Teori Graf dalam Rantai Markov dan Implementasinya pada Algoritma PageRank

Richard Rivaldo 13519185[1]
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
[1]13519185@std.stei.itb.ac.id

***Abstrak*—Pada era data dan informasi ini intensitas kita dalam menggunakan mesin pencari atau yang kerap disapa *search engine* meningkat secara drastis dan signifikan. Salah satu mesin pencari populer yang digunakan oleh semua orang di seluruh belahan dunia adalah Google. Ternyata, algoritma yang menjadi cikal bakal Google, yaitu PageRank, dapat didefinisikan dalam sebuah model matematika yang menggambarkan rangkaian peristiwa dan kejadian yang mungkin terjadi, yaitu rantai Markov. Dalam konteks tersebut pula, rantai Markov ini memiliki ikatan fundamental dengan Teori Graf yang cukup besar dipengaruhi oleh Euler. Di dalam makalah ini, penulis akan menganalisis dasar Teori Graf dalam rantai Markov serta implementasi model tersebut dalam Algoritma PageRank.**

***Kata Kunci*—Mesin pencari, PageRank, rantai Markov, Teori Graf.**

I. PENDAHULUAN

Sejak zaman kerajaan yang didominasi oleh peperangan, kekuatan intelijen dan kemampuan untuk mendapatkan informasi menjadi suatu hal yang sangat penting. Intelijen dan mata-mata untuk mendapatkan informasi mengenai musuhnya melalui berbagai cara, seperti penyamaran dan penyusupan, menjadi sebuah dasar bagi para pemimpin untuk merumuskan dan mengambil keputusan selanjutnya untuk kepentingan dan keberlangsungan kerajaan mereka.

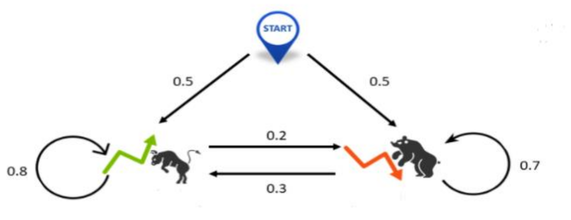
Namun demikian, peperangan telah berakhir dan dunia telah memasuki era baru yang ditopang oleh keberadaan teknologi. Pada zaman inipun informasi masih menjadi suatu hal yang sangat penting bagi manusia. Tidak bisa dibayangkan betapa banyaknya informasi yang keluar masuk setiap detiknya melalui arus digital yang ada. Adapun satu hal yang berbeda adalah manusia pada saat ini tidak perlu lagi melakukan hal-hal berbahaya seperti yang dilakukan pada zaman dahulu untuk mendapatkan informasi tersebut. Dengan nilai praktis yang disediakan oleh keberadaan teknologi yang ada saat ini, manusia dapat dengan mudah memperoleh informasi yang mereka inginkan melalui mesin pencari yang sudah banyak macamnya. Di antara mesin pencari atau yang biasa disebut dengan *search engine* tersebut, bisa dikatakan bahwa Google sungguh menguasai industri ini.

Google awalnya dikenali dengan nama BackRub, sebuah referensi untuk *back links* yang digunakan untuk meningkatkan akurasi pencarian. Larry Page dan Sergey Brin yang pada saat itu masih menjadi mahasiswa dari Universitas Stanford, California, Amerika Serikat, kemudian menemukan sebuah algoritma yang digunakan untuk mengurutkan hasil pencarian menjadi lebih relevan lagi. Hingga saat ini, Google menjadi sebuah gudang informasi yang sangat disegani karena inovasi dan investasi yang tinggi dalam bidang teknologi.

Seiring berjalannya waktu, semakin banyak informasi yang mengalami perubahan, baik dari segi kuantitas maupun kualitas. Kita bisa melihat banyaknya halaman yang ditampilkan oleh mesin pencari dengan hanya menginput satu kata saja sebagai *query*, belum lagi ketika kita menimbang banyaknya halaman yang dimiliki oleh sebuah *website*.

Dari banyaknya informasi dan data yang ada, mungkin kita sering bertanya-tanya mengenai cara atau proses yang bisa dilakukan untuk mengekstraksi informasi penting, seperti misalnya memprediksi cuaca berdasarkan cuaca di hari sebelumnya, angka atau permukaan dari dadu dan koin yang dimainkan, atau memprediksi tren harga untuk sebuah saham di pasar modal. Bisa jadi kita menemukan kemungkinan atau probabilitas untuk semua kejadian yang mungkin dalam setiap kasus tersebut. Lalu, kemudian apa?

Andrey Andreyevich Markov, seorang matematikawan Rusia yang saat itu menentang pernyataan Nekrasov tentang peristiwa yang bersifat dependen terhadap peristiwa yang terjadi sebelumnya, berhasil membuat sebuah model matematika yang hingga saat ini memegang peranan penting dalam berbagai bidang industri dan ilmu pengetahuan. Model matematika ini dikenal dengan nama *Markov Chain*, atau diartikan sebagai rantai Markov. Permasalahan-permasalahan tersebut kemudian bisa dimodelkan kembali sebagai sebuah rantai Markov asalkan kita mengetahui probabilitas setiap peristiwa, misalnya sebagai berikut.



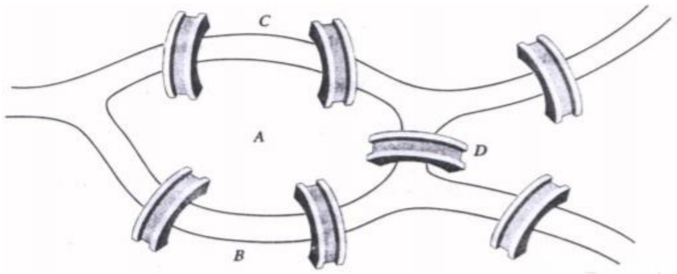
Gambar 1. Model Pasar Saham dalam Rantai Markov
(Sumber: https://letianzj.github.io/hidden-markov-chain.html)

Algoritma PageRank yang digunakan dalam mesin pencari Google ternyata juga bisa dimodelkan ke dalam rantai Markov. Dengan menggunakan representasi rantai Markov tersebut, maka prinsip dasar yang dimiliki oleh Algoritma PageRank dapat menjadi lebih mudah dipahami. Selain itu, Algoritma PageRank yang dimodelkan dengan Rantai Markov tidak hanya berhubungan dengan Teori Graf, tapi juga bisa berkaitan dengan vektor Eigen (*eigenvectors*).

II. LANDASAN TEORI

A. *Teori Graf*

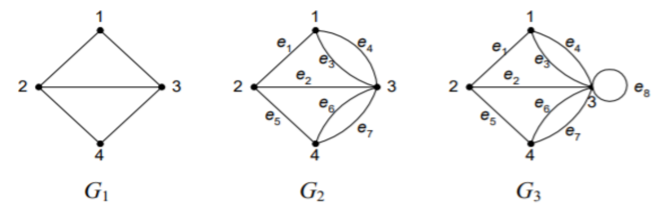
Pada tahun 1736 ada sebuah permasalahan yang cukup terkenal, yaitu persoalan Jembatan Königsberg. Bisakah seseorang kembali ke posisi semula dia berada dengan melewati semua jembatan tersebut tepat sekali?



Gambar 2. Visualisasi Persoalan Jembatan Königsberg (*Sumber:* [*https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf*](https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf))

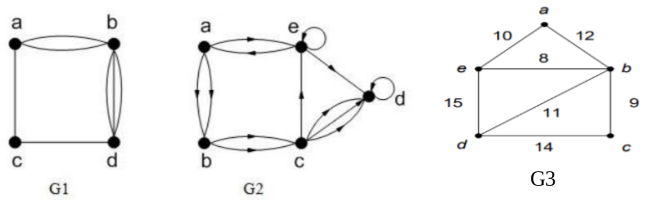
Permasalahan ini menjadi awal mula didefinisikannya prinsip-prinsip penting serta teori yang berkaitan dengan graf. Graf merupakan sebuah alat visualisasi untuk merepresentasikan hubungan yang dimiliki oleh objek-objek diskrit. Komponen umum dari sebuah graf G = (V, E) adalah adanya himpunan simpul V (*vertices)* dan sisi E (*edges*). Dalam konteks di atas, simpul dapat dinyatakan sebagai objek-objek diskrit dan sisi merupakan hubungan dari objek-objek tersebut.

Jika sisi dari suatu simpul menuju ke simpul tersebut pula, sisi tersebut disebut dengan gelang, kalang, atau *loop*. Jika ada dua sisi menuju ke suatu simpul yang sama dari sebuah simpul, sisi tersebut disebut sisi ganda. Graf yang tidak memiliki kedua jenis sisi tersebut dikenal dengan graf sederhana seperti pada G_1. Sebaliknya, graf bersisi ganda disebut *multigraph* pada G_2, dan graf yang memiliki sisi gelang disebut *pseudograph* pada G_3. Keduanya termasuk ke dalam kategori graf tidak sederhana.



Gambar 3. Graf Berdasarkan Jenis Sisi (*Sumber:* [*https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf*](https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf))

Selain itu, sisi graf juga dapat diberikan arah untuk menunjuk simpul terminalnya seperti pada gambar berikut. Graf yang sisi-sisinya demikian disebut dengan graf berarah atau *directed graph* (G_2), dan sebaliknya disebut dengan graf tidak berarah atau *undirected graph* (G_1). Sisi pada graf juga bisa diberiikan bobot untuk menentukan penting atau tidaknya hubungan antarobjek. Graf yang diberikan bobot disebut dengan graf berbobot atau *weighted graph* (G_3).

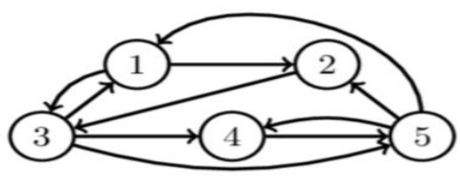


Gambar 4. Graf Berdasarkan Arah dan Graf Berbobot (*Sumber:* [*https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf*](https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf))

Hubungan ketetanggaan atau *adjacency* menyatakan simpul yang berhubungan langsung dengan simpul lain. Selain itu, hubungan kebersisian atau *incidency* menyatakan hubungan sisi dengan dua simpul yang dihubungkan olehnya. Jika sebuah simpul tidak memiliki sisi yang bersisian dengannya, atau dalam kata lain tidak memiliki tetangga, disebut sebagai simpul terpencil atau *isolated vertex*. Graf yang sisinya adalah impunan kosong disebut graf kosong atau *null graph*.

Jumlah sisi yang bersisian dari suatu simpul adalah derajat dari simpul tersebut. Untuk graf berarah sendiri, derajat simpulnya dibedakan menjadi derajat masuk dan derajat keluar. Derajat masuk merupakan jumlah sisi yang menunjuk dirinya, sedangkan derajat keluar merupakan jumlah sisi yang menunjuk simpul lain oleh simpul tersebut.

Di dalam sebuah graf ada yang disebut dengan lintasan dan sirkuit. Lintasan adalah jalur sisi-simpul yang bisa dilalui untuk mencapai sebuah simpul tertentu. Adapun sirkuit adalah lintasan yang berawal dan berakhir pada simpul yang sama. Adanya lintasan dalam graf memengaruhi sifat keterhubungan yang dimiliki oleh graf. Jika terdapat lintasan dari simpul v_1 menuju simpul v_2, maka dikatakan bahwa kedua simpuul tersebut terhubung. Dalam konteks graf berarah, graf disebut terhubung kuat apabila untuk setiap pasangan simpul sembarang v_1 dan v_2 di G, terdapat lintasan berarah dari v_1 menuju v_2 dan sebaliknya.



Gambar 5. Graf Berarah Terhubung Kuat (*Sumber:* [*https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf*](https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf))

Ada beberapa jenis graf khusus yang cukup penting untuk diketahui, yaitu misalnya graf lengkap, graf lingkaran, dan graf teratur. Graf lengkap adalah graf yang setiap simpulnya saling bertetangga satu sama lain. Graf lingkaran adalah graf yang setiap simpulnya berderajat dua. Adapun graf teratur adalah graf yang setiap simpulnya memiliki derajat yang sama.

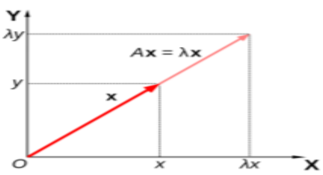
Sebuah graf bisa direpresentasikan dengan berbagai cara visual. Matriks ketetanggaan menyatakan semua simpulnya sebagai baris dan kolom, tetapi untuk matriks bersisian baris dari matriks adalah simpul dan kolom adalah sisi. Untuk tiap komponen kedua matriks tersebut, dimasukkan angka 1 jika antara baris dan kolomnya saling berhubungan dan 0 jika sebaliknya. Senarai ketetanggaan menggambarkan sifat ketetanggaan sebuah simpul dengan simpul lainnya. Dalam graf berarah, senarai ketetanggaan mendeskripsikan simpul terminal yang dimiliki oleh sebuah simpul.

B. Vektor Eigen

Kata *eigen* berasal dari Bahasa Jerman yang berarti ‘asli’ atau ‘karakteristik’. Dalam hal ini, nilai eigen menyatakan nilai karakteristik yang dimiliki oleh sebuah matriks persegi A_{NxN}. Untuk matriks tersebut, ada sebuah vektor tidak nol di R^N yang disebut dengan vektor eigen. Vektor eigen tersebut, yaitu **x**, memiliki hubungan dengan matriks A dan sebuah skalar $\boldsymbol{\lambda}$:

$$A\boldsymbol{x} = \boldsymbol{\lambda x}$$

Vektor eigen **x** merupakan sebuah vektor kolom yang menjadi pengali skalar dari vektor tersebut jika dikalikan dengan sebuah matriks persegi. Oleh karena itu, skalar $\boldsymbol{\lambda}$ memegang peranan penting untuk menentukan penyusutan dan penambahan panjang serta arah dari vektor kelipatan tersebut.



Gambar 6. Visualisasi Vektor Kelipatan dari Vektor Eigen (*Sumber:* https://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/2020-2021/Algeo-18-Nilai-Eigen-dan-Vektor-Eigen-Bagian1.pdf)

Untuk persamaan tersebut, dapat dikalikan sebuah matriks identitas dari matriks A untuk kedua ruas persamaan. Dari hasil pengalian tersebut, akan didapatkan hasil akhir berikut:

$$(\boldsymbol{\lambda} I - A)\boldsymbol{x} = 0$$

Supaya solusi yang dihasilkan dari persamaan tersebut tidak 0, atau dalam kata lain bukan solusi trivialnya, maka matriks yang dihasilkan dari $(\boldsymbol{\lambda} I - A)$ harus memenuhi syarat persamaan karakteristik berikut yang kemudian menghasilkan akar-akar karakteristik $\boldsymbol{\lambda}$ yang disebut nilai eigen.

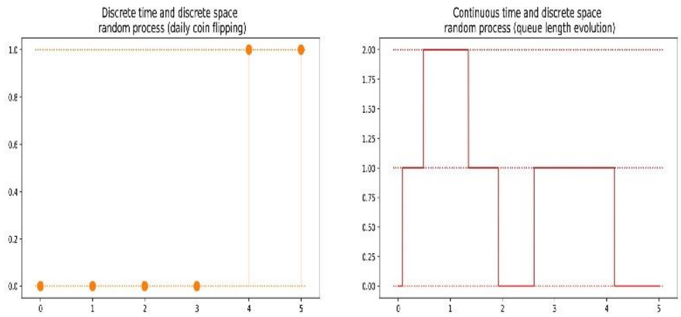
$$det(\boldsymbol{\lambda} I - A) = 0$$

III. Rantai Markov

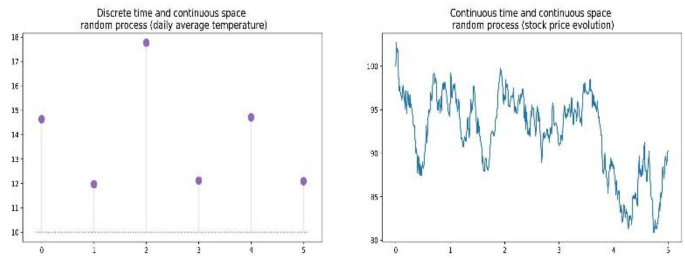
A. Variabel dan Proses Acak

Dalam teori probabilitas, ada dua hal yang harus dijelaskan terlebih dahulu sebelum memasuki rantai Markov. Kedua hal tersebut adalah variabel acak dan proses acak. Variabel acak, seperti namanya, merupakan sebuah variabel yang memilki nilai yang acak. Nilai yang acak dapat terjadi karena adanya fenomena acak yang mengakibatkan munculnya nilai tersebut. Variabel acak ini bisa jadi berupa variabel yang bernilai diskrit maupun kontinu. Contoh nyata dari variabel ini dapat kita lihat dengan mudah pada persoalan probabilitas matematika yang sering dipelajari, yaitu pelemparan koin atau dadu. Untuk setiap pelemparan dadu, kita bisa mendefinisikan variabel acak yang nilainya akan bergantung pada peristiwa ‘acak’, yaitu pelemparan dadu yang menyebabkan sisi atas dadu memiliki nilai yang acak pula.

Proses acak, atau yang lebih dikenal dengan proses stokastik dalam teori probabilitas, adalah sebuah proses yang pada akhirnya akan menghasilkan sebuah himpunan yang berisi nilai-nilai acak untuk setiap indeks waktu tertentu. Dalam hal ini, proses yang terjadi juga bisa merupakan proses diskrit maupun kontinu. Untuk setiap peristiwa yang terjadi pada waktu yang berbeda, bisa jadi kedua peristiwa tersebut memiliki ketergantungan terhadap yang lain, namun bisa juga tidak.



Gambar 7.1. Waktu Diskrit dan Kontinu (*Sumber:* https://towardsdatascience.com/brief-introduction-to-markov-chains-2c8cab9c98ab)



Gambar 7.2. Status Diskrit dan Kontinu (*Sumber:* https://towardsdatascience.com/brief-introduction-to-markov-chains-2c8cab9c98ab)

B. Rantai Markov

Rantai Markov merupakan sebuah model, sistem, atau proses stokastik matematika yang terjadi pada rentetan variabel acak yang ada karena suatu fenomena akibat proses Markov. Dalam konteks tersebut, rantai Markov seringkali terjadi dalam waktu dan ruang *state* yang diskrit. Waktu dan ruang status yang dimiliki oleh rantai Markov bisa jadi merupakan sebuah himpunan terbatas ataupun bukan. Akan tetapi, hal yang penting untuk digarisbawahi adalah setiap proses yang terjadi dalam rantai Markov terjadi pada waktu yang diskrit, berbeda dan tidak tumpang-tindih.

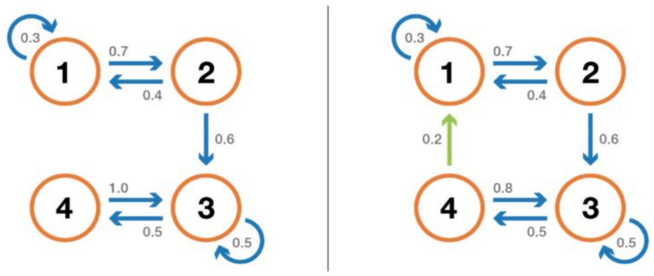
Misalkan X_n merupakan anggota dari sebuah himpunan elemen status E yang bernilai diskrit, dan n merupakan angka diskrit yang melambangkan waktu terjadinya proses Markov, maka rantai Markov secara matematis dapat dirumuskan sebagai berikut.

$$X = (X_0, X_1, X_2, \ldots, X_n) = (X_n)$$
$$X_n \in E$$

Proses yang dimiliki oleh rantai Markov bisa dikatakan sebagai proses stokastik yang berbeda dengan proses stokastik lainnya. Hal unik yang dimiliki oleh rantai Markov ini seringkali dikenal dengan properti Markov. Properti ini menyatakan bahwa ketika kita mengetahui nilai yang dimiliki dari suatu status pada waktu diskrit tertentu, maka kita tidak perlu peduli dengan nilai-nilai di status-status sebelumnya untuk memberikan gambaran mengenai status berikutnya. Hal tersebut mengakibatkan rantai Markov seolah-olah merupakan sebuah proses yang tidak memerlukan adanya memori atau disebut dengan *memoryless process*. Secara matematis, properti ini dapat dirumuskan sebagai berikut.

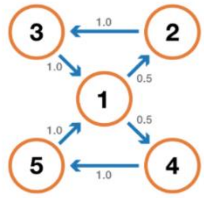
$$P(X_n = s_n \mid X_0 = s_0, X_1 = s_1, \ldots, X_{n-1} = i_{n-1})$$
$$= P(X_n = s_n \mid X_{n-1} = s_{n-1})$$

Selain memiliki properti dasar yang cukup unik, *Markov chain* juga memiliki beberapa properti lain yang bisa digunakan untuk mendeskripsikan rantai Markov tersebut. Properti tersebut adalah reduksibilitas, periodisitas, transien dan rekurens, ergodisitas, serta *absorbing state*. Sebuah rantai Markov dikatakan tidak bisa direduksi apabila kita bisa menuju suatu status dari status manapun, meskipun kita harus melewati beberapa tahapan untuk mencapai status tersebut.



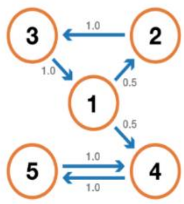
Gambar 8.1. Rantai Markov yang Bersifat Tidak *Irreducible* (Kiri) dan Bersifat *Irreducible* (Kanan)
(*Sumber: https://towardsdatascience.com/brief-introduction-to-markov-chains-2c8cab9c98ab*)

Sebuah *state* memiliki periode k apabila status tersebut bisa kembali ke status tersebut lagi dalam k langkah. Jika k bernilai 1, maka status tersebut dikatakan bersifat aperiodik. Jika semua status dalam rantai Markov bersifat aperiodik, maka dapat dikatakan rantai tersebut bersifat aperiodik.



Gambar 8.2. Rantai Markov dengan Semua Status Memiliki Periode 3
(*Sumber: https://towardsdatascience.com/brief-introduction-to-markov-chains-2c8cab9c98ab*)

Selain itu, sebuah status dikatakan sebagai status yang transien apabila ketika kita meninggalkan status tersebut, tidak ada siklus yang bisa membawa kita menuju status tersebut lagi. Sebaliknya, jika status tersebut memiliki sebuah siklus dan bisa kembali pada dirinya, maka status tersebut dikatakan sebagai status yang rekurens atau persisten. Apabila jumlah langkah yang diambil untuk kembali ke status awal adalah terbatas, maka status tersebut dikatakan memiliki sifat rekurens positif dan jika tidak maka bersifat rekurens kosong (*null recurrent*).



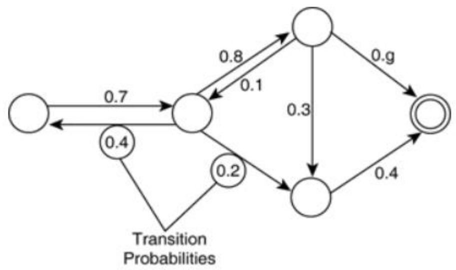
Gambar 8.3. Rantai Markov dengan Status 1, 2, dan 3 Bersifat Transien dan Status 4 dan 5 Bersifat Rekurens
(*Sumber: https://towardsdatascience.com/brief-introduction-to-markov-chains-2c8cab9c98ab*)

Sebuah status akan memiliki sifat ergodisitas apabila status tersebut merupakan status yang aperiodik dan memiliki rekurens positif. Jika semua status memiliki sifat ini, maka rantai tersebut dapat dikatakan sebagai rantai yang *ergodic*. Sifat terakhir, yaitu *absorbing state*, adalah sifat sebuah status yang tidak bisa keluar dari status tersebut, atau dalam kata lain kita tidak bisa berganti status ketika kita berada dalam status tersebut.

C. Representasi Rantai Markov

Untuk menggambarkan kemungkinan suatu peristiwa bisa terjadi karena adanya peristiwa lain, hal tersebut bisa dilakukan dengan menggunakan probabilitas sebagai perantara antarstatus yang ada. Dalam hal ini, probabilitas yang sebuah status ke status-status lainnya atau bahkan ke dirinya sendiri haruslah bernilai total satu.

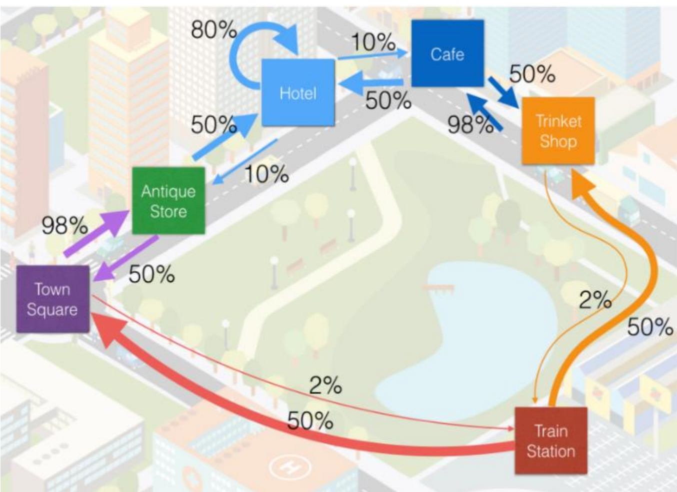
Sebuah rantai Markov bisa digambarkan dalam berbagai macam representasi. Contohnya, rantai Markov bisa digambarkan sebagai sebuah *Probabilistic Finite State Machine* yang ada di dalam Teori Bahasa Formal dan Automata, dengan status dari mesin tersebut menggambarkan semua status yang ada dalam rantai Markov dan fungsi transisinya menggambarkan probabilitas transisi yang dimiliki oleh sebuah status ke status lainnya sehingga sering disebut dengan *transition probabilities*.



$$PFSA = \{\Sigma, Q, F, \delta\}$$

Gambar 9. Ilustrasi Representasi dari Rantai Markov dalam Model Probabilistik *Finite State Machine* (FSM)
(*Sumber:* https://flylib.com/books/en/2.71.1.298/1/)

Namun demikian, representasi rantai Markov dalam bentuk graf dan matriks lebih sering digunakan secara bersamaan. Hal ini dikarenakan penggunaan representasi ini membuat hasil dari rantai Markov menjadi lebih jelas dan mudah terlihat, juga karena graf memiliki konsep-konsep dasar yang mirip dengan rantai Markov. Oleh karena itu, banyak orang sering menganggap bahwa rantai Markov dan Teori Graf memiliki kaitan yang cukup erat. Berikut merupakan contoh representasi visual dengan menggunakan graf dari sebuah rantai Markov yang merepresentasikan kemungkinan pergerakan seseorang di kota saat sedang berjalan-jalan.



Gambar 10. Representasi Permasalahan Rantai Markov dalam Bentuk Graf
(*Sumber:* https://www.compose.com/articles/graph-101-magical-markov-chains/)

Hanya dengan melihat graf tersebut, kita dapat dengan jelas mengetahui bahwa rantai Markov tersebut merepresentasikan kemungkinan tempat yang akan dituju selanjutnya oleh seseorang berdasarkan tempat yang mereka datangi saat itu. Dalam graf tersebut, status dengan jelas digambarkan sebagai sebuah simpul, sedangkan sisi yang menghubungkan simpul-simpul tersebut diberikan nilai berubah tingkat probabilitas dari setiap perpindahan yang mungkin terjadi.

Berdasarkan hal tersebut, dapat diketahui bahwa penggambaran Rantai Markov dalam graf menggunakan prinsip graf berarah sekaligus graf berbobot yang telah dijelaskan sebelumnya. Graf berarah digunakan untuk menentukan arah dari sebuah status menuju ke status berikutnya. Adapun graf berbobot digunakan dalam visualisasi rantai Markov untuk mendeskripsikan setiap probabilitas terjadinya sebuah kejadian diskrit yang ada dalam ruang status rantai Markov tersebut.

Mengingat bahwa graf dapat direpresentasikan dalam bentuk matriks, berarti model Markov ini juga bisa divisualisasikan demikian. Namun, kedua model matriks yang ada untuk menggambarkan graf, yaitu model matriks ketetanggaan dan matriks kebersisian kurang cocok untuk dijadikan sebagai representasi rantai Markov.

Model matriks ketetanggaan mungkin bisa menggambarkan arah dari status yang ada dalam rantai Markov, tetapi matriks ini tidak bisa menggambarkan probabilitas dari setiap sisinya. Model matriks lainnya, yaitu matriks kebersisian, memiliki problematika yang sama dengan matriks ketetanggaan. Dengan demikian, dari model matriks ketetanggaan yang ada, kita bisa memodifikasi pengisian komponen matriks.

Sebelumnya, komponen matriks *adjacency* diisi dengan satu dan nol, atau dengan jumlah sisi paralel yang menggambarkan hubungan ketetanggaan dari suatu simpul. Dengan mengganti prinsip pengisian tersebut menjadi pengisian elemen dengan probabilitas tiap sisi, maka matriks ini dapat digunakan sebagai visualiasasi dari rantai Markov. Model matriks yang demikian bisa dikatakan mirip dengan matriks transisi, seperti pada contoh berikut.

		Starting At					
		Train Station	Town Square	Antique Store	Hotel	Cafe	Trinket Shop
Going To	Train Station	0.0	0.02	0.0	0.0	0.0	0.2
	Town Square	0.50	0.0	0.50	0.0	0.0	0.0
	Antique Store	0.0	0.98	0.0	0.10	0.0	0.0
	Hotel	0.0	0.0	0.50	0.80	0.5	0.0
	Cafe	0.0	0.0	0.0	0.10	0.0	0.98
	Trinket Shop	0.50	0.0	0.0	0.0	0.5	0.0

Gambar 11. Representasi Permasalahan Rantai Markov pada Gambar 10 dalam Bentuk Matriks
(*Sumber:* https://www.compose.com/articles/graph-101-magical-markov-chains/)

D. Properti Graf Rantai Markov

Rantai Markov yang dapat direpresentasikan memiliki properti grafnya tersendiri (untuk selanjutnya akan disebut graf Markov). Hal ini dikarenakan adanya proses pendefinisian rantai Markov yang berbeda dibandingkan dengan bagian seharusnya dari objek-objek penyusun sebuah graf. Secara gamblang, hal ini telah dibahas tadi bahwa graf yang merepresentasikan model ini merupakan graf berarah sekaligus graf berbobot untuk mendefinisikan probabilitas dan status dari rantai Markov.

Ditinjau dari jenis dan jumlah sisi paralelnya, graf Markov tentu tidak memiliki sisi ganda karena semua probabilitas yang mungkin dari pasangan dua buah status akan digabungkan ke dalam sebuah sisi saja. Oleh karena itu, graf Markov tidak bisa dikemukakan sebagai *pseudograph* meskipun dalam graf ini sangat mungkin untuk memiliki sisi gelang atau *loop*.

Pendefinisian hubungan ketetanggaan dan kebersisian dalam graf ini sama saja dengan graf pada umumnya. Pendefinisian derajat masuk dan derajat keluar sendiri sebenarnya tidak terlalu diperlukan dalam graf Markov ini karena sangat jarang ada perhitungan yang melibatkan derajat tiap simpul dalam graf Markov. Namun, hal yang perlu diperhatikan adalah jumlah dari setiap bobot probabilitas suatu simpul harus bernilai satu.

Aturan untuk lintasan dan sirkuit masih berlaku untuk graf Markov ini. Dengan adanya pendefinisian demikian, maka kedua hal ini menjadi berkaitan erat dengan properti yang dimiliki oleh rantai Markov. Dalam graf Markov, langkah-langkah yang harus dilakukan untuk membentuk siklus dan lintasan dari suatu simpul menjadi sangat penting untuk dicatat karena akan menentukan properti rantai Markov yang didasari pada periodisitas pergerakan setiap simpulnya. Selain itu, graf Markov yang bersifat *irreducible* akan memiliki sifat graf yang terhubung kuat karena pengertian yang dimiliki untuk masing-masing properti adalah ekuivalen.

Adapun sifat *absorbing state* yang dimiliki oleh rantai Markov merupakan sifat yang mirip dengan *dead state* pada FSM.

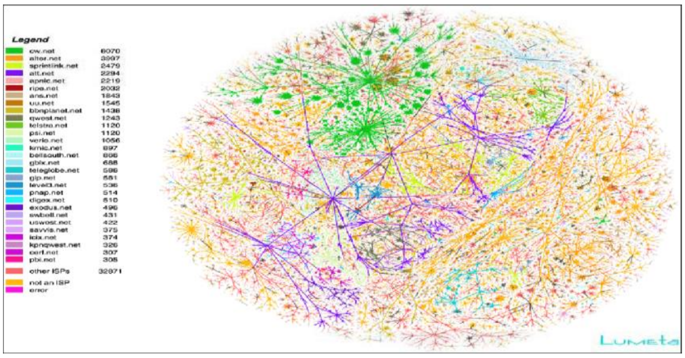
IV. ALGORITMA PAGERANK

Salah satu metode pencarian yang bisa digunakan dalam sistem temu-balik informasi untuk adalah dengan mengurutkan kemiripan teks biasa yang masuk ke dalam kategori karakter ASCII. Dalam hal ini pengukuran kemiripan yang bisa digunakan adalah dengan mengukur kemiripan Cosinus atau yang disebut dengan *cosine similarity*, yang konsep-konsep dasarnya diletakkan dalam ilmu Aljabar Linier dan Geometri.

Namun, semakin berkembangnya zaman, *World Wide Web* tumbuh dengan sangat cepat. Teknologi jaringan virtual yang ada semakin maju dan menyebabkan struktur jaringan yang ada menjadi semakin kompleks. Selain itu juga, karena penambahan volume informasi yang masuk ke dalam jaringan maka sistem pencari harus bisa memperbarui *database* informasi yang dimilikinya dengan tetap bisa mengurutkan informasi tersebut berdasarkan popularitas penggunaannya serta relevansi informasi yang dimiliki sebuah *website* dengan *query* yang ada.

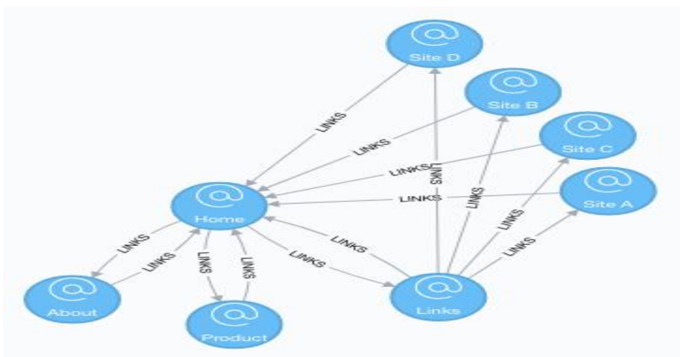
Algoritma PageRank merupakan sebuah algoritma yang dibuat untuk melakukan penyortiran tersebut. Hal yang membuat PageRank menjadi spesial adalah adanya konsiderasi terhadap struktur WWW yang sangat besar tersebut. Selain itu, algoritma ini juga menjadi salah satu algoritma yang memiliki tingkat efisiensi yang baik dalam sistem temu-balik informasi. Secara sederhana, PageRank adalah sebuah metode untuk menemukan dan mengurutkan sebuah *website* atau halaman berdasarkan ukuran baik tidaknya halaman tersebut ketika diurutkan dalam suatu pencarian.

Internet yang kita ketahui pada saat ini sebenarnya bisa digambarkan dalam strruktur graf dengan simpul menyatakan halaman atau *website* yang ada dan sisi menyatakan keterhubungannya dengan *website* lainnya. Tentunya, visualisasi untuk hal tersebut tidaklah cukup untuk menggambarkan semua halaman yang ada di dalam jaringan virtual tersebut. Seperti pada gambar di bawah ini, graf tersebut hanya menggambarkan topologi hubungan beberapa halaman saja, tetapi struktur sisi graf yang dimilikinya sudah sangat kompleks.



Gambar 12. Topologi Jaringan Beberapa *Website* dengan Representasi Graf
(*Sumber: https://www.researchgate.net/figure/2-A-graph-visualisation-of-the-topology-of-network-connections-of-the-core-of-the_fig2_239550496*)

Graf di atas hanya merepresentasikan topologi dari *webstie* yang ada. Untuk menggambarkan keterhubungan yang dimiliki oleh sebuah *website* dengan *website* lainnya, maka halaman tersebut perlu digambarkan sebagai sebuah graf berarah yang arahnya akan merepresentasikan *link* keluar-masuk dari sebuah halaman ke halaman lainnya. Hal ini bisa dilihat dalam sebuah sampel kecil yang ada pada gambar berikut.



Gambar 13. Representasi Situs dalam Graf Berarah
(*Sumber: https://neo4j.com/docs/graph-algorithms/current/algorithms/page-rank/*)

Ide awal dari PageRank adalah dengan memberikan sebuah angka yang merepresentasikan jumlah *link* yang merujuk pada sebuah halaman dari halaman lain. Tentunya, hal ini kurang cocok untuk dilakukan karena *link* dari situs-situs yang populer misalnya harus memiliki angka yang lebih besar. Percobaan lain misalnya adalah dengan memberikan angka yang merupakan jumlah dari tingkat otoritas yang dimiliki sebuah situs terhadap situs tersebut. Lagi-lagi, metode yang demikian kurang cocok, karena selain sistem yang dihasilkan hanya memiliki solusi trivial, juga rentan terhadap bias jika ada sebuah situs yang memiliki banyak *link* keluar.

Pada percobaan terakhir, Page dan Brin berhasil mendefinisikan pembobotan yang tepat untuk situs-situs tersebut. Pembobotan ini dilakukan dalam sebuah matriks Markov yang juga merupakan matriks stokastik *M*, yaitu matriks yang jumlah kolomnya merupakan adalah satu, yang menunjukkan jumlah probabilitasnya. Untuk sebuah matriks stokastik M_{NxN}, PageRank memiliki aturan pengisian komponen matriks sebagai berikut.

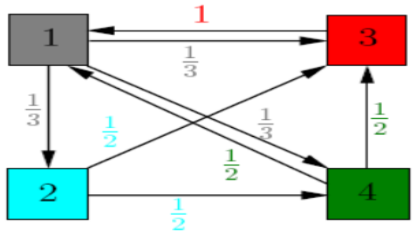
$$M_{ij} = \begin{cases} \frac{1}{n_j}, & n_j \textit{ adalah jumlah hubungan situs } j \textit{ dengan } i \\ 0, & \textit{Tidak ada hubungan antara kedua situs} \end{cases}$$

Dalam konteks rantai Markov, matriks PageRank bisa didefinisikan sebagai matriks yang setiap elemennya merupakan probabilitas dari tiap situs untuk menuju ke situs lainnya. Dalam hal ini, jika matriks tersebut direpresentasikan dalam graf Markov, maka simpul pada graf tersebut adalah himpunan situs-situs yang ada dan sisi merepresentasikan probabilitas transisi dari tiap situs ke situs lainnya. Misalkan kita memiliki matriks A_{4x4} berikut dengan baris dan kolomnya menunjukkan situs 1 sampai dengan situs 4.

$$A = \begin{bmatrix} 0 & 0 & 1 & \frac{1}{2} \\ \frac{1}{3} & 0 & 0 & 0 \\ \frac{1}{3} & \frac{1}{2} & 0 & \frac{1}{2} \\ \frac{1}{3} & \frac{1}{2} & 0 & 0 \end{bmatrix}$$

Maka, kita bisa mengetahui bahwa untuk komponen A_{21} yang bernilai 1/3 menunjukkan bahwa ada 3 buah *link* dari situs 1 yang merujuk ke situs 2. Selain itu, pada komponen A_{12} yang bernilai 0 menunjukkan bahwa tidak ada *link* yang merujuk ke situs 1 dari situs 2. Matriks ini juga memenuhi syarat sebuah matriks stokastik, yaitu matriks yang jumlah setiap kolomnya adalah satu dan setiap komponennya adalah angka non-negatif.

Matriks tersebut tentunya dibuat dengan asumsi bahwa keempat *website* tersebut telah difilter terlebih dahulu berdasarkan *query* yang dimasukkan oleh pengguna. Artinya, model sistem pencari hanya perlu mengurutkan tingkat otoritas dari setiap situs berdasarkan relevansi yang dimilikinya dengan *query* tersebut. Model matriks yang demikian berasal dari hipotesis Algoritma PageRank untuk memenuhi tujuan tersebut, yaitu semakin sering sebuah situs dikunjungi maka akan terdapat lebih banyak *link* keluar dari situs lain untuk menuju situs tersebut. Matriks yang demikian dapat direpresentasikan dalam graf Markov berikut.



Gambar 14. Representasi Matriks Pembobotan Setiap Situs dalam Graf Markov
(*Sumber:* *http://pi.math.cornell.edu/~mec/Winter2009/RalucaRemus/Lecture3/lecture3.html*)

Melalui matriks tersebut, kita bisa memodelkan sistem linier matriks yang ekuivalen dengan persamaan vektor Eigen sebelumnya. Dengan mengingat bahwa matriks *A* merupakan matriks stokastik, hal ini berarti bahwa nilai Eigen yang dimiliki oleh sistem persamaan tersebut adalah sebagai berikut.

$$\lambda = 1$$

$$A \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$

Dengan demikian, maka akan didapatkan vektor Eigen sebagai berikut. Vektor Eigen ini akan merepresentasikan tingkat otoritas situs tersebut, yang akan menjadi kunci dalam pengurutan situs-situs tersebut ketika pengguna memasukkan sebuah *query* terkait dengan halaman-halaman tersebut.

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 0.38 \\ 0.12 \\ 0.29 \\ 0.19 \end{bmatrix}$$

Vektor ini dikenal dengan nama vektor PageRank. Dalam hal ini, situs 1 menjadi situs dengan tingkat otoritas yang paling tinggi, diikuti situs 3, dan situs 2 menjadi situs dengan tingkat otoritas paling rendah, atau dalam kata lain memiliki relevansi yang rendah jika dibandingkan dengan situs-situs tersebut untuk *query* tertentu yang dimasukkan oleh pengguna.

Jaringan situs-situs yang ada nyatanya memang sangat masif dan kompleks. Tentu saja, ada kemungkinan bahwa graf yang digunakan untuk menggambarkan halaman-halaman yang ada di dalam internet tersebut menjadi tidak terhubung. Dalam kasus lain pula, bisa jadi bahwa sebuah situs atau halaman merupakan sebuah simpul terpencil atau *isolated vertex* yang tidak memiliki hubungan ketetanggaan dengan situs lainnya. Untuk mengatasi hal tersebut, Page dan Brin memberikan sebuah persamaan baru yang kemudian menjadi solusi untuk menyelesaikan persamaan tersebut. Dalam karya tulis ini, persamaan ini tidak akan dibahas lebih jauh karena memang persamaan ini memiliki keterkaitan yang lebih jauh dengan konsep-konsep vektor.

Persamaan tersebut adalah sebagai berikut.

$$M = (1 - d) \,.\, A + d \,.\, B$$

M = Matriks stokastik PageRank yang dihasilkan.
A = Matriks probabilitas transisi.
d = Faktor *damping*, memiliki rentang dari 0 sampai 1. Nilai yang sering digunakan adalah 0.15.
B = Sebuah matriks yang bisa didefinisikan sebagai berikut.

$$B = \frac{1}{n} \,.\, \begin{bmatrix} 1 & 1 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 1 \end{bmatrix}$$

V. LAIN-LAIN

Rantai Markov merupakan salah satu model matematika yang cukup unik untuk dipelajari. Selain Algoritma PageRank yang dimiliki oleh Google, rantai Markov juga digunakan dalam sebuah *bot* di dalam situs Reddit. *Bot* ini dinamakan *Subreddit Simulator*, yang melakukan berbagai aktivitas di dalam situs tersebut seperti memberikan komentar untuk suatu *post* atau menyukai foto dan *post* yang dibuat oleh manusia. Komentar yang dikeluarkan sendiri dimodelkan dengan menggunakan rantai Markov, yaitu kata-kata yang dikeluarkan dari sebuah kata tertentu didasarkan pada kata yang digunakan sebelumnya.

VI. SIMPULAN

Algortima PageRank merupakan sebuah algoritma mesin pencari milik Google yang digunakan untuk mencari tingkat otoritas setiap situs berdasarkan *query* yang dimasukkan oleh pengguna. Dari algoritma ini, mesin pencari bisa mengurutkan situs-situs yang akan ditampilkan pengguna berdasarkan relevansi dan popularitas situs tersebut terkait dengan informasi yang diinginkan dengan pengguna. Representasi struktur data yang ada dalam algoritma ini bisa dilakukan dengan menggunakan graf yang merepresentasikan rantai Markov. Dengan demikian, rantai Markov yang juga berkaitan erat dengan Teori Graf menjadi sebuah konsep dasar untuk algoritma tersebut.

Dasar-dasar Algoritma PageRank mungkin sudah tidak terlalu digunakan dalam algoritma mesin pencari yang dimiliki oleh Google saat ini, mengingat evolusi hebat yang dilakukan oleh perusahaan tersebut beberapa dekade ini. Namun, algoritma ini masih sangat menarik untuk dipelajari karena materi fundamental yang dimilikinya adalah konsep-konsep dasar dalam Matematika Diskrit dan Aljabar Linier dan Geometri. Selain itu, kita juga mengetahui bahwa setiap ilmu-ilmu dasar yang ada di dalam bidang Informatika memiliki keterkaitannya satu sama lain dan dapat dijadikan sebuah konsep yang *powerful* dan *versatile* untuk pembuatan teknologi kedepannya.

VII. UCAPAN TERIMA KASIH

Penulis mengucapkan puji syukur kepada Tuhan Yang Maha Esa atas segala bantuan lahir batin yang diberikan kepada penulis dalam menyelesaikan penyusunan karya tulis ini serta perkuliahan yang ada. Selain itu, penulis juga berterima kasih kepada keluarga yang telah senantiasa mendukung dan memberikan penulis semangat untuk menjadi pribadi yang lebih baik. Tidak lupa rasa hormat dan terima kasih yang sangat dalam dari penulis untuk semua dosen pengampu mata kuliah Matematika Diskrit IF2120 Semester 1 2020/2021 yang telah mengupayakan pembelajaran dalam masa pandemi ini. Selain itu juga, penulis mengucapkan terima kasih untuk teman-teman yang telah membantu dan menyemangati penulis.

DAFTAR PUSTAKA

[1] Anonim. *Linear Algebra – in a Nutshell*. Diakses melalui http://pi.math.cornell.edu/~mec/Winter2009/RalucaRemus/Lecture1/lecture1.html pada 5 Desember 2020 pukul 13.18 WIB.

[2] Anonim. *PageRank Algorithm – The Mathematics of Google Search*. Diakses melalui http://pi.math.cornell.edu/~mec/Winter2009/RalucaRemus/Lecture3/lecture3.html pada 5 Desember 2020 pukul 11.37 WIB.

[3] Anonim. *Random Variables and Processes*. Diakses melalui http://wits.ice.nsysu.edu.tw/course/pdfdownload/99CS/Comm-05-Random_Variables_and_Processes.pdf pada 4 Desember 2020 pukul 08.23 WIB.

[4] Anonim. *The PageRank Algorithm*. Diakses melalui https://neo4j.com/docs/graph-algorithms/current/algorithms/page-rank/ pada 5 Desember 2020 pukul 12. 10 WIB.

[5] Jaiswal, Sejal. *Markov Chains in Python: Beginner Tutorial*. Diakses melalui https://www.datacamp.com/community/tutorials/markov-chains-python-tutorial pada 4 Desember 2020 pukul 10.42 WIB.

[6] Jauregui, Jeff. *Markov Chains, Google's PageRank Algorithm*. Diakses melalui https://www2.math.upenn.edu/~kazdan/312F12/JJ/MarkovChains/markov_google.pdf pada 5 Desember 2020 pukul 14.34 WIB.

[7] Lisa. *Markov Chain (Discrete Time)*. Diakses melalui http://mytechroad.com/markov-chain-discrete-time/ pada 4 Desember 2020 pukul 09.21 WIB.

[8] Maltby, Henry, dkk.. *Markov Chains*. Diakses melalui https://brilliant.org/wiki/markov-chains/ pada 3 Desember 2020 pukul 22.25 WIB.

[9] Munir, Rinaldi. *Graf*. Diakses melalui https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/matdis20-21.htm#SlideKuliah pada 3 Desember 2020 pukul 19.02 WIB.

[10] Munir, Rinaldi. *Vektor di Ruang Euclidean (Bagian 2)*. Diakses melalui https://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/2020-2021/Algeo-18-Nilai-Eigen-dan-Vektor-Eigen-Bagian1.pdf pada 3 Desember 2020 pukul 20.13 WIB.

[11] O'Connor, John. *Graph 101: Magical Markov Chains*. Diakses melalui https://www.compose.com/articles/graph-101-magical-markov-chains/ pada 3 Desember 2020 pukul 21.42 WIB.

[12] Rocca, Joseph. *Introduction to Markov Chains*. Diakses melalui https://towardsdatascience.com/brief-introduction-to-markov-chains-2c8cab9c98ab pada 4 Desember 2020 pukul 13.37 WIB.

[13] Wenxing Ye. *On PageRank Algorithm and Markov Chain Reduction*. Diakses melalui https://www.cise.ufl.edu/~wye/Pagerank.pdf pada 5 Desember 2020 pukul 10.23 WIB.

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.

Palembang, 5 Desember 2020

Richard Rivaldo 13519185