disebut terhubung jika tiap simpul mempunyai pasangan yang terhubung oleh satu atau lebih sisi. Jika ada simpul yang terpisah dan tidak terhubung, maka graf itu disebut graf tak terhubung.

### 7. Graf Berbobot (Weighted Graph)

Graf berbobot adalah graf yang sisinya mempunyai nilai, yaitu panjang sisi. Misalnya pada suatu peta, panjang sisi akan melambangkan jarak tempuhnya. *Hyperlink graph* tergolong dalam graf berbobot, tetapi nilai dari sisinya itu merupakan sebuah alamat untuk mengantarkan pengguna ke suatu halaman atau data tertentu.

### 8. Terhubung

Dua buah simpul dikatakan terhubung apabila terdapat paling tidak satu lintasan dari  $v_1$  ke  $v_2$ . Suatu graf disebut graf terhubung (*connected graph*) jika untuk setiap pasang simpul sembarang pada graf tersebut terhubung, jika tidak maka graf tersebut disebut graf tak-terhubung (*disconnected graph*).



Gambar 5. Graf 1234 merupakan graf terhubung graf 5678 merupakan graf tak-terhubung.

Untuk graf berarah, suatu graf berarah dikatakan terhubung apabila graf tak-berarah dari graf tersebut merupakan graf terhubung. Suatu graf berarah dikatakan terhubung kuat apabila terdapat lintasan berarah dari  $u$  ke  $v$  dan sebaliknya, untuk sembarang node  $u$  dan  $v$ , jika graf berarah tidak memiliki sifat tersebut tetapi graf tak-berarahnya merupakan graf terhubung maka graf tersebut dikatakan terhubung lemah.

### 9. Upagraf

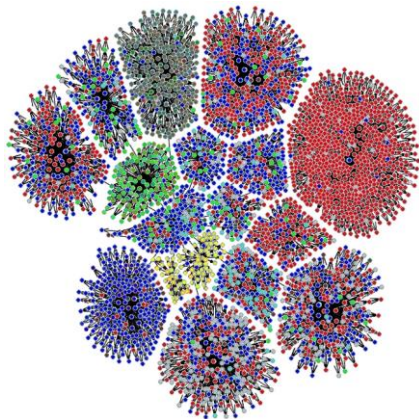
Misalkan  $G = (V, E)$  adalah sebuah graf.  $G_1 = (V_1, E_1)$  adalah upagraf (*subgraph*) dari  $G$  jika  $V_1 \subseteq V$  dan  $E_1 \subseteq E$ . Komplemen dari upagraf  $G_1$  terhadap graf  $G$

adalah graf  $G_2 = (V_2, E_2)$  sedemikian sehingga  $E_2 = E - E_1$  dan  $V_2$  adalah himpunan simpul yang anggota-anggota  $E_2$  bersisian dengannya.

## B. Teori Pewarnaan graf

Dalam pembuatan *hyperlink graph* kadangkala membutuhkan pewarnaan pada graf, hal ini karena akan ada banyak sekali halaman(simpul dari graf) yang perlu dihubungkan. Oleh karena itu agar mempermudah pengguna mencari dan mempermudah penggunaannya untuk kegiatan lain, *hyperlink graph* diwarnai dengan beberapa warna, warna ini diberikan kepada simpul sesuai dengan isi dari data atau halaman tersebut, sehingga apabila dilakukan pencarian khusus terhadap data yang mempunyai karakteristik isi tertentu maka akan menjadi lebih cepat dibandingkan dengan mencari pada setiap simpul secara keseluruhan.

Terdapat dua macam pewarnaan graf, yaitu pewarnaan simpul dan pewarnaan sisi. Pada pewarnaan simpul, yang diberi warna adalah simpul-simpul graf sehingga dua simpul yang bertetangga mempunyai warna yang berbeda. Suatu pewarnaan sisi  $k$  untuk graf  $G$  adalah suatu penggunaan sebagian atau semua  $k$  warna untuk mewarnai semua sisi di  $G$  sehingga setiap pasang sisi yang mempunyai titik persekutuan diberi warna yang berbeda.



Gambar 6. Contoh pewarnaan *Hyperlink graph* yang kompleks Diambil dari adsri.anu.edu.au

## C. Hyperlink

*Hyperlink* adalah referensi menuju ke suatu data yang dapat digunakan pengguna untuk menuju ke data tersebut, dengan suatu cara tertentu. *Hyperlink* terdiri atas dua jenis, yaitu *inline links* dan *anchor*. *Inline link* adalah jenis *hyperlink* yang bersifat otomatis, dimana pengguna tidak perlu melakukan sesuatu agar isi data atau suatu konten itu keluar atau muncul pada halaman. Contohnya adalah iklan-iklan pada *web*. Sedangkan *anchor* merupakan jenis *hyperlink* yang akan mengeluarkan isi data atau memunculkan informasi hanya jika pengguna melakukan sesuatu(biasanya mengklik atau memindahkan *cursor mouse* ke tempat *hyperlink*).

## III. HUBUNGAN ANTARA HYPERLINK DAN GRAPH

### A. Web Structure Mining

*Hyperlink graph* merupakan termasuk dalam cakupan *web structure mining*. *Web structure mining* memiliki tujuan untuk menghasilkan kesimpulan terstruktur tentang sebuah *websites* atau *web page*. Secara teknis, *web structure mining* berusaha untuk menemukan struktur hubungan dari kumpulan *hyperlink* pada level bagian dalam dokumen. Informasi terstruktur yang dihasilkan oleh *web structure mining* adalah sebagai berikut :

- 1) Informasi mengenai banyaknya penggunaan *local link* di dalam *web*.
- 2) Informasi mengenai banyaknya penggunaan *link* terhadap data yang berada pada dokumen yang sama dan bersifat lokal.
- 3) Informasi mengenai banyaknya penggunaan *link* terhadap data yang berada pada dokumen yang sama dan bersifat global.
- 4) Informasi mengenai banyaknya penggunaan *web tuple* dan *web table* yang relatif sama.

### B. Struktur dari Webgraph

Struktur dari *webgraph* terdiri atas halaman *web* sebagai simpul, dan *hyperlink* sebagai sisi dari graf yang menghubungkan dua halaman yang saling berhubungan. Saat terjadi pencarian informasi, pencarian konvensional akan berfokus pada teks dari dokumen *web* sementara informasi-informasi tambahan akan ditambahkan lewat *hyperlink* yang tersedia pada halaman yang tersedia.

Beberapa terminologi *Web Structure*

#### 1. Webgraph

Merupakan graf berarah yang merepresentasikan *web*.

#### 2. Simpul

Setiap halaman *web* merupakan simpul pada *webgraph*.

#### 3. Sisi

Setiap sisi pada *webgraph* merupakan *hyperlink* yang terdapat pada *web* dan merupakan penghubung antara dua halaman *web*.

#### 4. Derajat dalam

Derajat dalam dari sebuah simpul adalah jumlah *link* yang menunjuk untuk simpul tersebut.

#### 5. Derajat luar

Derajat luar dari sebuah simpul adalah jumlah *link* yang menunjuk pada simpul lain yang berasal dari simpul itu sendiri.

#### 6. Directed Path

Kumpulan *link* yang dimulai dari halaman awal yang bisa diikuti untuk menuju ke halaman lainnya.

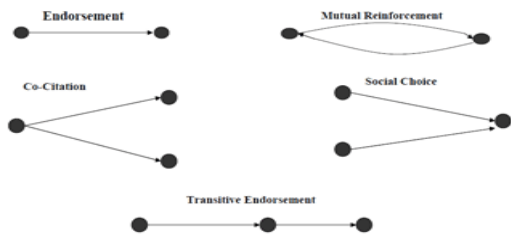
#### 7. Shortest Path

Kumpulan *link* yang paling sedikit yang diperlukan untuk menuju kepada halaman tertentu dari halaman awal.

#### 8. Diameter

Maksimum dari semua *shortest path* antara pasangan simpul  $p$  dan  $q$ , untuk semua pasangan simpul  $p$  dan  $q$  dalam graf.





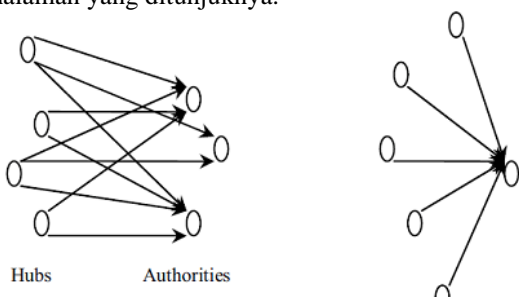
Gambar 7. Bentuk-bentuk *webgraph*  
Diambil dari [expertsupdates.com/seo-articles](http://expertsupdates.com/seo-articles)

### C. Ranking berbasis konektivitas (*Connectivity-Based Ranking*)

Sesuai dengan namanya, ranking berbasis konektivitas adalah suatu bentuk pengurutan(perengkingan) yang dilakukan terhadap halaman *web* berdasarkan jumlah konektivitas yang dimilikinya.

Ada dua tipe ranking berbasis konektivitas yang pertama adalah *query-independent*, yang melakukan pengurutan hanya dengan melihat konektivitas halaman/dokumen saja. Dan yang Kedua adalah *query-dependent*, yang mampu membuat irisan atau upagraf dari *webgraph* yang dapat disusun kembali dan menghasilkan hasil pencarian yang lebih berarti.

*Hyperlink-induced topic search (HITS)* merupakan algoritma iteratif untuk menggali *webgraph* agar dapat menemukan topik *hubs*(halaman dengan sumber konten yang bagus) dan *authorities* (halaman dengan peringkat tinggi untuk topik yang diberikan). *Hubs* adalah halaman yang mempunyai *link* dengan *authorities*. Algoritma diterima sebagai masukan hasil pencarian oleh teknik pemberi indeks teks tradisional, dan menyaring hasil ini untuk mengidentifikasi *hubs* dan *authorities*. Nomor dan berat dari *hubs* yang menunjuk kepada halaman menjadi faktor penentu *authorities* dari sebuah halaman. Algoritma memberi berat terhadap *hubs* berdasarkan *authorities* dari halaman yang ditunjuknya.



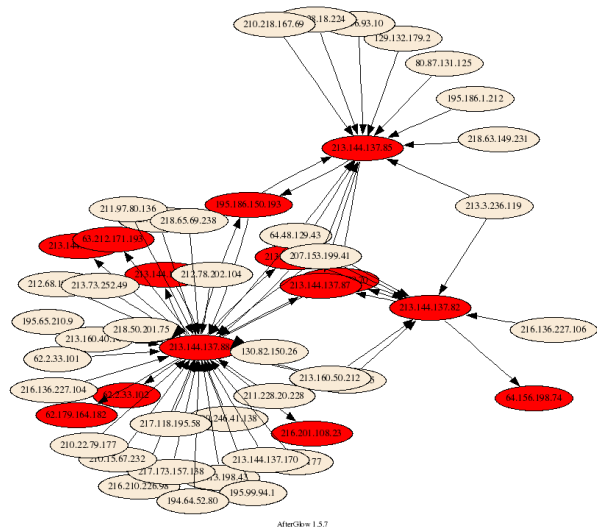
Gambar 8. Contoh bentuk *webgraph* dengan *hubs* dan *authorities*  
Diambil dari [www.expertsupdates.com/seo-articles](http://www.expertsupdates.com/seo-articles)

### D. Link Graph

Ada beberapa jenis graf yang ada dalam *web* yang sangat berguna bagi *search engine* untuk memproses dan mengevaluasi relevansi dan reputasi dari suatu halaman dalam indeks mereka. Beberapa graf yang digunakan dalam *web* saat ini :

### 1. Link graph

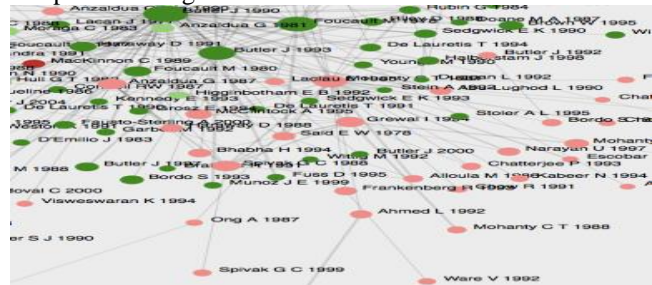
Simpul merupakan representasi dari halaman atau dokumen.



Gambar 9. Contoh *link graph* berisi alamat  
Diambil dari [raffy.ch](http://raffy.ch)

### 2. Co-citation graph

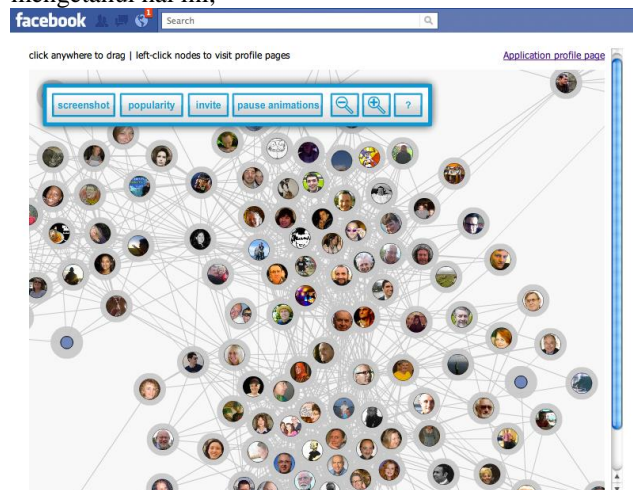
Simpul dihubungkan oleh sisi tak berarah.



Gambar 10. Contoh *co-citation graph*  
Diambil dari [signsat40.signsjournal.org/cocitation](http://signsat40.signsjournal.org/cocitation)

### 3. Social graph

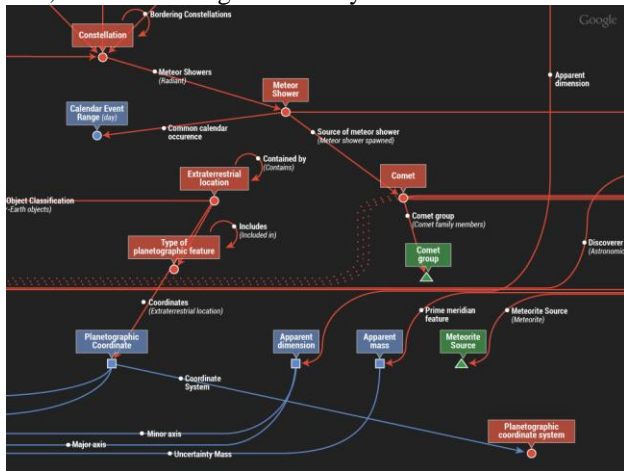
Koneksi implisit dan eksplisit antara individu dan *web*. Graf ini biasanya dipakai oleh media sosial untuk mencari tahu hubungan antara para penggunanya, dengan mengetahui hal ini,



Gambar 11. Contoh *social graph*  
Diambil dari [www.stephendale.com](http://www.stephendale.com)

#### 4. Knowledge graph

Kumpulan entitas kanonik yang tidak berhubungan dan atribut mereka. Kumpulan yang dibuat bisa berupa individu, tempat, organisasi, tim olahraga, suatu karya seni, film dan berbagai hal lainnya.



Gambar 12. Contoh *knowledge graph*  
Diambil dari mashable.com

#### IV. MANFAAT PENGGUNAAN TEORI GRAF DALAM PEMBUATAN SUATU WEBGRAPH

*Webgraph* sendiri digunakan untuk berbagai keperluan salah satunya adalah untuk melakukan pengurutan prioritas hasil pencarian seperti yang dipaparkan. Karena menggunakan teori graf, maka hal ini dapat dilakukan dengan mudah karena tinggal melihat koneksi antara halaman. Selain itu apabila kita menggunakan mesin pencari seperti google, kita juga dapat mendapatkan saran pencarian lebih lanjut, ini juga merupakan aplikasi dari teori graf, hal ini dilakukan dengan melihat hubungan hal yang kita cari dengan hal lainnya dengan cara melihat halaman yang terhubung lewat *hyperlink* dalam *webgraph*. Telah dijelaskan juga mengenai algoritma HITS yang dapat mengidentifikasi *hubs* dan *authorities*, hal ini juga dapat dilakukan karena ada dengan teori dan struktru graf yang diterapkan dalam pembuatan *webgraph*, akan terlihat seberapa banyak suatu halaman ditunjuk oleh halaman lain lewat *hyperlink*. Hal lainnya yang sangat menguntungkan dari penggunaan teori graf untuk merepresentasikan hubungan dari halaman-halaman atau dokumen-dokumen yang ada dalam web adalah kemudahan untuk menambahkan suatu *hyperlink* dari suatu halaman atau dokumen baru ke dalam halaman atau dokumen yang sebelumnya sudah ada dalam graf. Karena menggunakan graf maka hanya perlu menambahkan simpul kemudian sebuah sisi baru yang menunjuk ke halaman atau dokumen yang diinginkan, hal ini belum tentu dapat dilakaukan apabila hubungan antara dokumen-dokumen direpresentasikan dengan bentuk lain. Dengan struktur graf juga, konten suatu halaman yang perlu digunakan pada halaman-halaman lain bisa disimpan dalam sebuah halaman lain (simpul baru) kemudian halaman-halaman yang membutuhkan konten

tersebut tinggal memanggil konten dengan menggunakan *hyperlink*. Selain hemat memori, hal ini juga membuat sebuah halaman lebih mudah diakses karena tidak perlu pengunduhan data yang banyak. Hal ini juga dapat membuat pembuat halaman lebih mudah mencari kesalahan (error) yang terjadi pada halaman, hal ini karena apabila ada suatu konten dalam suatu halaman dan konten tersebut disimpan dalam suatu halaman tersendiri maka pembuat tinggal memperbaiki halaman yang rusak saja tanpa melakukan pencarian error yang lebih dalam dan kadangkala sulit dilakukan karena banyaknya data yang disimpan dalam sebuah halaman. Perlu dingat juga mengenai kemampuan untuk menampilkan informasi umum saja pada halaman utama, sedangkan informasi khusus disimpan dalam halaman lain yang dapat diakses apabila pengguna menekan *hyperlink* yang menghubungkan kedua halaman tersebut, hal ini menguntungkan karena kadang kala pengguna hanya memerlukan informasi menyeluruh atau umum mengenai suatu hal, dan pemberian terlalu banyak informasi khusus malah dapat membuat pengguna bingung karena terlalu banyak informasi yang disediakan.

Kelemahan dari struktur *webgraph* ini adalah apabila suatu halaman terlalu banyak mempunyai *hyperlink* otomatis diakses apabila halaman tersebut dibuka, hal ini membuat halaman akan lama dibuka karena harus mengakses halaman-halaman yang berisi konten untuk halaman tersebut. Kelemahan lainnya adalah apabila terlalu banyak halaman atau dokumen yang menunjuk pada satu halaman atau dokumen tertentu, hal ini membuat halaman tersebut dapat mengalami *overload* apabila diakses secara sekaligus oleh beberapa halaman. Kemudahan menambahkan hubungan antara halaman juga dapat menjadi pedang bermata dua, karena dengan kemudahan seperti ini, para peretas dapat dengan muda menaruh sebuah *hyperlink* pada sebuah halaman yang ramai akan mengunjung yang akan mengantarkan pengunjung ke halaman yang bersifat mencuri data pengguna atau melakukan peretasan pada komputer pengguna. Hal ini juga kadang dimanfaatkan pihak-pihak tertentu untuk memasang iklan-iklan tanpa izin dari pemilik halaman, biasa berupa *popout adds*. Oleh karena itu para pengakses *web* mesti berhati-hati dan melakukan pencegahan. Pencegahan dapat dilakukan dengan beberapa cara, antara lain dengan cara mengatur *setting* browser agar memblok *pop adds* dan juga menggunakan anti virus yang dapat melindungi dari peretasan pada komputer, hal ini perlu dilakukan agar pengguna terhindar dari berbagai kerugian akibat pihak-pihak tak bertanggung jawab yang berusaha memanfaatkan kemudahan akses antara halaman yang tersedia pada *web*.

#### V. KESIMPULAN

Berdasarkan penelitian yang dilakukan, maka penulis mengambil kesimpulan bahwa teori graf merupakan teori yang sangat penting dalam pembuatan *web*. Hal ini membuat *web* memiliki suatu struktur yang sangat mudah diakses dan menghemat banyak sekali

memori karena tidak perlu menyimpan hal yang sama berulang-ulang kali. Hal ini juga menguntungkan karena *webgraph* dapat digunakan untuk berbagai tujuan lain, akan tetapi bisa juga menjadi hal yang merugikan. Oleh karena itu pengguna *web* perlu waspada akan ancaman-ancaman yang ada pada saat mengakses *web*.

## VI. UCAPAN TERIMA KASIH

Pertama-tama penulis ingin bersyukur kepada Tuhan Yang Maha Esa, karena hanya oleh karena rahmat-Nya penulis dapat menyelesaikan tulisan ini. Penulis juga berterima kasih kepada dosen yang memberikan tugas ini Dr. Ir. Rinaldi Munir atas bimbingan dan jasa beliau yang selama ini telah mengajar dan memberikan ilmu bagi penulis, sehingga penulis mampu membuat tulisan ini. Tak lupa juga penulis berterima kasih atas rekan-rekan yang senantiasa memberikan dorongan dan semangat bagi penulis.

## REFERENCES

- [1] Barabási, Albert-László; Albert, Réka (October 1999). "Emergence of scaling in random networks". *Science* 286 (5439): 509–512. doi:10.1126/science.286.5439.509. PMID 10521342..
- [2] <http://whatis.techtarget.com/definition/Google-Knowledge-Graph>, 7 Desember 2014
- [3] Munir, Rinaldi, "Matematika Diskrit", Informatika, Bandung: 2010
- [4] <http://dejanseo.com.au/search-quality-the-link-graph-theory>, 7 Desember 2014
- [5] <http://adsri.anu.edu.au/graduate-study/ssi/network1>, 7 Desember 2014
- [6] Ravi Kumar, Prabhakar Raghavan, Sridhar Rajagopalan, Andrew Tomkins, Trawling the Web for emerging cyber-communities, *Computer Networks*, Volume 31, Issues 11–16, 17 May 1999, Pages 1481–1493, ISSN 1389–1286, doi:10.1016/S1389-1286(99)00040-7.
- [7] S. Brin, L. Page, *Computer Networks and ISDN Systems* 30, 107 (1998)

## 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, 7 Desember 2014



Kevin Yauris / 13513036