

Aplikasi Graf Berarah untuk Predictive Search pada Google Search Engine

Haikal Lazuardi Fadil - 13519027
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
13519027@std.stei.itb.ac.id

Abstrak—Teknologi saat ini telah membantu mempermudah hidup kita. Salah satunya internet. Kita semua terhubung dengan satu sama lain dengan internet. Pengguna internet menggunakan *search engine* untuk mencari sesuatu yang mereka inginkan. Google Search adalah program *web search engine* yang dibuat untuk mencari informasi yang tersimpan pada suatu *database* sesuai dengan masukan pengguna. Salah satu *search engine* yang populer di masyarakat yaitu *Google Search*. Google Search juga berupaya agar *user experience* semakin membaik tiap tahunnya. Google pun berinovasi meluncurkan Google Autocomplete yaitu program yang berfungsi untuk memprediksi masukan pengguna. Dengan menggunakan graf berarah, Google Autocomplete dapat memprediksi masukan pengguna sehingga bisa mempersingkat waktu penerimaan *query*.

Kata Kunci — Graf, Google Search, Google Autocomplete, Prediksi.

I. PENDAHULUAN

Kemajuan teknologi informasi yang berkembang dengan sangat pesat memudahkan kita untuk beraktivitas sehari-hari. Salah satunya *web* dan internet yang sekarang memudahkan kita dalam mencari informasi.

Pengguna internet menggunakan *search engine* untuk mencari yang mereka inginkan. Mayoritas pengguna internet biasanya hanya tertarik pada laman pertama dari SERP (*Search Engine Result Page*). Tentu saja, *website* yang menempati posisi teratas berpotensi untuk lebih banyak dikunjungi dibanding yang lainnya.

Google Search adalah program *web search engine* yang dibuat untuk mencari informasi yang tersimpan pada suatu *database* sesuai dengan masukan pengguna. Google Search akan menampilkan hasilnya pada SERP. Hasil yang diberikan ada berbagai macam seperti *website*, gambar, berita, video, berita, dan petunjuk lokasi.

Google Search juga mempunyai kegunaan lain yang bisa kita manfaatkan untuk mempermudah hidup kita. Pertama, Google Search dapat digunakan untuk mencari definisi dari suatu kata. Hasil yang dikeluarkan berupa definisi dari kata tersebut, termasuk jenis apa, dan contoh penggunaannya.

Kedua, Google Search dapat digunakan untuk mencari informasi cuaca pada suatu wilayah. Hasil yang dikeluarkan berupa kecepatan angin, kelembapan, suhu, serta prediksi cuaca untuk beberapa jam kedepan. Sebagai contoh, dengan memasukkan **cuaca bandung**, akan keluar hasil berupa info

yang sudah disebutkan sebelumnya untuk daerah Bandung.

Ketiga, Google Search dapat digunakan sebagai kalkulator dengan memberikan masukan berupa angka yang harus dihitung. Hasil yang dikeluarkan berupa hasil hitungan dari masukan. Sebagai contoh, dengan memasukkan **56 x 65**, maka akan menghasilkan **3640**.

Keempat, Google Search dapat digunakan sebagai *timer*. Sebagai contoh, dengan memasukkan **set timer 30 minutes**, maka pada SERP akan muncul *timer* yang menghitung mundur selama 30 menit.

Selain untuk mencari informasi, Google Search juga dapat dimanfaatkan sebagai media pemasaran. Jumlah pengguna Google Search yang sangat banyak tentu menjadi banyak target bagi pengiklan untuk mengiklankan produknya di Google Search. Jika kita perhatikan secara seksama, biasanya di paling atas bagian hasil pencarian merupakan iklan yang berkaitan dengan informasi yang ingin kita dapatkan.

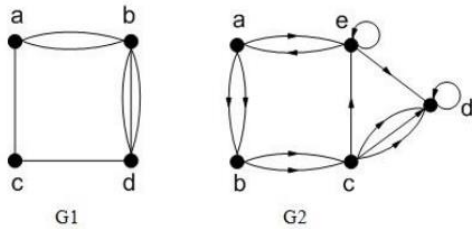
Google Search juga berupaya agar *user experience* semakin membaik tiap tahunnya. Google pun berinovasi meluncurkan Google Autocomplete yaitu program yang berfungsi untuk memprediksi masukan pengguna. Google Search memberikan prediksi bukan saran untuk masukan pengguna karena Google Autocomplete berfungsi untuk mempercepat proses masukan *query* oleh pengguna berdasarkan relevansi. Tentunya prediksi-prediksi yang dinilai tidak pantas tidak akan ditampilkan pada layar pengguna.

Pada makalah ini, penulis akan membahas tentang aplikasi graf berarah untuk *predictive search* pada Google Search Engine.

II. LANDASAN TEORI

Graf adalah himpunan yang terdiri dari simpul (*vertex* atau *node*) yang dihubungkan oleh sisi (*edge*). Misal G adalah sebuah graf maka $V(G)$ adalah simpul graf dan $E(G)$ adalah sisi graf dan membentuk pasangan himpunan $(V(G), E(G))$.

Berdasarkan orientasi arah pada sisi, graf dibagi menjadi dua yaitu graf tidak berarah dan berarah. Graf tidak berarah yaitu graf yang tidak mempunyai orientasi arah pada sisinya. Sedangkan graf berarah adalah graf yang memiliki orientasi arah pada sisinya

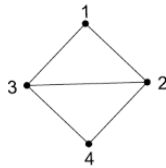


Gambar 2.1 G1 graf berarah dan G2 graf tidak berarah

(sumber:

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

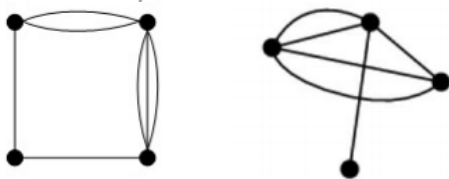
Berdasarkan ada tidaknya sisi ganda atau gelang, graf juga dibagi menjadi dua yaitu graf sederhana dan graf tidak sederhana. Graf sederhana adalah graf yang tidak mengandung sisi ganda atau gelang. Graf tidak sederhana adalah graf yang mengandung sisi ganda atau gelang



Gambar 2.2 Graf sederhana

(sumber:

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



Gambar 2.3 Graf tidak sederhana

(sumber:

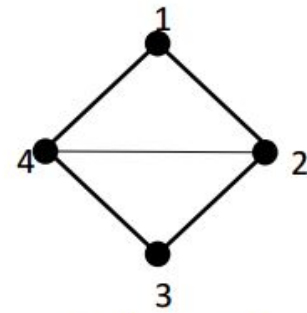
<https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf>)

Misalkan G adalah graf pada gambar 2.2. Maka $V = \{1, 2, 3, 4\}$ dan $E = \{(1, 2), (1, 3), (2, 1), (2, 3), (2, 4), (3, 4)\} = \{e_1, e_2, e_3, e_4\}$.

Graf memiliki beberapa terminologi, yaitu :

1. Bertetangga (*adjacent*)

Titik A dan B dikatakan bertetangga jika A terhubung langsung dengan B oleh satu sisi yang bisa disimbolkan dengan sisi $e = (A, B)$.



Gambar 2.4 Contoh graf yang bertetangga

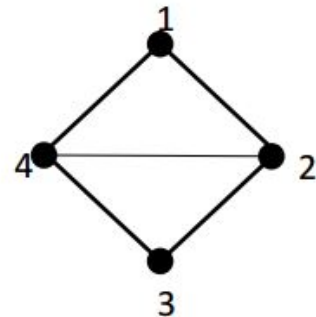
(sumber:

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

Misalkan graf pada Gambar 2.4 adalah graf G. Pada graf G, simpul 1 bertetangga dengan simpul 2 dan simpul 4. Simpul 2 bertetangga dengan simpul 1, simpul 4, dan simpul 3. Simpul 3 bertetangga dengan simpul 2 dan simpul 4. Simpul 4 bertetangga dengan simpul 1, simpul 2, dan simpul 3.

2. Bersisian (*incident*)

Misalkan ada sisi $e = (A, B)$, maka sisi e bersisian dengan simpul A dan simpul B.



Gambar 2.5 Contoh graf yang bersisian

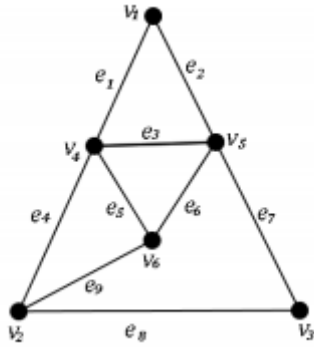
(sumber:

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

Misalkan graf pada Gambar 2.4 adalah graf G. Pada graf G, simpul 1 bersisian dengan sisi (1, 2) dan sisi (1, 4). Simpul 2 bersisian dengan sisi (2, 1), sisi (2, 4), dan sisi (2,3). Simpul 3 bersisian dengan sisi (3, 2) dan sisi (3, 4). Simpul 4 bersisian dengan sisi (4, 1), sisi (4, 2), dan sisi (4, 3).

3. Jalan (*walk*)

Jalan adalah suatu barisan yang elemen-elemennya terdiri dari ujung-ujung dan simpul-simpul yang tersusun bergantian, membuat ujung-ujung simpul-simpul itu $v_i + 1$ dan v_i .

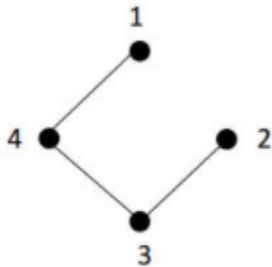


Gambar 2.6 Contoh jalan pada graf
(sumber:

<http://eprints.uny.ac.id/28515/2/BAB%20II.pdf>)

Misalkan graf pada Gambar 2.6 adalah graf G dengan sisi $V = \{v_1, v_2, v_3, v_4, v_5, v_6\}$ dan simpul $E = \{e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8, e_9\}$, yang membentuk barisan berhingga $W = \{v_1, e_2, v_5, e_3, v_4, e_4, v_2, e_9, v_6, e_6, v_5\}$.

4. Jejak (*trail*)
Jejak adalah jalan dengan sisi yang berbeda. Contoh walk pada Gambar 2.5 $W = \{v_1, e_2, v_5, e_3, v_4, e_4, v_2, e_9, v_6, e_6, v_5, e_7, v_3\}$.
5. Lintasan (*path*)
Lintasan adalah kumpulan simpul dan sisi yang bergantian, membentuk jalur dari simpul ke simpul.

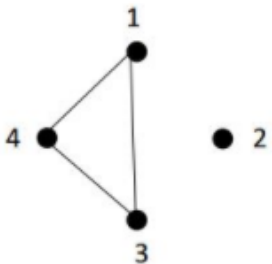


Gambar 2.7 Contoh lintasan pada graf
(sumber:

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

Misalkan graf pada Gambar 2.7 adalah graf G . Misalkan $e_1 = (1, 4)$, $e_2 = (4, 3)$, dan $e_3 = (3, 2)$. Lintasan yang terbentuk adalah $1, e_1, 4, e_2, 3, e_3, 2$.

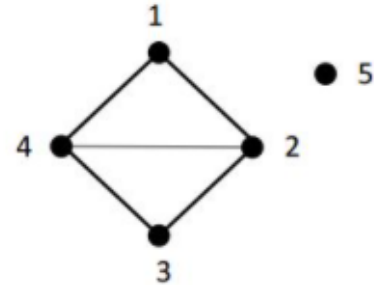
6. Sirkuit (*cycle*)
Sirkuit adalah lintasan yang dimulai dan diakhiri simpul yang sama.



Gambar 2.8 Contoh sirkuit pada graf

Misalkan graf pada Gambar 2.8 adalah graf G . Misalkan $e_1 = (1, 4)$, $e_2 = (4, 3)$, dan $e_3 = (3, 1)$. Maka sirkuit yang terbentuk adalah $1, e_1, 4, e_2, 3, e_3, 1$.

7. Simpul terpencil (*isolated vertex*)
Simpul terpencil adalah simpul yang tidak bersisian dengan sisi manapun dan tidak bertetangga dengan simpul manapun.



Gambar 2.9 Contoh simpul terpencil
(sumber:

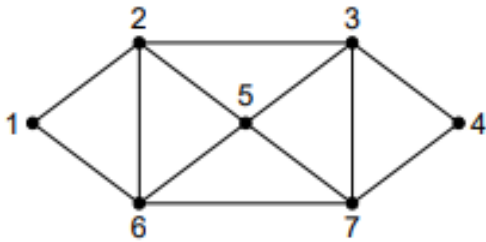
<https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf>)

Misalkan graf pada Gambar 2.9 adalah graf G . Simpul terpencil pada graf G adalah simpul 5 karena tidak terhubung ke simpul manapun.

8. Upagraf
Upagraf adalah kumpulan simpul dan sisi yang merupakan *subset* dari sebuah graf. Pada upagraf, simpul tidak boleh kosong sedangkan sisi boleh kosong. Sisi yang terdapat pada upagraf juga harus menjadi penghubung simpul-simpul pada upagraf.

Graf juga memiliki pengelompokan lain berdasarkan simpul dan sisinya. Pertama, ada graf Euler yang terbagi menjadi lintasan Euler dan sirkuit Euler. Sirkuit Euler adalah ketika setiap simpul dalam graf G muncul setidaknya satu kali dan setiap sisi muncul tepat satu kali. Graf yang mempunyai Sirkuit Euler disebut Graf Euler. Lintasan Euler adalah Lintasan yang melalui masing-masing sisi didalam graf G tepat satu kali. Graf yang mempunyai Lintasan Euler disebut Graf Semi-Euler.

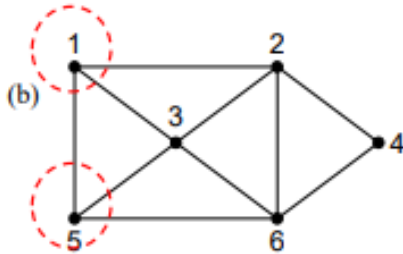
Graf Euler memiliki beberapa teorema. Pertama, Graf tidak berarah yang terhubung dengan setidaknya dua simpul memiliki sirkuit Euler jika dan hanya jika sebuah titik memiliki derajat genap. Kedua, Jika dan hanya jika grafik memiliki dua titik dengan derajat ganjil, grafik tersebut memiliki grafik yang terhubung dengan jalur acak tetapi tidak ada sirkuit acak. Ketiga, Graf berarah G memiliki lintasan Euler hanya jika G dihubungkan dan setiap titik memiliki derajat masuk dan keluar yang sama. kecuali dua titik, yang pertama memiliki derajat masuk satu lebih banyak dari pada derajat keluarnya dan titik yang kedua memiliki derajat keluar satu lebih banyak dari pada derajat masuknya. Sebaliknya graf berarah G memiliki sirkuit Euler hanya jika dihubungkan ke G dan setiap titik memiliki derajat masuk dan keluar yang sama.



Gambar 2.10 Contoh graf Euler

(sumber:

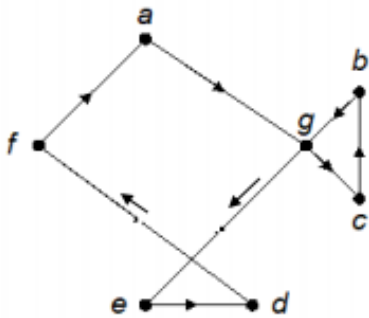
<https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian3.pdf>)



Gambar 2.11 Contoh graf Semi-Euler

(sumber:

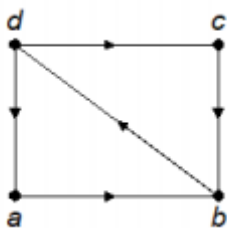
<https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian3.pdf>)



Gambar 2.12 Contoh graf berarah Euler

(sumber:

<https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian3.pdf>)



Gambar 2.13 Contoh graf berarah Semi-Euler

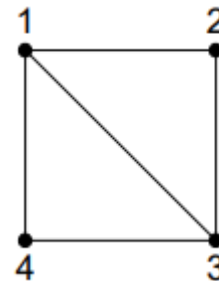
(sumber:

<https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian3.pdf>)

Pada Gambar 2.12, Graf berarah euler terdapat pada (a, g, c,

b, g, e, d, f, a). Sedangkan pada Gambar 2.13, graf berarah Semi-Euler terdapat pada (d, a, b, d, c, b).

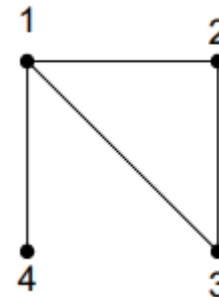
Kedua, ada graf Hamilton yang terbagi menjadi sirkuit Hamilton dan lintasan Hamilton. Lintasan Hamilton adalah lintasan sederhana di graf G yang melalui setiap titik tepat satu kali. Graf yang mempunyai Lintasan Hamilton disebut Graf Semi-Hamilton. Sirkuit Hamilton adalah sebuah sirkuit sederhana di graf G yang melalui setiap titik tepat satu kali (kecuali titik awal yang boleh dilalui dua kali). Graf yang mempunyai Sirkuit Hamilton disebut Graf Hamilton.



Gambar 2.14 Contoh graf Semi-Hamilton

(sumber:

<https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian3.pdf>)



Gambar 2.15 Contoh graf Hamilton

(sumber:

<https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian3.pdf>)

Pada Gambar 2.14, graf Hamilton terdapat pada (1, 2, 3, 4, 1). Sedangkan pada Gambar 2.15, graf Semi-Hamilton terdapat pada (3, 2, 1, 4).

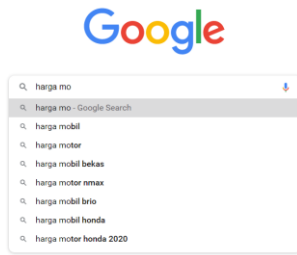
Misalkan graf pada Gambar 2.14 adalah graf G. Maka, syarat agar graf sederhana G terdapat graf Hamilton yaitu ketika graf dengan simpul lebih dari sama dengan tiga harus memiliki derajat untuk masing-masing simpul lebih besar sama dengan dua

III. PEMBAHASAN

A. Google Autocomplete

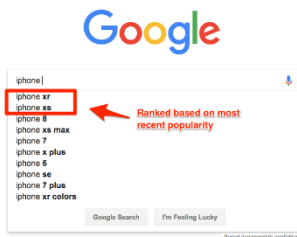
Google Autocomplete merupakan program yang berfungsi untuk memprediksi masukan pengguna. Google Search memberikan prediksi bukan saran untuk masukan pengguna karena Google Autocomplete berfungsi untuk mempercepat proses memasukkan *query* oleh pengguna berdasarkan *search*

query pengguna lain Google Search sebelumnya.



Gambar 3.1 Gambar Google Autocomplete

Google Autocomplete berjalan berdasarkan lokasi pengguna saat itu serta bahasa yang digunakan. Jika kita menggunakan Bahasa Indonesia, tentu prediksi yang akan muncul berbahasa Indonesia. Fitur *autocomplete* ini memberikan prediksi berdasarkan popularitas. Namun, bukan berarti prediksi yang ditampilkan berupa sesuatu yang selalu memiliki volume penelusuran paling tinggi tetapi fitur ini memberikan prediksi yang lebih mengarah pada relevansi. Algoritma Google memiliki pengukuran yang disebut "*freshness layer*" yaitu ketika ada *query* yang memiliki lonjakan tinggi dalam waktu yang singkat maka akan ditampilkan pada kolom prediksi. Selain itu, *history* dari *search query* pengguna juga memengaruhi Google Autocomplete.



Gambar 3.2 Algoritma *freshness layer*

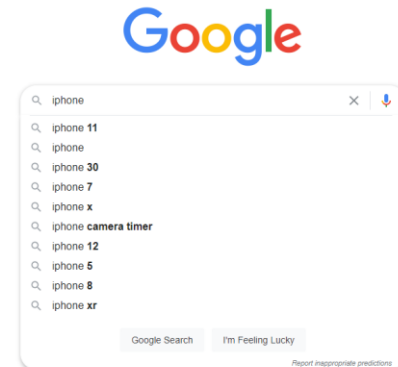
(sumber: <https://keywordtool.io/blog/google-autocomplete/>)

Masukkan dari pengguna sebelumnya akan ditampilkan pada kolom prediksi jika tetap memenuhi kriteria prediksi. Namun, *history* ini bisa dihapus jika menurut pengguna tidak perlu ditampilkan. Fitur Google Autocomplete ini tidak bisa dihapus dari Google Search pengguna karena merupakan fitur *built in* tetapi pengguna bisa memilih untuk tidak mengklik prediksi yang ditampilkan.

Ada saatnya ketika Google Autocomplete kehabisan prediksi. Biasanya terjadi Ketika pengguna sudah memasukkan banyak karakter pada *search bar* dan tidak biasa dicari oleh pengguna lain sehingga Google Autocomplete tidak memunculkan prediksi lagi. Perlu diingat juga bahwa prediksi yang diberikan tidak perlu dipilih oleh pengguna. Pengguna bisa saja memasukkan *query* yang diinginkan meskipun tidak sesuai dengan prediksi yang ada yaitu dengan cara menekan 'enter' pada *keyboard*.

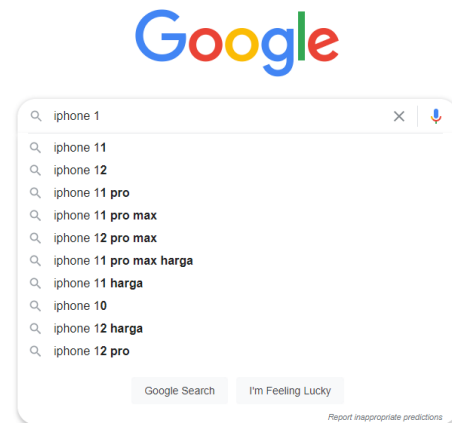
B. Aplikasi Graf Berarah Terhadap Google Autocomplete

Untuk memprediksi, Google Autocomplete membaca masing-masing karakter dari masukkan pengguna kemudian menampilkan prediksi yang memungkinkan dari masukkan pengguna.



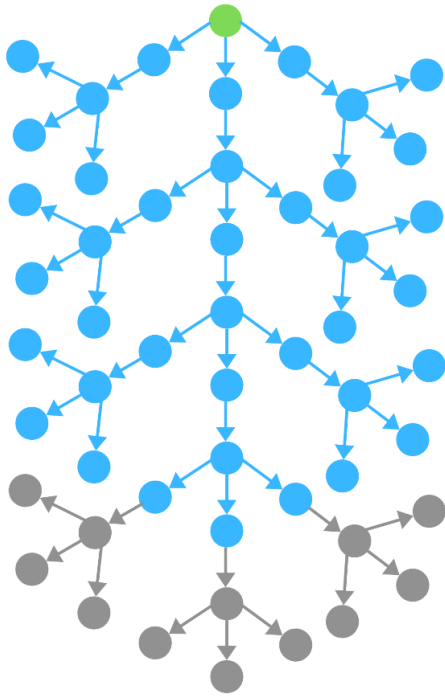
Gambar 3.3 Contoh prediksi awal

Jika kita memasukkan karakter yang berbeda dari prediksi yang sudah ada, maka akan muncul prediksi baru yang memungkinkan dari masukkan pengguna.



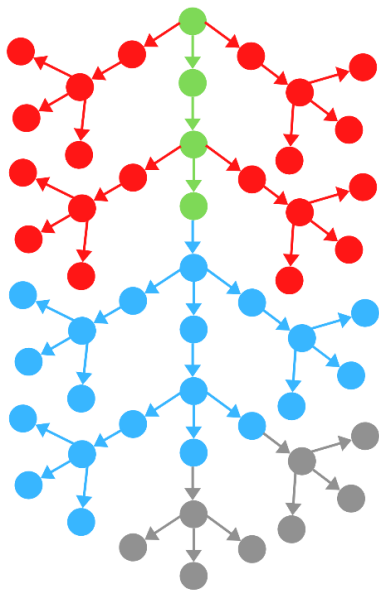
Gambar 3.4 Contoh prediksi lebih lanjut

Seperti yang kita lihat dari Gambar 3.3 dan Gambar 3.4, dengan memasukkan karakter lain prediksi pun berubah menyesuaikan dengan masukkan pengguna.



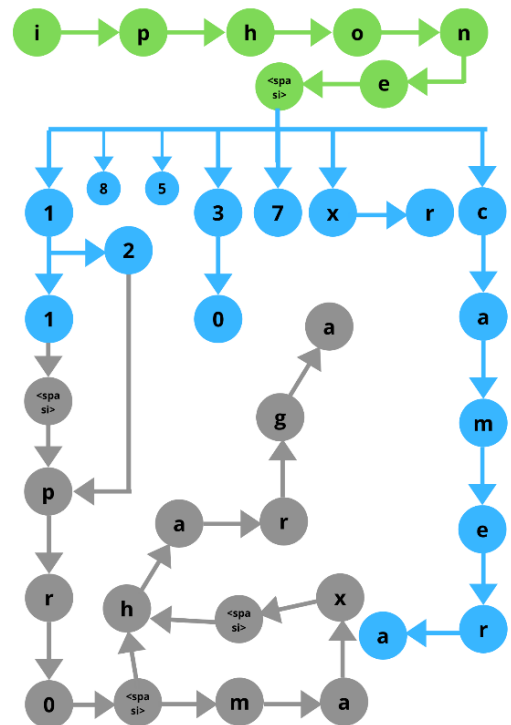
Gambar 3.5 Ilustrasi Google Autocomplete dengan graf berarah

Pada Gambar 3.5 terlihat graf berarah dengan sisi yang mengarah pada banyak simpul dengan satu simpul merepresentasikan satu karakter. Simpul yang berwarna hijau menyatakan karakter yang dimasukkan pengguna, bagian yang berwarna biru merupakan prediksi yang diberikan, dan bagian abu-abu merupakan prediksi yang akan ditampilkan oleh Google Autocomplete jika memasukkan kata lebih lanjut.



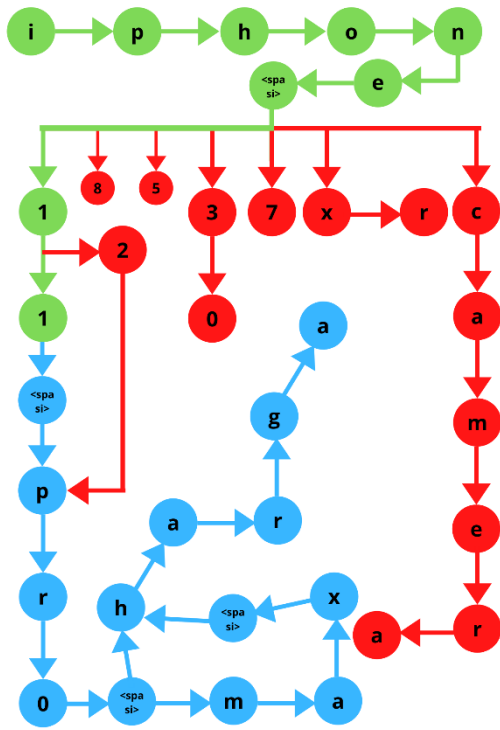
Gambar 3.6 Ilustrasi graf berarah setelah pengguna memasukkan banyak karakter

Pada Gambar 3.6 pengguna sudah memasukkan beberapa karakter sehingga prediksi yang ditampilkan pun berubah-ubah. Upagraf yang berwarna merah merupakan prediksi yang sempat ditampilkan pada layar pengguna namun berganti karena karakter yang dimasukkan pengguna sudah tidak memenuhi kriteria prediksi lagi. Beberapa bagian graf yang berwarna abu-abu berubah menjadi warna biru menandakan bahwa prediksi tersebut ditampilkan pada layar pengguna karena pengguna telah memasukkan karakter lebih banyak dan masih memenuhi kriteria Google Autocomplete.



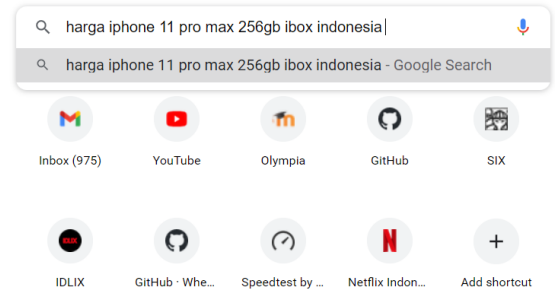
Gambar 3.7 Aplikasi graf pada Gambar 3.3

Gambar 3.7 merupakan gambaran graf untuk Gambar 3.3. Setelah pengguna memasukkan *query* 'iphone', prediksi yang keluar adalah 'iphone 11', 'iphone 30', 'iphone 7', 'iphone 5', 'iphone 8', 'iphone 5', 'iphone xr', dan 'iphone camera viewer' yang digambarkan pada graf sebagai warna biru. Warna abu-abu pada graf belum ditampilkan pada layar pengguna. Prediksi yang berwarna abu-abu sebenarnya lebih banyak daripada yang terdapat pada Gambar 3.7. Untuk mempermudah ilustrasi, penulis mempersingkatnya dengan hanya memberi ilustrasi sesuai dengan kebutuhan. Dengan kata lain, graf prediksi seharusnya lebih besar ukurannya melihat banyak sekali kemungkinan karakter masukkan pengguna.

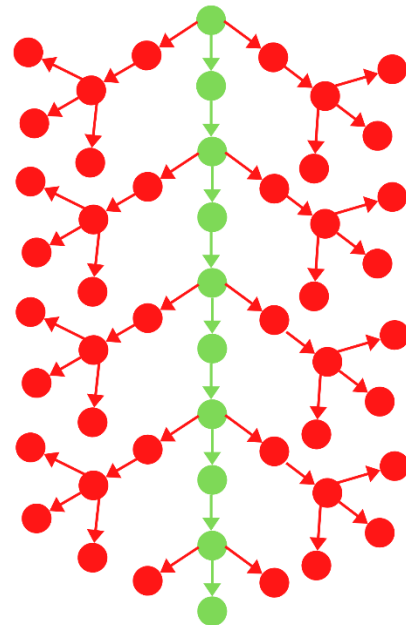


Gambar 3.8 Aplikasi graf pada Gambar 3.4

Gambar 3.8 merupakan gambaran graf untuk Gambar 3.4. Setelah pengguna memasukkan banyak karakter, Google Autocomplete menyesuaikan dengan memberikan prediksi baru yaitu prediksi yang awalnya berwarna abu-abu menjadi warna biru menandakan prediksi tersebut baru ditampilkan dengan menghapus prediksi yang sudah tidak lagi sesuai ditandai dengan mengubah bagian graf berwarna biru menjadi merah.

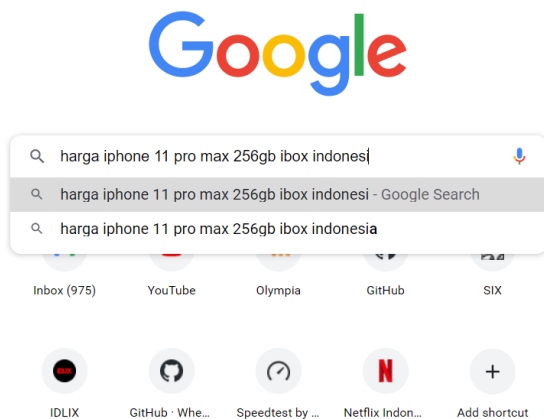


Gambar 3.10 Google Autocomplete kehabisan prediksi



Gambar 3.11 Ilustrasi Gambar 3.10

Ada kalanya Google Autocomplete kehabisan prediksi. Pada Gambar 3.9 terlihat bahwa Google Autocomplete tinggal memiliki satu prediksi lagi untuk masukkan pengguna dan kemudian prediksi pun habis seperti pada Gambar 3.10 dan ilustrasi dengan graf terdapat pada Gambar 3.11



Gambar 3.9 Google Autocomplete tersisa satu prediksi

IV. SIMPULAN

Graf jika dimanfaatkan dengan baik ternyata memiliki banyak kegunaan. Berdasarkan pemaparan diatas, graf bisa diaplikasikan untuk *predictive search* dengan cara membaca setiap karakter masukan pengguna kemudian memeriksa apakah kumpulan karakter tersebut sesuai dengan kriteria prediksi yang akan ditampilkan. Keberadaan *predictive search* ini tentu sangat berguna karena dapat mempersingkat waktu yang harus digunakan untuk memasukkan *query* serta meningkatkan *user experience* terhadap *search engine* tersebut.

V. UCAPAN TERIMA KASIH

Pada kesempatan kali ini, penulis mengucapkan terima kasih kepada Allah SWT atas rahmat-Nya, penulis dapat menyelesaikan makalah ini. Ucapan terima kasih juga penulis sampaikan kepada orang tua penulis karena telah mendukung penulis baik dari segi moral maupun material. Tentu saja, makalah ini jauh dari kata sempurna. Meskipun begitu, penulis tetap berusaha sekuat tenaga untuk menyelesaikan makalah ini sebaik mungkin. Penulis juga ingin berterima kasih kepada Ibu Fariska Zakhralatifa Ruskanda selaku pengampu mata kuliah IF2120 – Matematika Diskrit Kelas 03, beserta Ibu Harlili dan Pak Rinaldi Munir selaku dosen mata kuliah IF 2120 – Matematika Diskrit atas materi yang sudah disampaikan serta memberikan kesempatan kepada penulis untuk membuat makalah ini. Penulis berharap, makalah ini bisa menjadi pembelajaran bagi pembaca dan bisa dimanfaatkan dengan baik.

VI. DAFTAR PUSTAKA

- [1] Maxx, Axel. "How Search Engines Work [Beginner's Guide to SEO]." Moz, 2017, moz.com/beginners-guide-to-seo/how-search-engines-operate.
- [2] Albert, Quentine. "How Google Search Works." Google, Google, 2016, www.google.com/search/howsearchworks/.
- [3] Albert, Quentine. "How Search Predictions Work on Google - Android." Google Search Help, Google, 2018, support.google.com/websearch/answer/106230?co=GENIE.Platform.
- [4] Wembley, Clinton. "Google Autocomplete: 3 Clever Tips To Improve Your SEO." Keyword Tool Blog: How to Find Great Keywords, 7 Jan. 2020, keywordtool.io/blog/google-autocomplete/.
- [5] Sullivan, Danny. "How Google Autocomplete Works in Search." Google, Google, 20 Apr. 2018, blog.google/products/search/how-google-autocomplete-works-search/.
- [6] Eric, Vergner. "Fun Math Practice Games for Kindergarten to 5th Grade." SplashLearn, 2016, www.splashlearn.com/math-vocabulary/geometry/graph.
- [7] Maher, Akhsan. "Pengertian Graf Terminologi Graf." Beranda, 2016, text-id.123dok.com/document/lq53w1kgz-pengertian-graf-terminologi-graf.html.
- [8] Adiwadji, Hartoto. "MAT DIS: 5. Terminologi Graf." SPADA Indonesia, 2018, lmsspada.kemdikbud.go.id/mod/page/view.php?id=64470.
- [9] Munir, Rinaldi. 2014. Matematika Diskrit. Bandung: Informatika Bandung.
- [10] Kevin. 2012. Penggunaan Graf dalam Memahami Huruf Mandarin Makalah Program Studi Teknik Informatika, Institut Teknologi Bandung. Bandung: tidak diterbitkan.

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, 11 Desember 2020



Haikal Lazuardi Fadil
13519027