

Aplikasi Graf dalam Algoritma PageRank untuk Search Engine

Hobert Anthony Jonatan – 13521079¹

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

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

¹13521079@std.stei.itb.ac.id

Abstract—Terdapat banyak sekali *website* yang ada di internet saat ini. Banyak *website* yang memiliki kesamaan topik atau bahasan dalam menyediakan informasi, hal ini menyebabkan mesin pencari atau *search engine* harus melakukan penyaringan atau proses *ranking* dari *website* agar dapat menampilkan *website* yang paling sesuai dengan apa yang dicari oleh pengguna internet. Salah satu algoritma yang digunakan untuk melakukan proses tersebut adalah PageRank yang dikembangkan untuk Search Engine Google. Landasan utama dari algoritma PageRank adalah graf untuk memodelkan hubungan antar *website* di dalam internet.

Keywords—*website*, internet, *search engine*, *ranking*, PageRank.

I. PENDAHULUAN

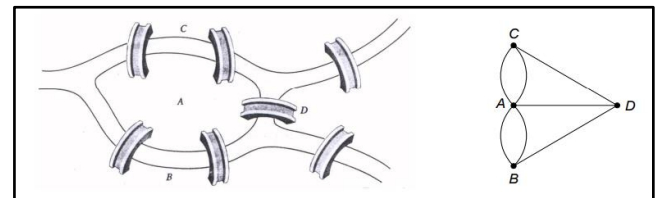
Di era digital ini, kita hidup di dalam dunia yang serba terhubung melalui sistem yang kita kenal dengan nama internet. Adanya internet membuat kita dapat mengakses berbagai informasi yang tersedia di dalamnya cukup dengan memasukkan beberapa kata pencarian pada *search engine* dan informasi yang kita cari akan ditampilkan pada layar. Informasi di internet banyak dimuat dalam *website* yang disimpan pada *web server*. Terdapat milyaran *website* yang telah ada di internet yang menyimpan berbagai informasi, dan tidak jarang beberapa *website* akan menyimpan informasi yang mengandung topik atau bahasan yang sama dengan *website* lainnya. Oleh karena itu, *search engine* atau mesin pencari perlu dirancang sedemikian rupa sehingga mampu menampilkan hasil pencarian (*website*) yang paling relevan dengan apa yang dicari oleh pengguna berdasarkan kata kunci yang dimasukkan pengguna.

II. DASAR TEORI

A. GRAF

1. Pengertian Graf

Graf adalah kumpulan titik yang dihubungkan oleh sisi untuk merepresentasikan objek-objek diskrit dan hubungan antara objek-objek tersebut. Konsep graf pertama kali diperkenalkan dan digunakan oleh seorang matematikawan Swiss, Leonhard Euler, untuk menyelesaikan persoalan Jembatan Königsberg pada tahun 1736.



Gambar 2.1. Kiri : persoalan Jembatan Königsberg; Kanan : representasi graf persoalan

Sumber :

<https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf>

Persoalan tersebut berbunyi "bisakah orang melalui setiap jembatan tepat sekali dan kembali lagi ke tempat semula?". Dari pemecahan masalah ini, kemudian dikembangkan definisi graf secara matematika sebagai berikut,

$Graf G = (V, E)$, yang dalam hal ini :

V merupakan himpunan tidak-kosong dari simpul-simpul (*vertices*). $V = \{v_1, v_2, \dots, v_n\}$

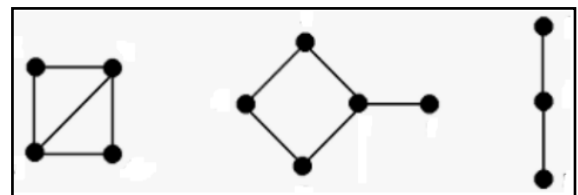
E merupakan himpunan sisi (*edges*) yang menghubungkan sepasang simpul. $E = \{e_1, e_2, \dots, e_n\}$

2. Jenis-jenis Graf

Graf dapat digolongkan berdasarkan beberapa kategori. Berdasarkan ada tidaknya gelang atau sisi ganda pada suatu graf, maka graf digolongkan menjadi dua jenis, yaitu:

1. Graf sederhana (*simple graph*)

Graf sederhana adalah graf yang tidak mengandung gelang maupun sisi ganda.



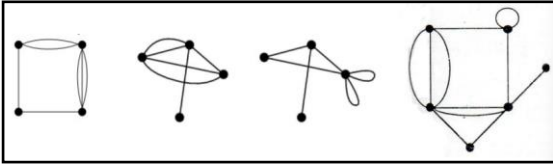
Gambar 2.2. Contoh Graf sederhana

Sumber :

<https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf>

2. Graf tak-sederhana (*unsimple-graph*)

Graf tak-sederhana adalah graf yang mengandung sisi ganda atau gelang.



Gambar 2.3. Contoh Graf tak-sederhana

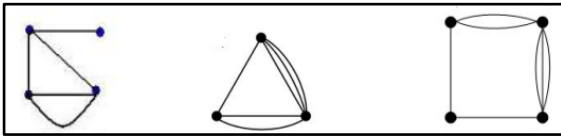
Sumber :

<https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf>

Graf tak-sederhana dibedakan lagi menjadi:

1. Graf ganda (*multi-graph*)

Graf ganda adalah graf yang mengandung sisi ganda.



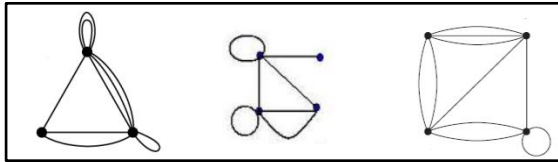
Gambar 2.4. Contoh graf ganda

Sumber :

<https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf>

2. Graf semu (*pseudo-graph*)

Graf semu adalah graf yang mengandung sisi gelang



Gambar 2.5. Contoh graf semu

Sumber :

<https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf>

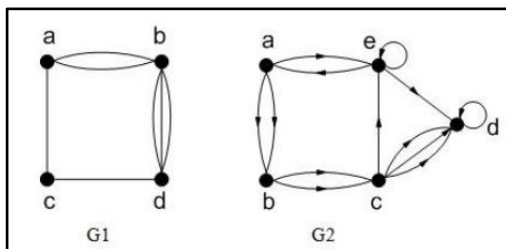
Berdasarkan orientasi arah pada sisi, graf dibedakan atas 2 jenis:

1. Graf tak-berarah (*undirected graph*)

Graf tak-berarah adalah graf yang sisinya tidak mempunyai orientasi arah.

2. Graf berarah (*directed graph* atau *digraph*)

Graf berarah adalah graf yang setiap sisinya diberikan orientasi arah.



Gambar 2.6. G1 : Graf tak-berarah; G2 : Graf berarah

Sumber :

<https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf>

3. Terminologi Graf

Terdapat beberapa terminologi atau istilah ketika kita berbicara mengenai graf, berikut adalah beberapa terminologi graf yang disebutkan dalam makalah ini.

1. Ketetanggaan (*Adjacent*)

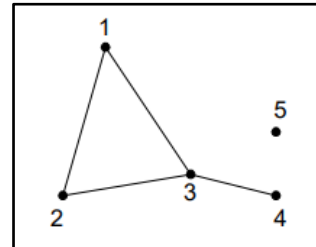
Dua buah simpul dikatakan bertetangga bila keduanya terhubung langsung. Tinjau graf G1 pada gambar 2.6, dapat dilihat bahwa simpul a bertetangga dengan simpul b dan c, tetapi simpul a tidak bertetangga dengan simpul d.

2. Bersisian (*Incidency*)

Untuk sembarang sisi $e = (v_j, v_k)$ dikatakan e bersisian dengan simpul v_j , atau e bersisian dengan simpul v_k . Tinjau graf G1 pada gambar 2.6, dapat dilihat bahwa sisi(a,c) bersisian dengan simpul a dan c, sisi(c,d) bersisian dengan simpul c dan d, tetapi sisi(a,c) tidak bersisian dengan simpul d.

3. Simpul Terpencil (*Isolated Vertex*)

Simpul terpencil adalah simpul yang tidak mempunyai sisi yang bersisian dengannya.



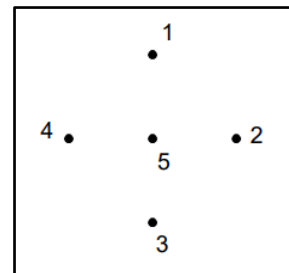
Gambar 2.7. Simpul 5 adalah simpul terpencil

Sumber :

<https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf>

4. Graf Kosong (*null graph* atau *empty graph*)

Graf kosong adalah graf yang sisinya merupakan himpunan kosong.



Gambar 2.8. Graf kosong

Sumber :

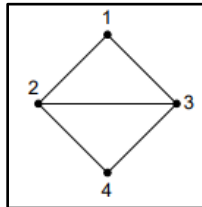
<https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf>

5. Derajat (*Degree*)

Derajat suatu simpul adalah jumlah sisi yang bersisian dengan simpul tersebut. Notasi untuk derajat adalah $d(v)$. Pada gambar 2.7 dapat dilihat bahwa $d(1) = d(2) = 2$, $d(3) = 3$, $d(4) = 1$, dan $d(5) = 0$.

6. Lintasan (*Path*)

Lintasan 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 terbentuk $v_0, e_1, v_1, e_2, v_2, \dots, v_{n-1}, e_n, v_n$ sedemikian sehingga $e_1 = (v_0, v_1), e_2 = (v_1, v_2), \dots, e_n = (v_{n-1}, v_n)$ adalah sisi-sisi dari graf G .



Gambar 2.9. Graf sederhana

Sumber :

<https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf>

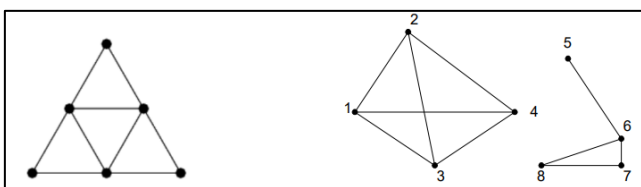
Pada gambar 2.9, lintasan 1, 2, 4, 3 adalah lintasan dengan barisan sisi (1,2), (2,4), (4,3). Panjang lintasan adalah jumlah sisi dalam lintasan tersebut. Lintasan 1, 2, 4, 3 pada gambar 2.9 memiliki panjang 3.

7. Siklus (*Cycle*) atau Sirkuit (*Circuit*)

Lintasan yang berawal dan berakhir pada simpul yang sama disebut sirkuit atau siklus. Pada gambar 2.9, jalur 1, 2, 3, 1 adalah sebuah sirkuit. Panjang sirkuit adalah jumlah sisi dalam sirkuit tersebut. Sirkuit 1, 2, 3, 1 pada gambar 2.9 memiliki panjang 3.

8. Keterhubungan (*Connected*)

Dua buah simpul v_1 dan v_2 disebut terhubung jika terdapat lintasan dari v_1 ke v_2 . G disebut graf terhubung (*connected graph*) jika untuk setiap pasang simpul v_i dan v_j dalam himpunan V terdapat lintasan dari v_i ke v_j . Jika tidak, maka G disebut graf tak-terhubung.



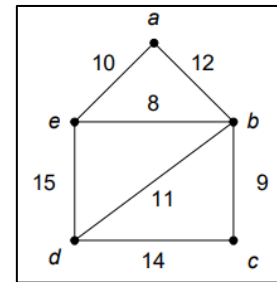
Gambar 2.10. Kiri : contoh graf terhubung; Kanan : contoh graf tak-terhubung

Sumber :

<https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf>

9. Graf Berbobot (*Weighted Graph*)

Graf berbobot adalah graf yang setiap sisinya diberi sebuah harga (bobot).



Gambar 2.11. Contoh graf berbobot

Sumber :

<https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf>

B. WEBSITE

Website adalah kumpulan dari banyak *webpage*, dimana *webpage* sendiri adalah *file digital* yang ditulis menggunakan HTML (HyperText Markup Language). Agar sebuah *website* dapat tersedia untuk diakses seluruh orang di dunia melalui internet, *website* tersebut harus disimpan atau *dihosting* di computer yang terhubung ke internet sepanjang waktu, atau computer tersebut lebih dikenal dengan nama *web server*. *Webpage* dari *website* dihubungkan melalui *hyperlinks* dan *hypertext*, dan mereka menggunakan antarmuka dan desain yang serupa. *Website* juga dapat berisi beberapa dokumen atau informasi dalam bentuk selain *text*, misalnya seperti gambar, video, atau asset digital lainnya. Untuk membangun suatu *website* diperlukan berbagai komponen, berikut adalah komponen utama yang umumnya menjadi penyusun setiap *website*.

1. *Webhost*

Hosting adalah lokasi tempat *website* berada secara fisik. Sekumpulan *webpage* dilisensikan sehingga dapat disebut *website* hanya jika *web hosting* di sebuah *web server*. *Web server* adalah sekumpulan *file* yang ditransmisikan ke computer pengguna ketika pengguna mencari *address website* yang spesifik.

2. *Address*

Address atau alamat *website* sering juga disebut sebagai URL dari sebuah *website*. Ketika seorang pengguna mau mengakses sebuah *website* secara langsung, pengguna perlu memasukkan alamat atau URL *website* tersebut ke dalam *web browser* dan *website* tersebut kemudian akan diberikan kepada pengguna oleh *web server*.

3. *Homepage*

Homepage adalah halaman *web* pertama yang muncul ketika pengunjung mengunjungi *website*. *Homepage* sangat penting karena akan mengatur tampilan dan nuansa dari *website* serta mengarahkan pengunjung ke halaman lainnya di *website* tersebut.

4. *Design*

Design berarti tampilan dan nuansa akhir dari sebuah

website yang melingkupi integrasi dari elemen-elemen yang terdapat dalam *website*, seperti menu navigasi, grafik, *layout website*, yang bertujuan untuk mencapai penggunaan yang tepat dari *website*.

5. Content

Content adalah isi utama dari *website*, yaitu informasi atau layanan apa yang terkandung dalam *website* tersebut.

6. Navigation Structure

Navigation Structure website adalah urutan halaman *web*, kumpulan tautan dari halaman yang saling menghubungkan atau dapat disebut juga sebagai integrasi antar *webpage* dalam sebuah *website*.

C. SEARCH ENGINE

Search Engine adalah sistem yang memungkinkan pengguna untuk mencari informasi di internet dengan hanya memasukkan kata kunci atau frasa yang relevan. Adanya *search engine* sangat diperlukan karena di internet terdapat banyak sekali *website* yang memuat informasi dan banyak di antaranya juga memuat informasi dengan topik yang sama. Untuk itu diperlukan sebuah sistem untuk dapat melakukan penyaringan *website* sehingga dapat memberikan hasil *website* yang memuat informasi yang paling relevan dengan apa yang dicari oleh pengguna. Secara umum, *search engine* (Google) memiliki cara kerja dengan beberapa tahapan sebagai berikut.

1. Crawling

Crawling adalah proses yang dilakukan oleh Search Engine Web Crawlers (biasanya berupa *bot* atau *spiders*) untuk mengunjungi *webpage* yang ada pada internet dan kemudian mengambil informasi penting atau *keyword* yang ada dalam *webpage* tersebut sebagai inti dari *webpage* tersebut, *keyword* bersama dengan alamat *webpage* kemudian disimpan dalam index, yaitu suatu tempat penyimpanan yang sangat masif untuk menyimpan *keyword* beserta alamat *webpage*. Setelah selesai mengekstrak informasi dari suatu *webpage*, *crawler* kemudian bergerak ke *webpage* lain berdasarkan *link* yang didapatkan dari *webpage* sebelumnya, peristiwa ini dapat dimodelkan dalam graf, dengan *webpage* sebagai simpul (*vertex*) dan *link* sebagai sisi (*edge*).

2. Indexing

Indexing adalah proses yang dilakukan oleh *crawler* untuk menyimpan informasi penting atau *keyword* yang telah ditemukannya dari proses *crawling* ke dalam sebuah tempat penyimpanan data yang sangat besar yang menyimpan *keyword* dan alamat dari *webpage* yang memuat *keyword* tersebut. Indexing berfungsi untuk mempercepat *search engine* nantinya saat proses *Ranking website* yang akan ditampilkan kepada pengguna saat pengguna mencari sesuatu.

3. Ranking

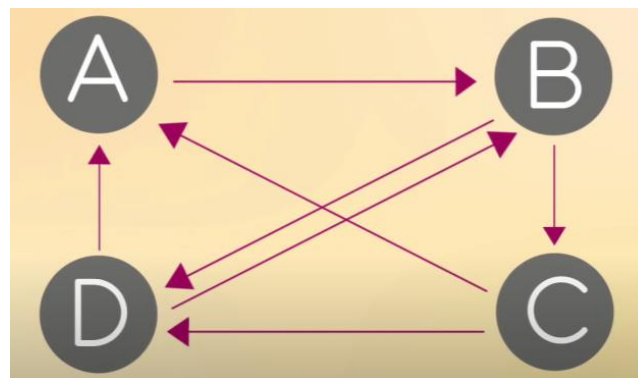
Ranking adalah proses yang dilakukan oleh *search engine* untuk mengurutkan hasil *website* yang sesuai

dengan pencarian yang dilakukan oleh pengguna. Semakin tinggi *rank* atau posisi dari sebuah *website* maka semakin relevan pula informasi yang terdapat di dalamnya dengan *keyword* yang dicari oleh pengguna. Terdapat berbagai algoritma untuk melakukan *Ranking website*, setiap *search engine* memiliki algoritmanya masing-masing untuk melakukan *Ranking*. Salah satu algoritma yang menarik dan akan dibahas lebih dalam di bab berikutnya adalah algoritma PageRank yang digunakan oleh Search Engine Google. PageRank adalah algoritma yang diciptakan oleh Larry Page dan Sergey Brin, *founder* dari Google, sebagai dasar algoritma utama Google untuk melakukan *Ranking website* yang akan ditampilkan kepada user yang melakukan pencarian. Penjelasan lebih lanjut mengenai algoritma PageRank akan dijelaskan dalam bab berikut.

III. ANALISIS

PageRank adalah algoritma yang dikembangkan oleh Google untuk menentukan relevansi suatu halaman web terhadap kata kunci tertentu. Secara umum, algoritma ini bekerja dengan mempertimbangkan jumlah dan kualitas *link* yang ada pada suatu halaman *web*. Halaman *web* yang memiliki banyak *link* berkualitas dari halaman *web* lain yang terpercaya akan mendapatkan nilai PageRank yang lebih tinggi, sehingga lebih mudah untuk ditemukan oleh *search engine* dalam suatu pencarian. Asumsi yang membuat algoritma ini bekerja cukup baik adalah bahwa sebuah *website* yang dianggap penting cenderung menerima *link* dari *website* lain yang mengarah kepadanya.

Dalam pengaplikasian algoritma PageRank, setiap *website* dan keterhubungan antar *website* dimodelkan dalam sebuah graf berarah, dengan simpul bermakna sebagai tiap *website* itu sendiri dan sisi yang berarah bermakna sebagai *link* dari satu *website* menuju ke *website* lainnya. Untuk penjelasan lebih lanjut dari permodelan graf yang dimaksud disini, tinjau gambar berikut.



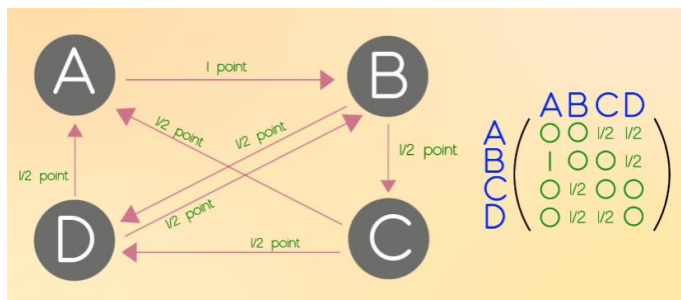
Gambar 3.1. Contoh sederhana permodelan *website* dalam graf

Sumber :

<https://www.youtube.com/watch?v=qxEkY8OScYY>

Dalam graf berarah di atas, setiap simpul, A, B, C, dan D merepresentasikan sebuah *website*, misalnya *website* A, B, C,

dan D. Setiap sisi dari suatu simpul menuju ke simpul lain menunjukkan adanya link dari suatu *website* menuju ke *website* lain, maka pada gambar di atas, sebagai contoh, dapat dikatakan bahwa *website* A memiliki *link* ke *website* B, dan *website* B memiliki *link* ke *website* C dan D, tetapi *website* B tidak memiliki *link* ke *website* A. Setelah *website* dan keterhubungannya dimodelkan dalam graf, setiap *website* dalam graf tersebut kemudian diberi nilai atau point awal sebesar 1 untuk masing-masing *website*. Hubungan *link* pada *website* kemudian akan menyatakan pemberian point bagi *website* lain yang ditunjuk atau dilink oleh *website* penunjuk. *Website* yang memiliki link ke *website* lainnya akan membagikan seluruh pointnya secara merata ke tiap *website* yang ditunjuk sampai point dari *website* itu sendiri habis, tetapi dalam iterasi yang sama pula *website* akan menerima point dari *website* lain yang memiliki link ke dirinya. Penggambaran pertukaran point dalam sistem *website* ini dapat digambarkan dengan graf berbobot serta matriks ketetanggaan sebagai berikut.



Gambar 3.2. Graf berbobot untuk merepresentasikan persoalan pertukaran point antar *website*

Sumber :

<https://www.youtube.com/watch?v=qxEkY8OScYY>

Pada gambar 3.2, dapat dilihat bahwa setiap simpul (*website*) akan memberikan seluruh pointnya secara merata kepada *website* lain yang ditunjuk olehnya. Misalnya *website* B, memberikan $\frac{1}{2}$ point kepada D dan juga $\frac{1}{2}$ point kepada C, ditotal menjadi sebesar 1 point adalah jumlah point yang diberikan oleh *website* B. Selanjutnya, matriks ketetanggaan pada gambar 3.2 juga merepresentasikan pemberian point dari tiap *website* ke *website* lainnya, dengan tanda bahwa kolom adalah pemberi point dan baris adalah penerima dari point tersebut. Contohnya, pada baris A, kolom C, terisi oleh angka $\frac{1}{2}$, hal ini berarti C memberikan $\frac{1}{2}$ point kepada A. Terdapat informasi lain yang dapat diperoleh dari matriks ketetanggaan ini, yaitu jumlah tiap baris menyatakan berapa point yang diperoleh oleh suatu *website* pada suatu iterasi. Misal pada baris A, jumlah elemen pada baris A adalah jumlah point yang didapat oleh *website* A dalam iterasi tersebut. Jika diteliti dengan lebih cermat, sesungguhnya point merepresentasikan kemungkinan kita berpindah ke *website* lain dari *website* yang sedang kita diami sekarang. Selanjutnya, *ranking* dari *website* dilakukan dengan menjawab pertanyaan berikut “jika dihabiskan waktu yang lama (dapat dianggap mendekati tak hingga) untuk mengklik atau berpindah-pindah antar *website* dalam jaringan tersebut, berapa persentase waktu yang dihabiskan di setiap *website*?”. Persentase total waktu akan berjumlah 100% dan

persentase waktu yang dihabiskan di tiap *website* akan menjadi rank dari tersebut dan menentukan ranking dari websitenya. Proses perhitungan rank *website* ini akan dilakukan sampai mendekati iterasi tak hingga dan itu adalah hasil akhir dari persentase waktu yang akan kita habiskan dalam tiap *website* jika kita menghabiskan cukup banyak waktu untuk berpindah-pindah secara acak dalam jaringan *website* yang ada. Kembali lagi pada contoh jaringan *website* A, B, C, D sebelumnya, maka pada mula-mula kita memiliki peluang 25% untuk menghabiskan waktu di tiap *website* (rank 0.25 untuk tiap *website*), kemudian untuk mencari rank tiap *website* pada perpindahan pertama, maka matriks ketetanggaan yang telah kita peroleh sebelumnya harus dikalikan dengan matriks rank hasil dari iterasi sebelumnya. Iterasi pertama dari perhitungan rank dalam contoh kasus kita dilakukan dalam perhitungan perkalian matriks sebagai berikut.

$$\begin{bmatrix} 0 & 0 & 1/2 & 1/2 \\ 1 & 0 & 0 & 1/2 \\ 0 & 1/2 & 0 & 0 \\ 0 & 1/2 & 1/2 & 0 \end{bmatrix} \times \begin{bmatrix} 0.25 \\ 0.25 \\ 0.25 \\ 0.25 \end{bmatrix} = \begin{bmatrix} 0.25 \\ 0.375 \\ 0.125 \\ 0.25 \end{bmatrix}$$

Hasil dari perkalian matriks tersebut kemudian menjadi rank yang baru untuk dapat dipakai dalam iterasi berikutnya. Selanjutnya kita akan melakukan iterasi terus menerus sampai mendekati tak hingga untuk memperoleh rank yang sebenarnya dari setiap *website*. Maka kita dapat merumuskannya menjadi:

$$\lim_{n \rightarrow \infty} (A)^n R_0$$

A : matriks ketetanggaan *website*

R₀ : matriks rank awal *website*

Dari contoh kasus yang kita bahas, didapatkan nilai akhir rank dari tiap *website* dalam jaringan, yaitu sebagai berikut :

Gambar 3.3 Hasil rank *website* dalam jaringan contoh

Sumber :

<https://www.youtube.com/watch?v=qxEkY8OScYY>

Nilai rank yang semakin besar menunjukkan bahwa *website* tersebut dinilai lebih penting dibanding *website* lain pada jaringan tersebut. Dengan nilai rank yang semakin besar, maka suatu *website* akan menempati posisi yang lebih tinggi ketika ditampilkan dalam hasil pencarian oleh *search engine*. Algoritma PageRank yang dibahas dalam makalah ini hanyalah algoritma PageRank sederhana yang diterapkan paling awal oleh Google, sekarang algoritma ini telah dikembangkan lebih jauh lagi untuk memberikan hasil pencarian yang lebih baik dan lebih relevan lagi dengan napa yang dicari oleh pengguna.

IV. KESIMPULAN

Pencarian informasi pada *website* di internet menjadi hal yang sangat sering dilakukan oleh manusia di era digital ini. Terdapat banyak sekali *website* yang ada di internet sehingga diperlukan adanya suatu algoritma atau cara penyaringan untuk dapat mencari dan menampilkan *website* yang mengandung informasi berkualitas kepada pengguna internet. Salah satu algoritma yang digunakan adalah PageRank yang memanfaatkan teori graf dalam penerapannya. Pada algoritma PageRank, setiap *website* yang ada dipetakan dalam suatu graf dengan simpul menyatakan *website* tersebut dan sisi berarah menyatakan hubungan *link* antar *website*. Kemudian diterapkan perhitungan untuk menilai seberapa penting atau relevan (*rank*) suatu *website* relatif terhadap *website* lainnya. Dengan didapatnya *rank* tiap *website*, maka hasil pencarian pada *Search Engine* dapat diurutkan sesuai dengan *rank* dari tiap *website* tersebut. PageRank merupakan algoritma tahap awal dalam *sorting website* pada *Search Engine* Google, pengaplikasian algoritma *sorting website* akhir-akhir ini tentu saja sudah mengalami banyak perubahan, tetapi PageRank telah menjadi landasan untuk pengembangan algoritma *sorting website* berdasarkan nilai pentingnya.

Bandung, 12 Desember 2022



Hobert Anthony Jonatan 13521079

V. UCAPAN TERIMA KASIH

Puji syukur kepada Tuhan Yang Maha Esa karena atas berkat dan rahmat-Nya, penulis dapat menyelesaikan Tugas Makalah Matematika Diskrit ini dengan lancar. Tidak lupa, penulis juga ingin mengucapkan terima kasih kepada Dr. Nur Ulva Maulidevi, S.T., M.Sc. selaku dosen pengampu mata kuliah IF2120 Matematika Diskrit Kelas 01, yang telah banyak memberikan ilmu kepada penulis selama mengikuti kelas Matematika Diskrit. Penulis juga ingin mengucapkan terima kasih kepada Dr. Ir. Rinaldi Munir, M.T. atas slide kuliah yang sangat lengkap dan telah menjadi salah satu referensi utama penulis untuk menyelesaikan makalah ini. Terakhir, penulis juga ingin mengucapkan terima kasih kepada keluarga, terkhususnya orang tua yang sangat mendukung kegiatan perkuliahan penulis, serta penulis juga berterima kasih kepada semua orang yang membaca makalah ini, harapannya tulisan dalam makalah ini dapat bermanfaat untuk berbagi informasi dengan orang banyak.

REFERENSI

- [1] Munir, Rinaldi. 2020. "Graf (Bag. 1)". <https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf>, diakses pada 12 Desember 2022.
- [2] <https://www.geeksforgeeks.org/page-rank-algorithm-implementation/>, diakses pada 12 Desember 2022.
- [3] <https://www.youtube.com/watch?v=qxEkY8OScYY>, diakses pada 12 Desember 2022.
- [4] <https://www.geeksforgeeks.org/what-is-a-website/>, diakses pada 12 Desember 2022.
- [5] <http://ilpubs.stanford.edu:8090/422/1/1999-66.pdf>, diakses pada 12 Desember 2022.

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.