

Tugas Besar 2 IF 2123 Aljabar Linier dan Geometri
Aplikasi Dot Product pada Sistem Temu-balik Informasi
Semester I Tahun 2020/2021



Kelompok 71 (Sabab) :
Giovani Anggasta (NIM 13519155)
M. Ibnu Syah Hafizh (NIM 13519177)
Farhan Fadillah Rafi (NIM 13519204)

BAB 1

Deskripsi Masalah

Pada mata kuliah IF 2123 Aljabar Linier dan Geometri dipelajari vektor dengan *dot product*. Salah satu pengaplikasian *dot product* pada kehidupan sehari-hari adalah sistem temu-balik informasi atau yang lebih dikenal dengan *search engine*. Salah satu contoh *search engine* yang saat ini banyak digunakan oleh orang-orang di seluruh dunia adalah Google. Pada program ini akan dibuat *search engine* dengan sebuah *website* lokal sederhana. Spesifikasi program adalah sebagai berikut:

1. Program mampu menerima *search query*. *Search query* dapat berupa kata dasar maupun berimbuhan.
2. Dokumen yang akan menjadi kandidat dibebaskan formatnya dan disiapkan secara manual. Minimal terdapat 15 dokumen berbeda sebagai kandidat dokumen.
3. Hasil pencarian yang terurut berdasarkan similaritas tertinggi dari hasil teratas hingga hasil terbawah berupa judul dokumen dan kalimat pertama dari dokumen tersebut. Sertakan juga nilai similaritas tiap dokumen.
4. Program disarankan untuk melakukan pembersihan dokumen terlebih dahulu sebelum diproses dalam perhitungan *cosine similarity*. Pembersihan dokumen bisa meliputi hal-hal berikut ini.
 - a. *Stemming* dan penghapusan *stopwords* dari isi dokumen.
 - b. Penghapusan karakter-karakter yang tidak perlu.
5. Program dibuat dalam sebuah *website* lokal sederhana. Dibebaskan untuk menggunakan *framework* pemrograman website apapun. Salah satu *framework* website yang bisa dimanfaatkan adalah Flask (Python), ReactJS, dan PHP.
6. Dapat menambahkan fitur fungsional lain yang menunjang program yang anda buat (unsur kreativitas diperbolehkan/dianjurkan).
7. Program modular dan mengandung komentar yang jelas.
8. Tidak menggunakan *library cosine similarity* yang sudah jadi.

BAB 2

Teori Singkat

1. Perkalian Titik (*Dot Product*)

Dimisalkan terdapat vektor \mathbf{u} dan \mathbf{v} yang tidak nol pada \mathbb{R}^2 atau \mathbb{R}^3 sehingga dapat diperoleh perkalian titik atau *dot product* dari vektor \mathbf{u} dan \mathbf{v} adalah sebagai berikut

$$\mathbf{u} \cdot \mathbf{v} = \|\mathbf{u}\| \|\mathbf{v}\| \cos \Theta$$

dimana $\|\mathbf{u}\|$ merupakan panjang vektor \mathbf{u} , $\|\mathbf{v}\|$ merupakan panjang vektor \mathbf{v} , dan Θ merupakan sudut yang terbentuk di antara vektor \mathbf{u} dan vektor \mathbf{v} . Jika salah satu dari vektor \mathbf{u} atau vektor \mathbf{v} bernilai nol maka hasil dari perkalian titik juga nol.

Selain itu perkalian titik juga dapat dua buah vektor di \mathbb{R}^n juga dapat didefinisikan sebagai berikut:

Dimisalkan $\mathbf{u} = (u_1, u_2, \dots, u_n)$ dan $\mathbf{v} = (v_1, v_2, \dots, v_n)$ maka perkalian titiknya adalah

$$\mathbf{u} \cdot \mathbf{v} = u_1 v_1 + u_2 v_2 + \dots + u_n v_n$$

Dari rumus perkalian titik $\mathbf{u} \cdot \mathbf{v} = \|\mathbf{u}\| \|\mathbf{v}\| \cos \Theta$, dapat diubah menjadi

$$\cos \Theta = \frac{\mathbf{u} \cdot \mathbf{v}}{\|\mathbf{u}\| \|\mathbf{v}\|}$$

Diketahui bahwa $\mathbf{u} \cdot \mathbf{v} = u_1 v_1 + u_2 v_2 + \dots + u_n v_n$ sehingga didapat

$$\cos \Theta = \frac{u_1 v_1 + u_2 v_2 + \dots + u_n v_n}{\|\mathbf{u}\| \|\mathbf{v}\|}$$

2. Cosine Similarity

Kesamaan atau *similarity* dari dua buah vektor $\mathbf{Q} = (q_1, q_2, \dots, q_n)$ dan $\mathbf{D} = (d_1, d_2, \dots, d_n)$ dapat diukur dengan menggunakan rumus *cosine similarity* dimana rumus tersebut merupakan bagian dari perkalian titik vektor atau *dot product* dua buah vektor. Rumus cosine similarity adalah sebagai berikut

$$\text{sim}(\mathbf{Q}, \mathbf{D}) = \cos \Theta = \frac{\mathbf{Q} \cdot \mathbf{D}}{\|\mathbf{Q}\| \|\mathbf{D}\|}$$

Jika $\cos \Theta = 1$ maka vektor \mathbf{Q} dan \mathbf{D} berimpit karena memiliki $\Theta = 0$. Hal ini berarti bahwa vektor \mathbf{D} sesuai dengan vektor \mathbf{Q} . Apabila hasil dari *cosine similarity* mendekati 1 maka kedua vektor yang dibandingkan cenderung memiliki kesesuaian.

3. Sistem Temu-balik Informasi dengan *Cosine Similarity*

Temu-balik informasi merupakan sistem yang menemukan kembali informasi yang relevan terhadap kebutuhan pengguna dari kumpulan informasi secara otomatis. Temu-balik informasi biasanya digunakan pada pencarian informasi yang isinya tidak terstruktur. Contoh dari informasi tidak terstruktur yaitu dokumen, *website*, atau artikel. Pengaplikasian dari sistem temu-balik informasi adalah *search engine*. *Search engine* akan menerima input *query* atau *search key* lalu menampilkan *website* atau dokumen yang sesuai dengan *query* yang dicari. Salah satu *search engine* yang umum digunakan oleh banyak orang adalah Google.

Salah satu model sistem temu-balik informasi adalah dengan model ruang vektor. Dimisalkan terdapat m kata yang berbeda dalam suatu dokumen atau indeks kata. Kata-kata tersebut akan membentuk sebuah ruang vektor yang memiliki dimensi m . Setiap dokumen atau *query* akan dinyatakan sebagai vektor dimana isi dari vektor tersebut adalah jumlah kemunculan setiap kata unik yang ada di dalam dokumen tersebut. Penentuan dokumen yang sesuai dengan *query* ditentukan dengan mengukur kesamaan antara *query* dengan dokumen. Semakin sama suatu *query* dengan dokumennya maka akan semakin sesuai. Pengukuran kesamaan tersebut dapat dilakukan dengan memanfaatkan bagian dari perkalian titik atau dot product yaitu *cosine similarity*. Apabila hasil dari *cosine similarity* antara vektor dokumen dan vektor *query* semakin dekat dengan 1 maka dapat dinyatakan bahwa dokumen tersebut semakin sesuai dengan *query*.

Dalam sistem temu-balik informasi, setiap dokumen yang ada pada koleksi dokumen akan dihitung kesamaanya dengan *query* menggunakan *cosine similarity*. Kemudian dari hasil perhitungan tersebut akan dilakukan *pe-ranking-an* berdasarkan nilai *cosine similarity* dari besar ke kecil dimana hasil *cosine similarity* yang paling besar menyatakan bahwa dokumen itu paling sama dengan *query*. *Pe-ranking-an* tersebut menyatakan dokumen mana yang paling sesuai dengan *query* sehingga nilai *cosine similarity* yang besar menyatakan dokumen yang sesuai dengan *query* dengan nilai *cosine similarity* yang kecil menyatakan dokumen yang kurang sesuai dengan *query*.

BAB 3

Implementasi Program

Pada program yang kami buat, kami menggunakan beberapa *library* yang sudah disediakan oleh bahasa pemrograman Python yaitu *index*, *math*, *split*, *sort*, *dot*, *array*, *Counter*, *csr_matriks*, *pandas*, dan *os*. Selain itu, kami juga menggunakan *library* di luar bahasa pemrograman Python yaitu Sastrawi yang akan digunakan untuk stemming pada dokumen dan *query*.

Langkah pertama yang dilakukan dari program kami adalah membaca 15 dokumen serta melakukan input *query*, kemudian dilakukan stemming terhadap seluruh dokumen dan juga *query*. Selanjutnya dokumen dan *query* tersebut dimasukkan ke dalam suatu array. Pada program kami array tersebut kami beri nama variabel *doc*.

Pada program kami terdapat fungsi *dict_kata* yang berfungsi untuk mencari kata unik yang terdapat pada tiap dokumen dan juga *query*. Selanjutnya, akan dicari kata unik dari setiap dokumen dan *query* menggunakan fungsi *dict_kata*. Kemudian akan dihitung kemunculan kata unik pada setiap dokumen dan *query* menggunakan *library Counter*. Setelah itu, hasil perhitungan jumlah setiap kata unik pada masing-masing dokumen dan *query* akan dimasukkan ke dalam array dengan nama variabel *arrvec*.

Setelah terbentuk *arrvec*, akan dihitung perkalian titik antara vektor *query* dengan masing-masing vektor dokumen. Kemudian akan dihitung *cosine similarity* dari vektor *query* dengan masing-masing vektor dokumen. Selanjutnya hasil dari *cosine similarity* masing-masing dokumen akan diurutkan dari yang terbesar hingga yang terkecil

Selanjutnya, untuk menampilkan tabel pertama-tama digunakan bantuan 16 *dictionary* yang terdiri atas 15 *dictionary* dokumen dan satu *dictionary query*. *Dictionary* ini berfungsi untuk menyimpan value yang merupakan jumlah kemunculan kata dari kata yang ada pada *query*. dilakukan pengulangan kata pada *termq* yang mana *termq* adalah kalimat *query* yang telah di-stemming dan displit. Pada pengulangan, terdapat *conditional* yang berfungsi untuk mengecek apakah kata itu sudah ada di masing-masing *dictionary* term atau belum, Jika belum maka akan ditambahkan kata tersebut kedalam masing-masing *dictionary* term yang telah disiapkan beserta jumlah kemunculannya sebagai value.

Setelah didapatkan jumlah kemunculan kata dalam *query* pada setiap dokumen dan *query*, value dari masing-masing *dictionary* dijadikan *list* dan disimpan didalam variable bernama “termvalue1” – “termvalue15” untuk dokumen dan “termqvalues” untuk *query*. Kemudian dibentuk *list* yang berisi *list* value sehingga membentuk matriks, matriks ini disimpan dalam variabel ‘matriks_of_term’. Terakhir, dibuat tabel menggunakan fungsi *pd.DataFrame()* yang menyajikan jumlah kemunculan kata yang ada dalam *query* pada setiap dokumen dan *query* itu sendiri. Namun, tabel tersebut belum seperti yang ada pada deskripsi masalah sehingga

harus di-*transpose* terlebih dahulu menggunakan command yang sudah disediakan oleh pandas yaitu menambahkan (.T) pada akhir fungsi `pd.DataFrame()`.

Django

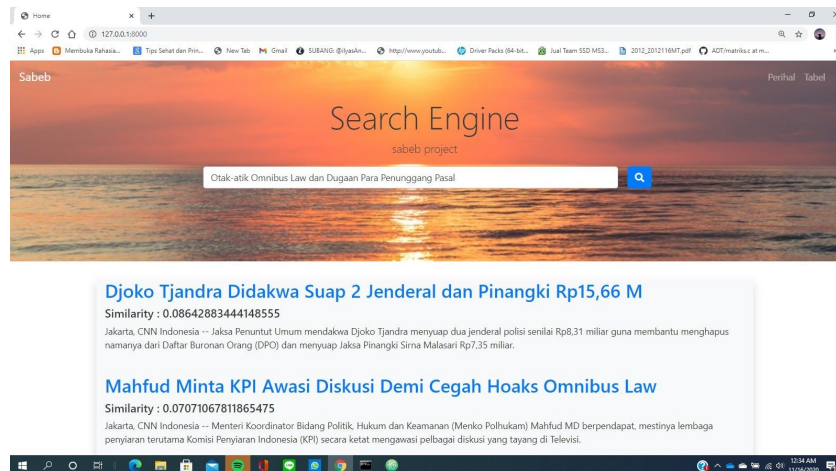
Program di atas kami simpan sebagai `views.py` dimana `views.py` ini yang akan menghubungkan ke `index.html` sebagai apa yang akan ditampilkan di web browser. Kami menggunakan variabel 'context' sebagai dictionary yang nantinya dipakai di `index.html` untuk ditampilkan menggunakan template tags django. variabel context ini memuat beberapa key yang mempunyai value untuk nantinya ditampilkan. key tersebut antara lain `cari`, `doc`, `tabel`, `header` di mana value dari key `cari` adalah hasil input dari form di `index.html`, value dari key `doc` berisi link, judul, similaritas, paragraf serta key `tabel` dan `header` ini untuk menampilkan tabel di `index.html`.

di sini kami membuat folder `static` yang sudah di setting di `setting.py` dengan base direktorinya adalah 'static'. Di folder `static` ini terdapat beberapa folder lain seperti `doc` yang berisi document uji serta folder `css`, `js`, `img` untuk memperindah tampilan web menggunakan bootstrap.

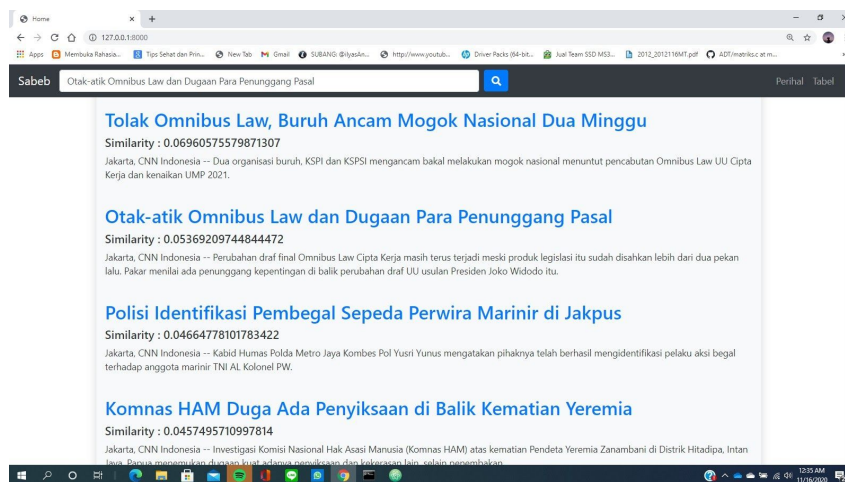
BAB 4

Eksperimen

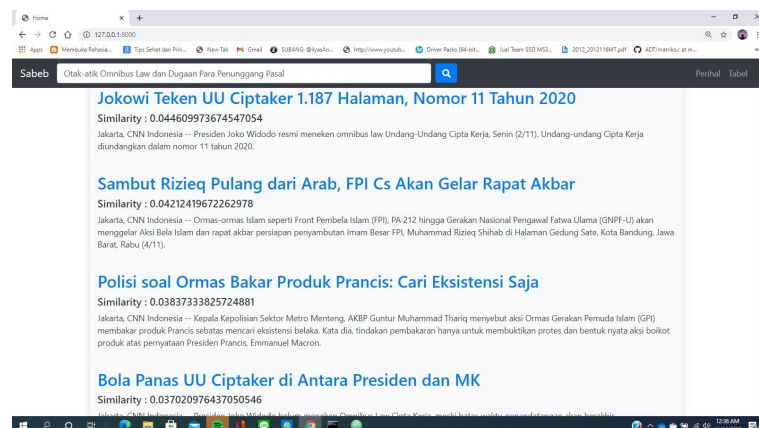
1. Test case dengan *query* salah satu judul dari artikel



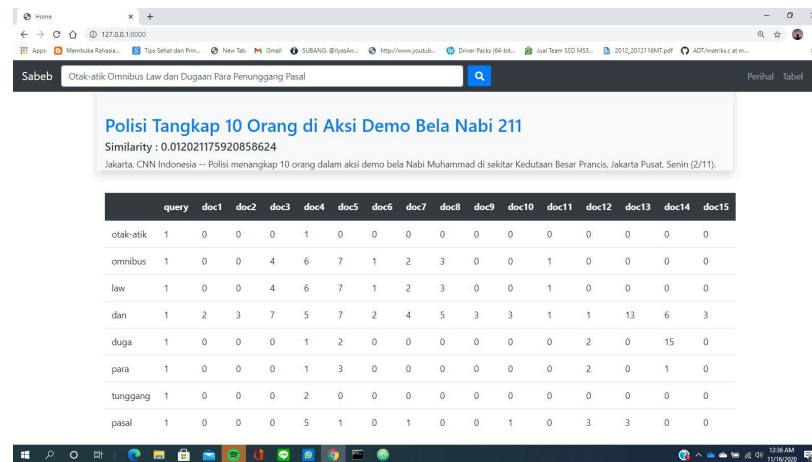
Gambar 4.1 Urutan Artikel Test Case 1.1



Gambar 4.2 Urutan Artikel Test Case 1.2



Gambar 4.3 Urutan Artikel Test Case 1.3



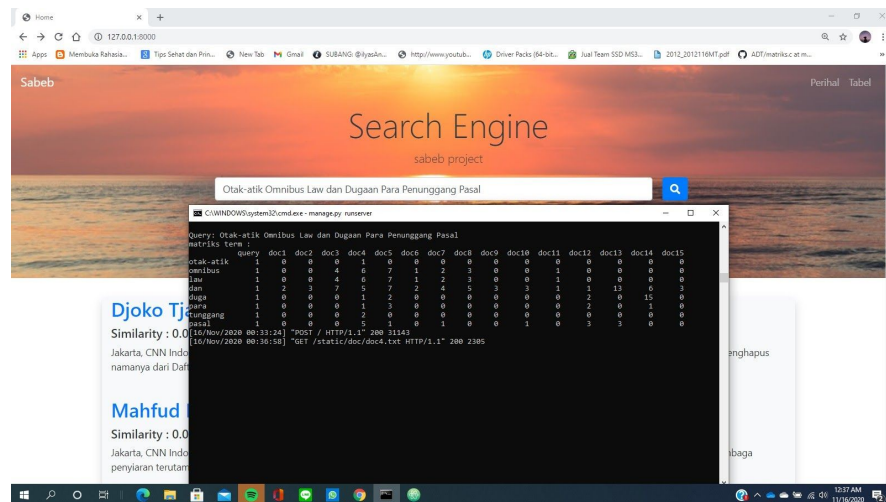
Home x + 127.0.0.1:8000

Sabeb Otak-atik Omnibus Law dan Dugaan Para Penunggang Pasal

Polisi Tangkap 10 Orang di Aksi Demo Bela Nabi 211
Similarity : 0.012021175920858624
Jakarta, CNN Indonesia -- Polisi menangkap 10 orang dalam aksi demo bela Nabi Muhammad di sekitar Kedutaan Besar Prancis, Jakarta Pusat, Senin (2/11).

	query	doc1	doc2	doc3	doc4	doc5	doc6	doc7	doc8	doc9	doc10	doc11	doc12	doc13	doc14	doc15
otak-atik	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
omnibus	1	0	0	4	6	7	1	2	3	0	0	1	0	0	0	0
law	1	0	0	4	6	7	1	2	3	0	0	1	0	0	0	0
dan	1	2	3	7	5	7	2	4	5	3	3	1	1	13	6	3
duga	1	0	0	0	1	2	0	0	0	0	0	0	2	0	15	0
para	1	0	0	0	1	3	0	0	0	0	0	0	2	0	1	0
tunggang	1	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0
pasal	1	0	0	0	5	1	0	1	0	0	1	0	3	3	0	0

Gambar 4.4 Tabel Term Sesuai dengan Kalimat yang Ada Pada *Query*



Home x + 127.0.0.1:8000

Sabeb Search Engine
sabeb project

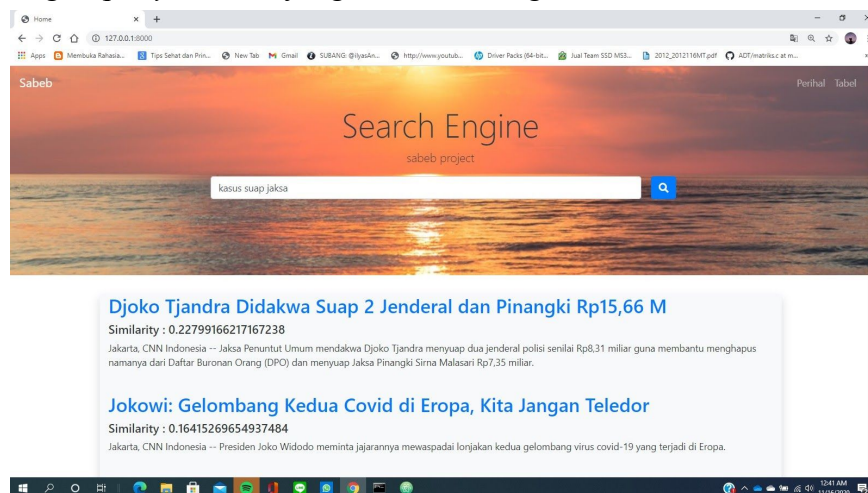
Otak-atik Omnibus Law dan Dugaan Para Penunggang Pasal

Djoko Tjandra Didakwa Suap 2 Jenderal dan Pinangki Rp15,66 M
Similarity : 0.022799166217167238
Jakarta, CNN Indonesia -- Jaksa Penuntut Umum mendakwa Djoko Tjandra menyuap dua jenderal polisi senilai Rp8,31 miliar guna membantu menghapus namanya dari Daftar Buronan Orang (DPO) dan menyuap Jaksa Pinangki Sirna Malasari Rp7,35 miliar.

Jokowi: Gelombang Kedua Covid di Eropa, Kita Jangan Teledor
Similarity : 0.16415269654937484
Jakarta, CNN Indonesia -- Presiden Joko Widodo meminta jajarannya mewaspadai lonjakan kedua gelombang virus covid-19 yang terjadi di Eropa.

Gambar 4.5 Hasil Dari Program Apabila Dijalankan Pada CMD

2. Test case dengan *query* kalimat yang berkaitan dengan isi artikel



Home x + 127.0.0.1:8000

Sabeb Search Engine
sabeb project

kasus suspi jaksa

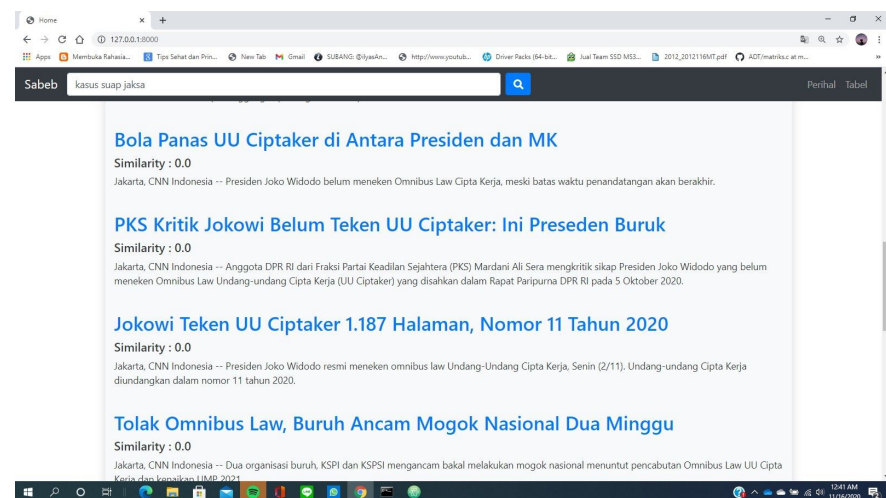
Djoko Tjandra Didakwa Suap 2 Jenderal dan Pinangki Rp15,66 M
Similarity : 0.022799166217167238
Jakarta, CNN Indonesia -- Jaksa Penuntut Umum mendakwa Djoko Tjandra menyuap dua jenderal polisi senilai Rp8,31 miliar guna membantu menghapus namanya dari Daftar Buronan Orang (DPO) dan menyuap Jaksa Pinangki Sirna Malasari Rp7,35 miliar.

Jokowi: Gelombang Kedua Covid di Eropa, Kita Jangan Teledor
Similarity : 0.16415269654937484
Jakarta, CNN Indonesia -- Presiden Joko Widodo meminta jajarannya mewaspadai lonjakan kedua gelombang virus covid-19 yang terjadi di Eropa.

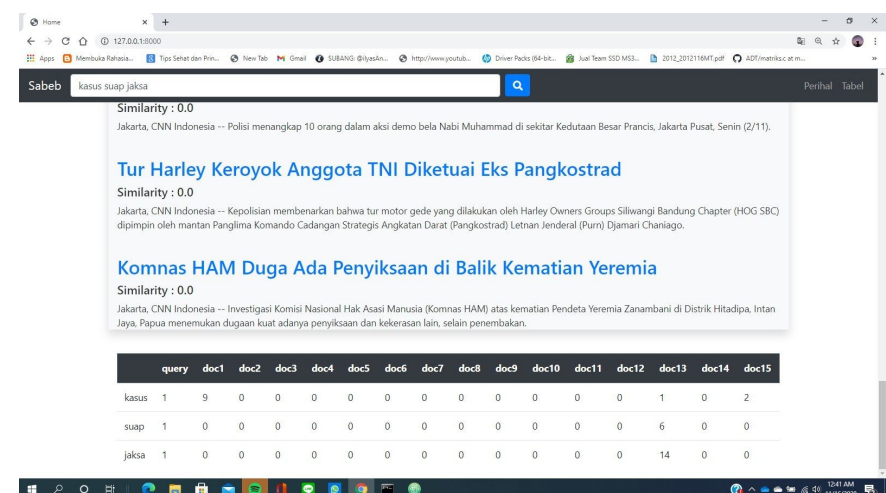
Gambar 4.6 Urutan Artikel Test Case 2.1



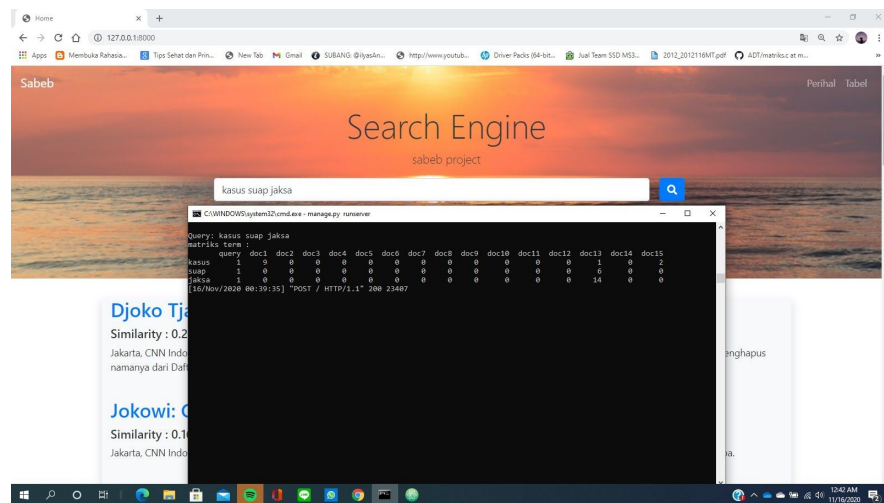
Gambar 4.7 Urutan Artikel Test Case 2.2



Gambar 4.8 Urutan Artikel Test Case 2.3



Gambar 4.9 Urutan Artikel Test Case 2.4 dan Tabel Term yang Sesuai dengan Query Test Case 2

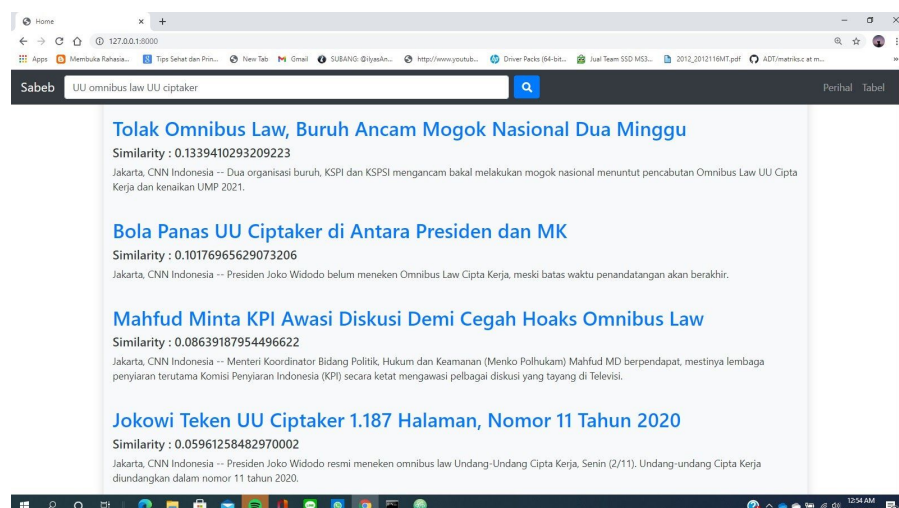


Gambar 4.10 Hasil Dari Program Apabila Dijalankan Pada CMD

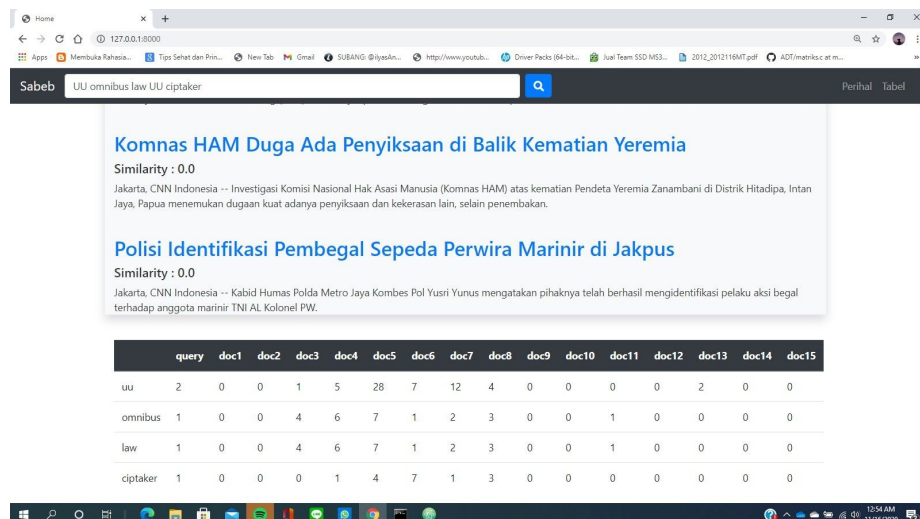
3. Test case dengan *query* yang memiliki kata yang berulang



Gambar 4.11 Urutan Artikel Test Case 3.1



Gambar 4.12 Urutan Artikel Test Case 3.2

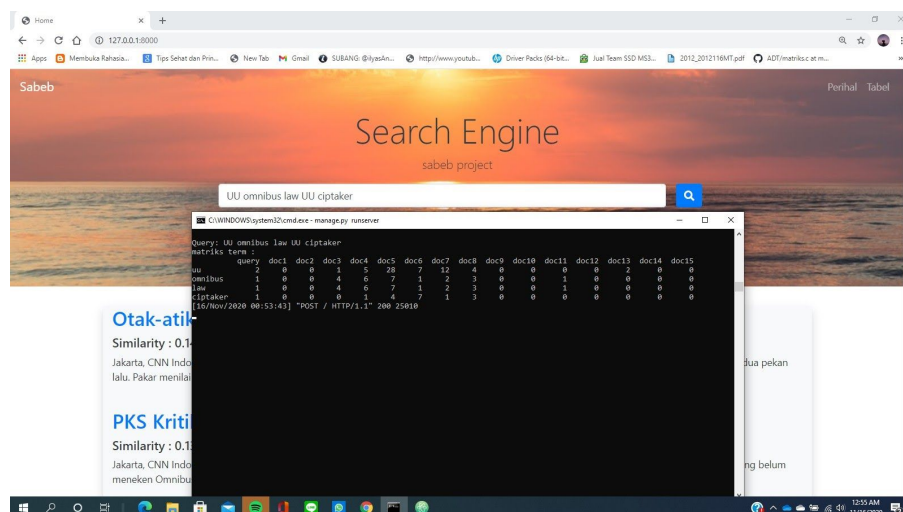


Komnas HAM Duga Ada Penyiksaan di Balik Kematian Yeremia
Similarity : 0.0
Jakarta, CNN Indonesia -- Investigasi Komisi Nasional Hak Asasi Manusia (Komnas HAM) atas kematian Pendeta Yeremia Zanambani di Distrik Hitadipa, Intan Jaya, Papua menemukan dugaan kuat adanya penyiksaan dan kekerasan lain, selain penembakan.

Polisi Identifikasi Pembegal Sepeda Perwira Marinir di Jakpus
Similarity : 0.0
Jakarta, CNN Indonesia -- Kabid Humas Polda Metro Jaya Kombes Pol Yusri Yunus mengatakan pihaknya telah berhasil mengidentifikasi pelaku aksi begal terhadap anggota marinir TNI AL Kolonel PW.

	query	doc1	doc2	doc3	doc4	doc5	doc6	doc7	doc8	doc9	doc10	doc11	doc12	doc13	doc14	doc15
uu	2	0	0	1	5	28	7	12	4	0	0	0	0	2	0	0
omnibus	1	0	0	4	6	7	1	2	3	0	0	1	0	0	0	0
law	1	0	0	4	6	7	1	2	3	0	0	1	0	0	0	0
ciptaker	1	0	0	0	1	4	7	1	3	0	0	0	0	0	0	0

Gambar 4.13 Urutan Artikel Test Case 3.3 dan Tabel Term yang Sesuai dengan *Query* Test Case 3



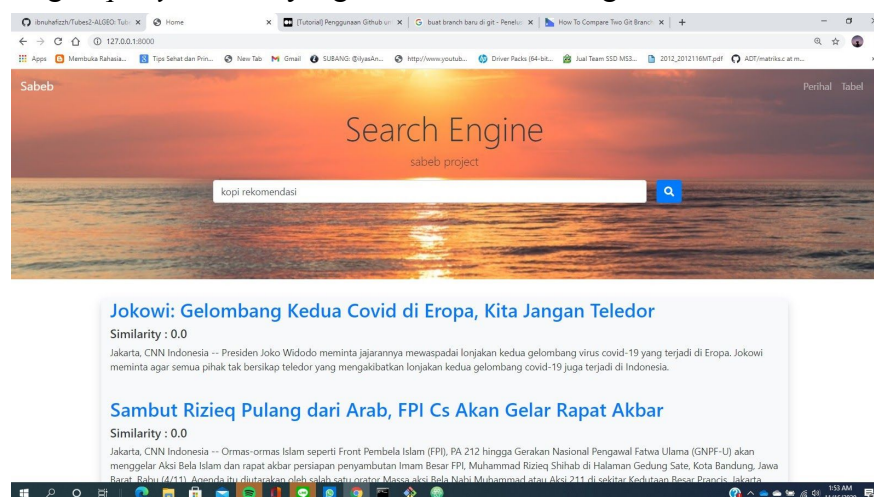
Otak-atik
Similarity : 0.1
Jakarta, CNN Indonesia -- Pakar menilai...

PKS Kritis
Similarity : 0.1
Jakarta, CNN Indonesia -- meneken Omnibus...

	query	doc1	doc2	doc3	doc4	doc5	doc6	doc7	doc8	doc9	doc10	doc11	doc12	doc13	doc14	doc15
uu	2	0	0	1	5	28	7	12	4	0	0	0	0	2	0	0
omnibus	1	0	0	4	6	7	1	2	3	0	0	1	0	0	0	0
law	1	0	0	4	6	7	1	2	3	0	0	1	0	0	0	0
ciptaker	1	0	0	0	1	4	7	1	3	0	0	0	0	0	0	0

Gambar 4.14 Hasil Dari Program Apabila Dijalankan Pada CMD

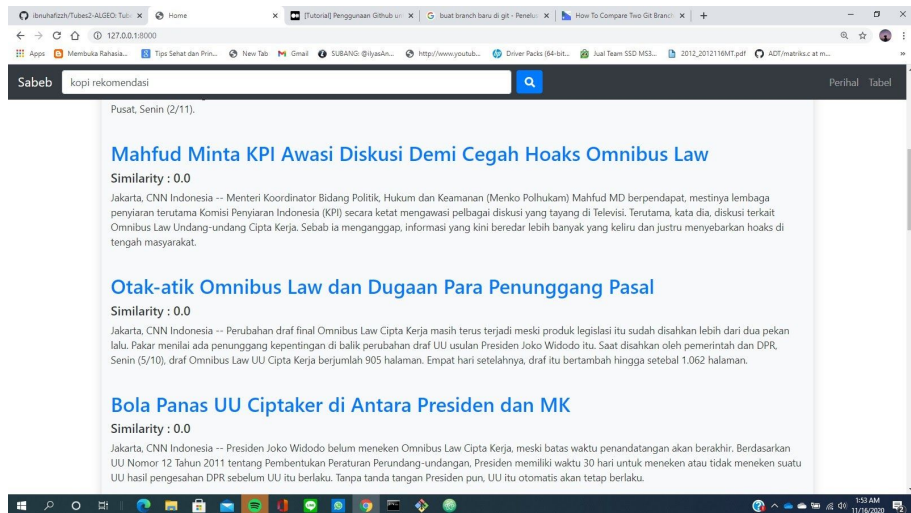
4. Test case dengan *query* kalimat yang tidak berkaitan dengan isi artikel



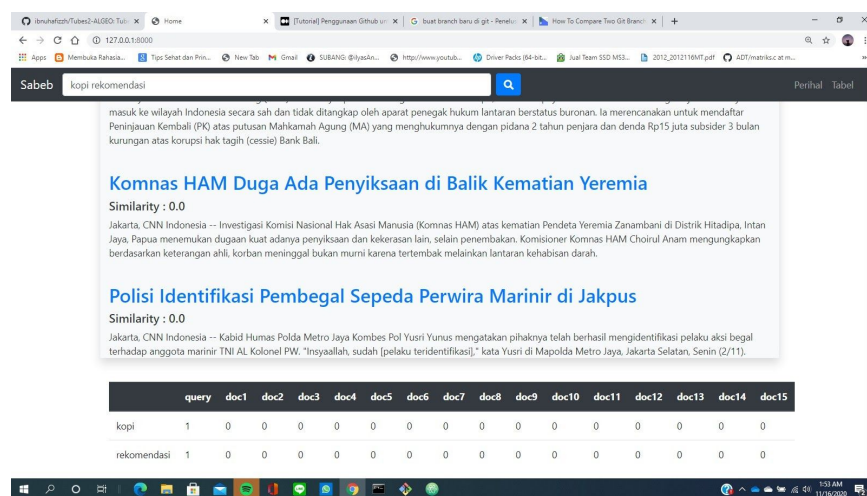
Jokowi: Gelombang Kedua Covid di Eropa, Kita Jangan Teledor
Similarity : 0.0
Jakarta, CNN Indonesia -- Presiden Joko Widodo meminta jajarannya mewaspadai lonjakan kedua gelombang virus covid-19 yang terjadi di Eropa. Jokowi meminta agar semua pihak tak bersikap teledor yang mengakibatkan lonjakan kedua gelombang covid-19 juga terjadi di Indonesia.

Sambut Rizieq Pulang dari Arab, FPI Cs Akan Gelar Rapat Akbar
Similarity : 0.0
Jakarta, CNN Indonesia -- Ormas-ormas Islam seperti Front Pembela Islam (FPI), PA 212 hingga Gerakan Nasional Pengawal Fatwa Ulama (GNPF-U) akan menggelar Aksi Bela Islam dan rapat akbar persiapan penyambutan Imam Besar FPI, Muhammad Rizieq Shihab di Halaman Gedung Sate, Kota Bandung, Jawa Barat, Rabu (4/11). Agenda itu diumumkan oleh salah satu orator Massa Aksi Bela Nabi Muhammad atau Aksi 211 di sekitar Kedutaan Besar Prancis, Jakarta.

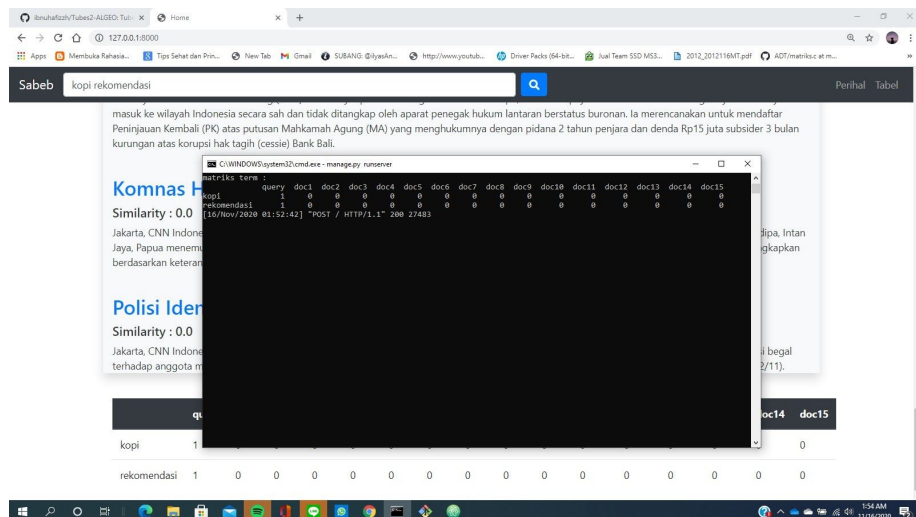
Gambar 4.14 Urutan Artikel Test Case 4.1



Gambar 4.15 Urutan Artikel Test Case 4.2



Gambar 4.16 Urutan Artikel Test Case 4.3 dan Tabel Term yang Sesuai dengan Query Test Case 4



Gambar 4.17 Hasil Dari Program Apabila Dijalankan Pada CMD

BAB 5

Kesimpulan, Saran, dan Refleksi

1. Kesimpulan

Dengan menerapkan bagian dari perkalian titik dua vektor, yaitu *cosine similarity*, dapat dibuat sebuah sistem temu-balik informasi sederhana atau yang biasa disebut dengan *search engine*. Nilai dari *cosine similarity* setiap dokumen yang ada pada *dictionary* dapat dijadikan acuan sebagai penentu apakah dokumen tersebut relevan dengan *query* atau *search key* yang diberikan atau tidak. Penggunaan *search engine* sangat memudahkan dalam mencari suatu dokumen yang sesuai dengan apa yang kita cari.

2. Saran

Kami sebagai penulis sadar bahwa program *search engine* sederhana yang telah kami buat masih jauh dari kata sempurna. Oleh karena itu, agar program dapat menjadi lebih baik dan maksimal, kami sebagai penulis memberikan saran sebagai berikut:

- a. Digunakannya komentar dengan maksimal pada program agar program dapat lebih dimengerti
- b. Digunakannya fungsi atau prosedur dalam program agar jalannya program dapat lebih efisien
- c. Penggunaan loop pada bagian program yang akan memproses seluruh dokumen agar jalannya program dapat lebih efisien

3. Refleksi

Dengan mengerjakan tugas besar kedua ini, kami diajarkan bagaimana melakukan manajemen waktu dengan baik. Selain itu kami juga diajarkan bagaimana bekerja sama dengan baik dan memberikan kepercayaan kepada teman kelompok. Tidak hanya itu, dengan mengerjakan tugas besar kedua ini, kami menjadi lebih mengetahui pengaplikasian perkalian titik dua buah vektor pada kehidupan sehari-hari.

REFERENSI

Referensi Laporan:

1. <http://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/2020-2021/Algeo-11-Vektor-di-Ruang-Euclidean-Bag2.pdf>
2. <http://informatika.stei.itb.ac.id/~rinaldi.munir/AljabarGeometri/2020-2021/Algeo-12-Aplikasi-dot-product-pada-IR.pdf>

Referensi Code:

1. <https://stackoverflow.com>
2. <https://pypi.org/project/Sastrawi/>
3. <https://www.w3schools.com/python/>
4. <https://link.medium.com/yEtxO932Kab>
5. https://medium.com/@saivenkat_/implementing-countvectorizer-from-scratch-in-python-exclusive-d6d8063ace22
6. <https://www.youtube.com/c/KelasTerbuka/>