

Aplikasi Aljabar Vektor dalam Algoritma Page Rank

Albertus Kelvin / 13514100
Program Studi Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
13514100@std.stei.itb.ac.id

Abstract—Pernahkah Anda membayangkan apa yang terjadi setelah Anda mengetikkan suatu query pada sebuah mesin pencari? Bagaimana mesin pencari memberikan daftar halaman situs yang relevan dengan keinginan Anda? Semua itu ternyata terletak pada algoritma yang digunakan dalam menentukan relevansi suatu situs sekaligus indeks kepentingan dari informasi yang terkandung di dalamnya. Salah satu algoritma yang terkenal adalah algoritma Page Rank yang digunakan oleh mesin pencari Google. Algoritma ini menggunakan aljabar vektor sebagai representasi indeks kepentingan dan relevansi sebuah situs.

Keywords—algoritma, aljabar vektor, mesin pencari, page rank

I. PENDAHULUAN

Ada 3 tahap utama yang dilakukan oleh sebuah mesin pencari, pertama mesin pencari menerima input berupa query dari user. Kedua, mesin pencari melakukan proses pencarian terhadap semua halaman situs yang mengandung kata-kata dari query user tersebut. Ketiga, mesin pencari mengurutkan halaman situs berdasarkan relevansi nya.

Berdasarkan tahapan di atas, pada tahap ke-3 mesin pencari sudah mendapat semua situs yang relevan terhadap kata pencarian. Langkah selanjutnya adalah bagaimana mengurutkan semua situs tersebut agar situs yang memiliki nilai relevansi tertinggi berada pada posisi teratas dan situs dengan nilai relevansi terendah berada pada posisi terbawah. Sebenarnya algoritma umum yang digunakan adalah dengan mengurutkan situs dari posisi teratas ke terbawah dengan kondisi situs yang memiliki jumlah kata query terbanyak pada posisi teratas. Namun, jika algoritma tersebut yang digunakan, maka nilai relevansi tidak akan didapat. Situs yang memiliki jumlah kata query terbanyak belum tentu menjamin kalau situs tersebut memiliki informasi yang penting dan relevan.

Oleh karena itu, dalam makalah ini akan dibahas sebuah algoritma yang digunakan oleh Google bernama Page Rank, dimana algoritma ini menjelaskan bagaimana mesin pencari Google memberikan indeks / ranking kepada suatu situs.

II. TEORI DASAR

A. Teori Algoritma

Algoritma adalah suatu metode / langkah kerja yang digunakan untuk menyelesaikan suatu pekerjaan.

Algoritma juga menjadi salah satu pondasi penting dalam bidang pemrograman di samping struktur data. Dalam proses pembuatan program, sebenarnya programmer menuliskan satu atau lebih algoritma yang jika digabungkan akan membuat program tersebut berjalan.

Contoh algoritma dalam bahasa C sebagai berikut.

```
[1] a=1;
[2] while (a <= 3) {
[3]   printf("Hello\n");
[4]   a++;
[5] }
```

Algoritma di atas dapat dijelaskan sebagai berikut: Pertama nilai variabel a diinisialisasi dengan 1 (baris [1]). Lalu, pada body "while" program akan mencetak string "Hello" dengan new line jika nilai variabel a kurang dari atau sama dengan 3 (baris [3]). Nilai variabel a akan ditambah 1 setelah pencetakan string (baris [4]), sehingga jika nilai a lebih dari 3, maka proses pencetakan string berakhir.

B. Teori Aljabar Vektor

B.1. Komponen vektor

Suatu vektor dibentuk oleh 2 titik, dimana titik pertama disebut sebagai titik pangkal dan titik kedua disebut sebagai titik ujung. Jika titik pangkal vektor adalah titik utama (0,0), maka komponen vektor adalah (a1, a2), dimana a1 dan a2 adalah koordinat dari titik ujungnya. Jika titik pangkal vektor di koordinat (p, q), maka komponennya adalah (a1-p, a2-q). Untuk kasus titik pangkal di (0,0), maka nilai p = q = 0.

B.2. Notasi vektor

Sebuah vektor yang memiliki komponen v1 dan v2 dapat dituliskan dalam salah satu cara berikut.

$$\vec{v} = \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}$$

$$\vec{v} = (v_1, v_2)$$

$$\vec{v} = v_1\vec{i} + v_2\vec{j}$$

Gambar B1. Notasi Vektor

Sumber:

<http://uyuhan.com/matematika/vektor/rangkuman-vektor-aljabar.php>

B.3. Panjang vektor

Suatu vektor merupakan sebuah ruas garis yang memiliki panjang. Panjang vektor ditentukan sebagai berikut (dengan v_1 , v_2 , dan v_3 adalah komponen vektor di ruang berdimensi 2 dan 3).

$$\vec{v} = \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} \Rightarrow |\vec{v}| = \sqrt{v_1^2 + v_2^2}$$

$$\vec{v} = \begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix} \Rightarrow |\vec{v}| = \sqrt{v_1^2 + v_2^2 + v_3^2}$$

Gambar B2. Panjang vektor

Sumber:

<http://uyuhan.com/matematika/vektor/rangkuman-vektor-aljabar.php>

B.4. Vektor satuan

Vektor satuan merupakan vektor yang memiliki panjang 1. Vektor satuan dari suatu vektor v didefinisikan sebagai vektor v tersebut dibagi dengan panjang vektor v .

Pada gambar B1 dapat dilihat suatu vektor dapat memiliki notasi $v = v_1\vec{i} + v_2\vec{j}$, dimana \vec{i} dan \vec{j} adalah vektor satuan juga. Komponen vektor \vec{i} adalah (1,0) dan komponen vektor \vec{j} adalah (0, 1).

B.5. Penjumlahan/Pengurangan vektor

Suatu vektor dapat dijumlahkan/dikurangkan dengan vektor lain dengan cara melakukan operasi yang sesuai terhadap komponen yang bersesuaian antar vektor tersebut.

$$\text{Jika } \vec{u} = \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \text{ dan } \vec{v} = \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}$$

$$\text{maka } \vec{u} \pm \vec{v} = \begin{pmatrix} u_1 \pm v_1 \\ u_2 \pm v_2 \end{pmatrix}$$

Gambar B3. Penjumlahan dan pengurangan vektor

Sumber:

<http://uyuhan.com/matematika/vektor/rangkuman-vektor-aljabar.php>

B.6. Perkalian vektor

Perkalian antar vektor dapat berupa perkalian sebuah vektor dengan konstanta atau perkalian sebuah vektor dengan vektor lain yang menghasilkan sebuah nilai skalar atau vektor lain.

Untuk kasus perkalian vektor dengan konstanta, yang dilakukan adalah mengalikan semua komponen vektor tersebut dengan konstanta. Vektor yang baru memiliki komponen vektor yang sudah dikalikan dengan konstanta tersebut.

Untuk kasus perkalian vektor dengan vektor lain yang menghasilkan nilai skalar, ini dinamakan sebagai *dot product* (perkalian titik).

$$\text{Misal } \vec{u} = u_1\vec{i} + u_2\vec{j} + u_3\vec{k} \text{ dan } \vec{v} = v_1\vec{i} + v_2\vec{j} + v_3\vec{k}$$

$$\vec{u} \cdot \vec{v} = u_1v_1 + u_2v_2 + u_3v_3$$

atau

$$\vec{u} \cdot \vec{v} = |\vec{u}||\vec{v}| \cos \theta$$

dengan θ : sudut antara dua vektor.

Gambar B4. Perkalian titik vektor

Sumber:

<http://uyuhan.com/matematika/vektor/rangkuman-vektor-aljabar.php>

Untuk kasus perkalian vektor dengan vektor lain yang menghasilkan vektor baru, ini dinamakan sebagai *cross product* (perkalian silang).

Misalkan $a = p\vec{i} + q\vec{j} + r\vec{k}$ dan $b = x\vec{i} + y\vec{j} + z\vec{k}$. Perkalian silangnya dapat dilakukan dengan cara berikut.

\vec{i}	\vec{j}	\vec{k}	\vec{i}	\vec{j}
p	q	r	p	q
x	y	z	x	y

Mencari determinan dari bagian yang diberi garis.

Determinan 1 : $py - xq \rightarrow$ vektor satuannya \vec{k}

Determinan 2 : $qz - yr \rightarrow$ vektor satuannya \vec{i}

Determinan 3 : $rx - zp \rightarrow$ vektor satuannya \vec{j}

Maka, vektor baru yang dihasilkan adalah:

$$(py-xq)\vec{k} + (qz-yr)\vec{i} + (rx-zp)\vec{j}$$

Rumus lain yang berhubungan dengan perkalian silang adalah: $a \times b = |a||b| \sin \theta$ dengan θ sudut antara vektor a dan b .

B.7. Proyeksi vektor

Untuk 2 buah vektor yang tidak saling berhimpit, dapat dicari proyeksi sebuah vektor pada vektor lainnya. Proyeksi tersebut dapat dibedakan menjadi proyeksi skalar dan proyeksi ortogonal.

Proyeksi skalar memberikan hasil berupa nilai panjang dari vektor proyeksi, sedangkan proyeksi ortogonal memberikan hasil berupa vektor proyeksi tersebut.

Berikut adalah rumus untuk menentukan panjang vektor proyeksi.

Misalkan a dan b adalah vektor dan c adalah vektor proyeksi dari a ke b . Panjang vektor proyeksi c adalah:

$$|c| = \frac{\vec{a} \cdot \vec{b}}{|\vec{b}|}$$

Gambar B5. Proyeksi skalar

Sumber:

<http://uyuhan.com/matematika/vektor/rangkuman-vektor-aljabar.php>

Berikut adalah rumus untuk menentukan vektor proyeksi.

Misalkan a dan b adalah vektor dan c adalah vektor proyeksi dari a ke b . Vektor c ditentukan sebagai berikut:

$$\vec{c} = \frac{\vec{a} \cdot \vec{b}}{|\vec{b}|^2} \vec{b}$$

Gambar B6. Proyeksi ortogonal

Sumber:

<http://uyuhan.com/matematika/vektor/rangkuman-vektor-aljabar.php>

B.8. Vektor bebas linier dan tak bebas linier

Himpunan vektor-vektor $\{v_1, v_2, \dots, v_n\}$ dikatakan bebas linier jika persamaan $a_1v_1 + a_2v_2 + \dots + a_nv_n = 0$ mengakibatkan solusi trivial $a_1 = a_2 = \dots = a_n = 0$.

Himpunan vektor-vektor $\{v_1, v_2, \dots, v_n\}$ dikatakan tak bebas linier jika terdapat skalar α_i ($\alpha_i = 1, 2, \dots, n$) dimana α_i tidak sama dengan 0, sehingga persamaan $a_1v_1 + a_2v_2 + \dots + a_nv_n = 0$.

B.9. Kombinasi linier

Sebuah vektor x dikatakan kombinasi linier dari vektor-vektor a_1, a_2, \dots, a_n jika vektor tersebut dapat dinyatakan dalam bentuk $x = k_1a_1 + k_2a_2 + \dots + k_na_n$ dimana k_1, k_2, \dots, k_n adalah skalar.

C. Teori Page Rank

Algoritma Page Rank adalah salah satu algoritma yang paling sesuai untuk menentukan relevansi sebuah situs website. Algoritma ini ditemukan oleh Larry Page dan Sergey Brin saat mereka lulus dari universitas.

Prinsip dasar dari algoritma ini adalah bahwa relevansi atau kepentingan informasi suatu website dapat dilihat dari kumpulan website lain yang mengarah atau memiliki link ke website tersebut. Hal ini menunjukkan bahwa semakin banyak suatu website yang memiliki *backlink* dari website lain, maka dapat dikatakan kalau website tersebut memiliki relevansi yang besar dengan informasi yang terkandung pada website pemberi *backlink* tersebut.

Algoritma Page Rank ini memberikan penilaian / ranking pada suatu website. Karena tidak semua website bersifat statis, maka Google (salah satu yang menjalankan algoritma ini) memberikan penilaian secara berkala 1 bulan sekali. Oleh karena itu, saat user

melakukan pencarian, Google akan memberikan daftar situs berdasarkan ranking yang didapat pada bulan yang bersangkutan.

D. Teori EigenVector dan EigenValue

Eigenvector dan Eigenvalue adalah salah satu konsep penting dalam aljabar vektor, dimana keduanya saling berhubungan erat.

Sebuah bilangan real λ dikatakan sebagai eigenvalue bagi matriks A (matriks simetris berukuran $n \times n$) jika ada sebuah vektor v (ukuran n) yang memenuhi persamaan $A \cdot v = \lambda \cdot v$. Vektor v tersebut dinamakan sebagai eigenvector daripada eigenvalue λ . Contohnya adalah sebagai berikut.

Matriks A yang didefinisikan sebagai berikut.

$$\begin{pmatrix} 1 & 0 \\ 2 & 3 \end{pmatrix}$$

Eigenvalue yang bernilai 3 dan eigenvector P yang didefinisikan sebagai berikut.

$$\begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

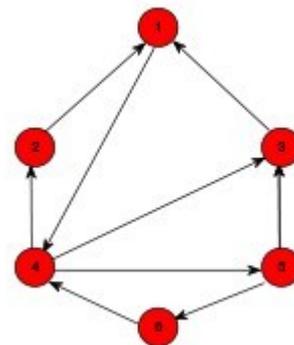
Jika ketentuan-ketentuan di atas dimasukkan ke dalam persamaan $A \cdot v = \lambda \cdot v$, maka persamaan di atas valid, sehingga benar jika 3 adalah eigenvalue bagi matriks A , dan eigenvector P berkorespondensi dengan eigenvalue bernilai 3 tersebut.

III. IMPLEMENTASI ALJABAR VEKTOR PADA ALGORITMA PAGE RANK

A. Page Rank Vector

Algoritma Page Rank menentukan relevansi berdasarkan jumlah link antar website lain dengan website yang bersangkutan.

Untuk sebuah kasus, asumsikan terdapat 6 buah website yang diberi indeks 1-6. Pengarahan link dari suatu website ke website lain dijelaskan sebagai berikut. Website 1 memiliki link menuju website 4, website 2 memiliki link menuju website 1, website 3 memiliki link menuju website 1, website 4 memiliki link menuju website 2, 3, dan 5, website 5 memiliki link menuju website 3 dan 6, dan website 6 memiliki link menuju website 4. Dengan memanfaatkan graf, maka hubungan antar website di atas dapat digambarkan sebagai berikut.



Gambar C1. Link antar website

Sumber:

https://www.math.washington.edu/~greenbau/Math_498/lecture03_pagerank.pdf

Selain merepresentasikan hubungan antar website dengan graf, kita juga dapat membuat sebuah matriks ketetangaan (*adjacency matrix*) nya. Berikut adalah matriks ketetangaan yang diperoleh (kita namakan matriks G).

	C1	c2	c3	c4	c5	c6
r1	0	0	0	1	0	0
r2	1	0	0	0	0	0
r3	1	0	0	0	0	0
r4	0	1	1	0	1	0
r5	0	0	1	0	0	1
r6	0	0	0	1	0	0

Pada matriks di atas, nilai 1 berarti terdapat link dari website ke-r dengan website ke-c.

Setelah mendapatkan matriks ketetangaan, kita dapat melihat bahwa setiap link yang ada pada sebuah website memiliki peluang yang sama untuk dikunjungi. Contohnya sebuah website memiliki 3 buah link menuju website lain, maka link ke-1 memiliki peluang diakses sebesar 1/3, link ke-2 memiliki peluang 1/3, dan link ke-3 memiliki peluang 1/3. Secara umum, jika sebuah website memiliki link sebanyak k buah, maka peluang akses setiap link itu adalah 1/k.

Ternyata, perhitungan relevansi oleh algoritma Page Rank ini menggunakan nilai peluang akses setiap link, jadi bukan ditentukan dari apakah dari suatu website memiliki link ke website lain. Oleh karena itu, terdapat sebuah matriks yang khusus bernilai peluang akses setiap link dan dinamakan sebagai matriks Stochastic.

Berdasarkan matriks ketetangaan G sebelumnya, dapat dibentuk matriks Stochastic M sebagai berikut.

	C1	c2	c3	c4	c5	c6
r1	0	0	0	1	0	0
r2	1	0	0	0	0	0
r3	1	0	0	0	0	0
r4	0	1/3	1/3	0	1/3	0
r5	0	0	1/2	0	0	1/2
r6	0	0	0	1	0	0

Setelah memperoleh matriks Stochastic di atas, kita akan melihat simulasi dari eksekusi algoritma Page Rank.

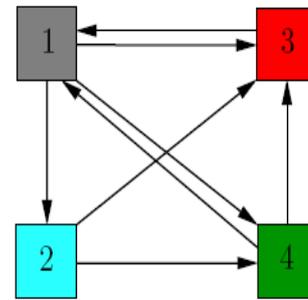
Pertama kita membentuk sebuah kondisi awal (*initial state*) untuk website ke-1 (asumsikan eksekusi dimulai dari website pertama) dalam bentuk vektor sebagai berikut: $v = [1 \ 0 \ 0 \ 0 \ 0 \ 0]$, dimana angka 1 menunjukkan state sekarang adalah website pertama. Kemudian, jika kita mengalikan vektor v dengan matriks Stochastic M diperoleh vektor baru $vM = [0 \ 0 \ 0 \ 1 \ 0 \ 0]$. Untuk menghitung vektor v_2 , maka langkah yang dilakukan adalah mengalikan v_1 dengan matriks Stochastic M, yaitu $v_2 = v_1.M$, dimana v_1 adalah $vM = [0 \ 0 \ 0 \ 1 \ 0 \ 0]$. Dalam kasus ini, nilai $v_2 = [0 \ 0.33 \ 0.33 \ 0 \ 0.33 \ 0]$. Langkah yang sama untuk mencari v_3, v_4, v_5 , sampai v_n , yaitu $v_k = V_k = V_{k-1} M$.

Berdasarkan penjelasan di atas, bentuk persamaan vektor dan matriks $vM = v$, dimana jika ada vektor positif yang memenuhi persamaan sebelumnya ($vM = v$), maka vektor positif itu disebut sebagai *steady-state vector*. Persamaan $vM = v$ dapat dipandang dari sudut lain, dimana vektor v dapat disebut juga sebagai *left-eigen vector* bagi matriks Stochastic M.

Mesin pencari Google mendefinisikan Page Rank dari website ke-k dalam bentuk V_k . Oleh karena itu, dapat disimpulkan bahwa nilai elemen maksimum dari V berhubungan langsung dengan nilai relevansi tertinggi, dan nilai elemen minimum dari V berhubungan langsung dengan nilai relevansi terendah. Oleh karena Google merepresentasikan Page Rank dari sebuah website dalam bentuk eigen-vector, maka dapat dikatakan bahwa Page Rank adalah sebuah vektor.

Mari kita lihat contoh penerapan aljabar vektor pada algoritma Page Rank yang lain.

Misalkan terdapat koneksi/ link antar website yang digambarkan dalam graf sebagai berikut.



Gambar C2. Link antar website
Sumber:

<http://www.math.cornell.edu/~mec/Winter2009/RalucaRebus/Lecture3/lecture3.html>

Dari graf di atas kita bisa mendapatkan matriks Stochastic M sebagai berikut.

	C1	c2	c3	c4
r1	0	0	1	1/2
r2	1/3	0	0	0
r3	1/3	1/2	0	1/2
r4	1/3	1/2	0	0

Matriks M di atas menjelaskan sebagai berikut. Website ke-1 memiliki link ke website 2, 3, dan 4 dimana masing-masing memiliki peluang akses sebesar 1/3. Begitu pula dengan website ke-2 yang memiliki link ke website 3 dan 4 yang masing-masing memiliki peluang akses sebesar 1/2.

Berdasarkan penjelasan sebelumnya, kita dapat melihat bahwa Page Rank merupakan sebuah vektor. Oleh karena itu, fokus kita saat ini adalah bagaimana cara mendapatkan vektor yang merepresentasikan Page Rank itu.

Kita mulai dengan merepresentasikan matriks Stochastic di atas ke dalam bentuk sistem persamaan linear.

$$\begin{aligned} x_1 &= x_3 + \frac{1}{2} x_4 \\ x_2 &= \frac{1}{3} x_1 \\ x_3 &= \frac{1}{3} x_1 + \frac{1}{2} x_2 + \frac{1}{2} x_4 \\ x_4 &= \frac{1}{3} x_1 + \frac{1}{2} x_2 \end{aligned}$$

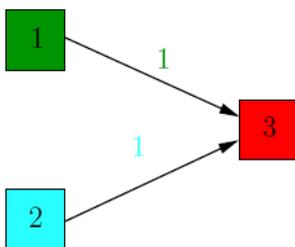
Sistem persamaan linear di atas ekuivalen dalam hal mencari nilai dari matriks A dalam persamaan berikut.

Persamaan: matriks A dikalikan dengan sebuah persamaan lain yang berisi $x_1, x_2, x_3,$ dan x_4 yang akan menghasilkan sebuah matriks berisi $x_1, x_2, x_3,$ dan x_4 .

Bentuk persamaan di atas adalah $A \cdot v = z \cdot v$ dimana merupakan bentuk eigenvalue dan eigenvector. Kita dapat melihat bahwa nilai eigenvalue adalah $z = 1$ dan eigenvector nya adalah $v = [x_1 \ x_2 \ x_3 \ x_4]$. Nilai dari eigenvector $[x_1 \ x_2 \ x_3 \ x_4]$ memiliki bentuk umum $c \cdot [12 \ 4 \ 9 \ 6]$, dimana c adalah konstanta. Kita harus memilih nilai konstanta c sedemikian sehingga total dari elemen-elemen eigenvector tersebut sama dengan 1, dengan kata lain $12c + 4c + 9c + 6c = 1$ dan kita dapatkan $c = 1/31$. Setelah digabungkan ke dalam persamaan, kita dapatkan eigenvector nya sebagai berikut $1/31 [12 \ 4 \ 9 \ 6] = [0.38 \ 0.12 \ 0.29 \ 0.19]$. Eigenvector ini disebut juga sebagai PageRank vector dimana ini adalah vektor yang merepresentasikan Page Rank.

B. Permasalahan umum

Sejak awal yang kita bahas adalah kumpulan website yang memiliki link ke website lain, dimana dalam representasi graf setiap vertex memiliki sisi menuju vertex lain. Jika kondisinya demikian, kita dapat mendapatkan Page Rank Vector dengan metode pada poin A sebelumnya. Lalu, bagaimana jika ada satu atau beberapa vertex yang tidak memiliki sisi menuju vertex lain, dimana artinya ada website yang tidak memiliki link menuju website lain? Berikut ilustrasinya.



Gambar C3. Link antar website
Sumber:

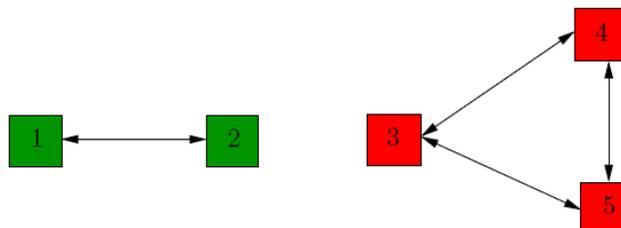
<http://www.math.cornell.edu/~mec/Winter2009/RalucaRebus/Lecture3/lecture3.html>

Website 1 memiliki link ke website 3, demikian dengan website 2, tetapi website 3 tidak memiliki link ke website manapun. Dalam kondisi di atas, maka website 3 dinilai penting oleh website 1 dan 2, dan seharusnya website 3 memiliki nilai rank / relevansi yang tinggi. Mari kita ilustrasikan perhitungan rank tersebut.

Dengan metode yang sama seperti poin A sebelumnya, kita membuat sebuah vektor untuk kondisi awal / *initial state* dari website 1, yaitu $v_0 = [1/3 \ 1/3 \ 1/3]$. Lalu kita mendapatkan v_1 dengan cara mengalikan v_0 dengan matriks ketetangaan, dimana elemen matriks ketetangaan itu bernilai 1 jika website ke- x memiliki link menuju website ke- y , dan bernilai 0 jika tidak ada link di antara kedua website tersebut.

Maka, v_1 bernilai $[0 \ 0 \ 2/3]$. Kita juga mencari nilai untuk v_2 yaitu dengan mengalikan matriks ketetangaan dengan v_1 , dimana didapat $v_2 = [0 \ 0 \ 0]$. Dari v_2 , dapat dilihat bahwa setiap website memiliki rank 0 dan ini merupakan sebuah kontradiksi dimana website 3 memiliki 2 *backlink*, sehingga seharusnya memiliki rank lebih dari 0.

Permasalahan lainnya adalah dimana komponen antar website tidak saling terhubung (*disconnected components*). Berikut ilustrasinya.



Gambar C4. Komponen link terpisah
Sumber:

<http://www.math.cornell.edu/~mec/Winter2009/RalucaRebus/Lecture3/lecture3.html>

Dari ilustrasi *small-internet* di atas, jika seorang berada di website 1, maka ia tidak akan bisa menuju website 5 dari website 1 karena tidak ada link yang menghubungkan website 1 dengan 5.

Berdasarkan 2 permasalahan umum di atas, kita dapat melihat bahwa akar permasalahannya terletak pada komponen dan lingkup dunia internet yang heterogen, dinamis, dan luas dimana kita tidak dapat mengharapkan semua website memiliki link ke website lain, sehingga kita dapat meluncur dari website ke- x ke website ke- y bermodalkan link antar website tersebut.

Lalu, apa yang perlu dilakukan untuk memanipulasi kondisi khusus tersebut? Kita memerlukan sebuah definisi lain yang bersifat tidak ambigu untuk menentukan rank dari sebuah website yang direpresentasikan sebagai graf berarah dengan n simpul.

Salah satu solusi yang dapat diimplementasikan adalah solusi dari Larry Page dan Sergey Brin yang dikenal dengan nama "The Solution of Page and Brin". Pertama-tama mari kita lihat persamaan berikut.

$M = (1 - p) \cdot A + p \cdot B$ dimana $M, A,$ dan B adalah matriks dan matriks B bernilai $1/n$ dikalikan dengan matriks 1 (matriks yang semua elemennya bernilai 1). Matriks M yang dibentuk dari persamaan di atas dinamakan sebagai Google Matrix yang merepresentasikan graf website. Matriks M juga bersifat Stochastic, artinya elemennya bernilai peluang akses setiap link website. Prinsip dari solusi ini adalah membatasi nilai p antara 0 sampai 1, dimana p ini dinamakan sebagai *damping factor*. Setelah menentukan nilai p yang sesuai, maka yang sebenarnya dilakukan oleh matriks Stochastic M ini adalah memanfaatkan nilai p untuk mendefinisikan peluang bagi user untuk keluar dari suatu website x untuk kemudian masuk ke website y . Karena user dapat pergi ke semua website yang memiliki backlink dari website asal, maka peluang bagi setiap

website untuk diakses adalah $1/n$, dimana n menyatakan jumlah website dalam representasi graf.

IV. KESIMPULAN

Algoritma Page Rank digunakan untuk memberikan indeks relevansi bagi suatu website, dimana indeks tersebut dapat direpresentasikan sebagai vektor yang didapat dari persamaan vektor dan matriks Stochastic.

V. REFERENSI

[1] http://math.stanford.edu/~brumfiel/math_51-06/PageRank.pdf

akses: Jumat 11 Desember 2015 jam 23.52

[2] http://ulva-s-nurmaryani-fst10.web.unair.ac.id/artikel_detail-60850-Aljabar%20Linier-Himpunan%20Vektor%20Bebas%20Linier.html

akses: Sabtu 12 Desember 2015 jam 09.00

[3] <http://www.math.cornell.edu/~mec/Winter2009/RalucaRemus/Lecture3/lecture3.html>

akses 1: Sabtu 12 Desember 2015 jam 12.10

akses 2: Senin 14 Desember 2015 jam 21.00

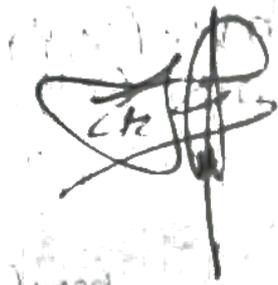
[4] <http://uyuhan.com/matematika/vektor/rangkuman-vektor-aljabar.php>

akses: Sabtu 12 Desember 2015 jam 19.00

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, 14 Desember 2015



Albertus Kelvin / 13514100