LAPORAN TUGAS BESAR 2

Mata Kuliah Aljabar Linear dan Geometri IF2123

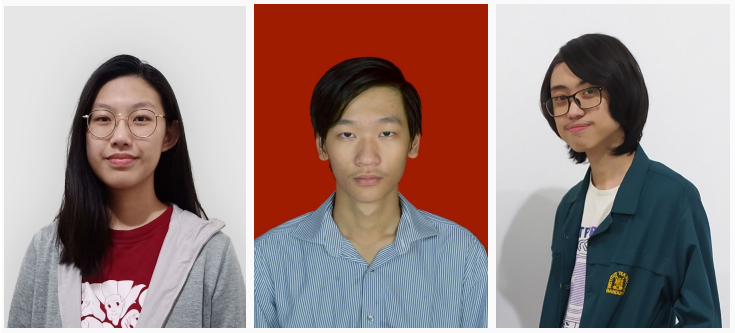
Dosen Pengampu :

Dr. Ir. Rinaldi Munir, M.T.

Nugraha Priya Utama S.T., M.A., Ph.D.

Dr. Judhi Santoso M.Sc.

Ir. Rila Mandala M.Eng., Ph.D.



Disusun Oleh Anggota Kelompok 69:

Jeane Mikha Erwansyah (13519116)

Fransiskus Febryan Suryawan (13519124)

Jeremia Axel Bachtera (13519188)

PROGRAM STUDI TEKNIK INFORMATIKA

SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA

INSTITUT TEKNOLOGI BANDUNG

2020

BAB I
DESKRIPSI MASALAH

Hampir semua dari kita pernah menggunakan search engine, seperti google, bing dan yahoo! search. Setiap hari, bahkan untuk sesuatu yang sederhana kita menggunakan mesin pencarian. Tapi, pernahkah kalian membayangkan bagaimana cara search engine tersebut mendapatkan semua dokumen kita berdasarkan apa yang ingin kita cari?

Sebagaimana yang telah diajarkan dalam kuliah pada materi vektor di ruang Euclidean, temu-balik informasi (*information retrieval*) merupakan proses menemukan kembali (*retrieval*) informasi yang relevan terhadap kebutuhan pengguna dari suatu kumpulan informasi secara otomatis. Biasanya, sistem temu balik informasi ini digunakan untuk mencari informasi pada informasi yang tidak terstruktur, seperti laman web atau dokumen.

Ide utama dari sistem temu balik informasi adalah mengubah search query menjadi ruang vektor Setiap dokumen maupun query dinyatakan sebagai vektor $w = (w_1, w_2, ..., w_n)$ di dalam R^n, dimana nilai w_i menyatakan jumlah kemunculan kata tersebut dalam dokumen (*term frequency*). Penentuan dokumen mana yang relevan dengan search query dipandang sebagai pengukuran kesamaan (*similarity measure*) antara *query* dengan dokumen. Semakin mirip suatu vektor dokumen dengan vektor *query*, semakin relevan dokumen tersebut dengan *query*. Kesamaan tersebut dapat diukur dengan *cosine similarity* dengan rumus:

$$sim(Q, D) = \cos\theta = \frac{Q \cdot D}{|Q||D|}$$

BAB II
TEORI SINGKAT

Information retrieval (temu balik informasi) adalah proses mencari data tidak terstruktur yang memenuhi kebutuhan informasi dari koleksi yang besar. Data tidak terstruktur adalah data yang tidak memiliki struktur yang jelas. Sistem temu balik informasi dapat dibedakan menjadi tiga kelompok besar menurut skala operasi. Dalam *web search*, sistem harus mencari dari sangat banyak dokumen yang tersebar pada banyak komputer. Di sisi lain, *personal information retrieval* bekerja dalam komputer personal untuk memenuhi kebutuhan pengguna individual. Di antara keduanya, ada *enterprise, institutional, and domain-specific search*, dimana temu-balik diberikan oleh dokumen internal yang disimpan dalam sistem terpusat.

Salah satu metode untuk melakukan temu-balik informasi adalah menggunakan vektor di ruang kosakata dokumen untuk menentukan kemiripan dokumen dengan pencarian yang diajukan. Kemiripan didefinisikan berdasarkan representasi *bag of words* yang berisi kata-kata independen, dan dikonversi ke suatu model ruang vektor yang memiliki dimensi sesuai dengan jumlah kata dalam *bag of words*.

Vektor merupakan suatu kuantitas yang memiliki besar dan arah. Pada ruang dua dimensi, vektor memiliki dua komponen yaitu dalam arah *i* dan dalam arah *j*. Sedangkan, pada ruang *n*-dimensi, vektor memiliki *n* komponen. Pada temu balik informasi, vektor digunakan sebagai penyimpan banyak kemunculan kata dari dokumen dan query sehingga nilai kemiripannya dapat ditentukan dengan memanfaatkan *dot product* atau disebut juga *cosine similarity*. Jadi dokumen atau query dengan *n* kata disimpan dalam bentuk vektor berukuran *n*, dengan besarannya bergantung pada kemunculan kata tersebut dalam dokumen atau query.

Istilah *cosine similarity* adalah tingkat kemiripan dengan dokumen. Besar dari *cosine similarity* dapat dicari dengan melakukan perkalian titik pada dua dokumen dalam model ruang vektor. Semakin besar nilainya, maka semakin mirip kedua dokumen yang diberikan. Namun, panjang vektor dokumen dapat memengaruhi hasil akhir dari perhitungan. Untuk mengatasi panjang dokumen yang dapat memengaruhi nilai kemiripan, maka hasil perkalian titik dibagi dengan panjang-panjang dokumen yang berkaitan.

BAB III
IMPLEMENTASI PROGRAM

Program ini dibagi menjadi dua bagian, yaitu bagian untuk pemrosesan dokumen, atau backend, dan bagian untuk menampilkan data kepada pengguna, atau frontend. Bagian backend dimuat dalam `server.js` yang menggunakan `vectorText.js` untuk membuat vektor dari dokumen serta memproses dokumen.

Komponen `vectorText.js` mengandung kelas `Vector` yang digunakan untuk membantu representasi data dalam model ruang vektor dokumen. Atribut yang digunakan dalam `Vector` adalah sebagai berikut:

1. **`vals`**: array yang menyimpan komponen-komponen vektor.

Sedangkan fungsi yang diimplementasikan dalam kelas `Vector` adalah sebagai berikut:

1. **`getComponent`**, **`setComponent`**: fungsi *getter/setter* untuk **`vals`**.
2. **`Length`**: fungsi *getter* untuk panjang vektor
3. **`Dot`**: fungsi untuk melakukan operasi perkalian titik vektor dengan vektor lain
4. **`cosineSimilarity`**: fungsi untuk menghitung *cosine similarity* vektor dengan vektor lain

Implementasi perkalian dot dalam kelas `Vector` yang telah disebutkan adalah dengan mengiterasi setiap elemen dalam vektor yang dioperasikan. Dengan mengasumsikan bahwa elemen kosong adalah 0, maka perkalian titik dua vektor adalah penjumlahan dari perkalian komponen-komponen yang berkaitan.

Komponen `server.js` menangani komunikasi yang dilakukan dari dan ke frontend. Implementasi server menggunakan *framework* Koa. Di backend, ada beberapa rute yang dapat diakses oleh pengguna lewat frontend, beberapa di antaranya adalah `/`, `/search`, `/upload`, dan `/docs`. Rute `/docs` menyimpan berkas-berkas yang diunggah oleh pengguna. Rute `/search` dan `/upload` berkaitan dengan pengunggahan berkas dan pencarian kata dalam berkas yang dapat dilakukan oleh pengguna.

BAB IV

EKSPERIMEN

Dokumen yang akan digunakan dalam eksperimen adalah :

1. Bagaimana Nasib Donald Trump Usai Turun dari Kursi Kepresidenan
2. Belum juga Respons Kemenangan Biden, Pakar Kim Jong Un Kecewa Berat Trump Kalah Pilpres AS
3. Diserang Roket dari Gaza, Israel Balas Tembaki Pos-pos Hamas
4. Festival Diwali, Joe Biden dan Kamala Harris Ucapkan Selamat di Twitter
5. Jarang Terjadi, 2 Panda Raksasa Terekam Kamera Berduaan di Siang Hari
6. Jerman Dakwa 12 Orang yang Berencana Bunuh dan Lukai Muslim Sebanyak-banyaknya
7. Jika Trump Menolak Pergi dari Gedung Putih, Bolehkah Militer AS Mengusirnya?
8. Joe Biden Menang Pilpres AS, Indonesia, China, dan Australia Beda Reaksi
9. Joe Biden Menang Pilpres AS, Taiwan Harap Hubungan Taipei Washington Tetap Terjalin
10. Kekurangan Polisi Minneapolis Siapkan Rp7 Miliar untuk Sewa Pasukan Lain
11. Kisah Nazi yang Mencuri Buku Masak dari Chef Yahudi
12. Konflik Ethiopia Meluas ke Luar Negeri, Roket Hantam Ibu Kota Eritrea
13. Orientasi Seksual Tidak Diterima di Indonesia, WNI Ini Cari Perlindungan di Australia
14. Pembuat Roti di Palestina Ini Pelihara 2 Ekor Anak Singa di Atap Rumah
15. Pengadilan Houthi, Yaman Hukum Mati 21 Mata-mata Koalisi Arab Saudi
16. Perusahaan Mobil VW Bantah Ada Kerja Paksa Etnik Uighur di Pabrik Xinjiang
17. PM Armenia Jadi Target Pembunuhan oleh Para Mantan Pejabat Dalam Negeri
18. Polisi Tangkap Pria Bugil yang Hantam Toko Kelontong dengan Mobil di AS
19. Putin Minta Azerbaijan Jaga Gereja dan Tempat Suci Kristen Peninggalan Armenia di Nagorno Karabakh
20. Ratapan Komunitas Penganut Mormon Setelah 3 Ibu dan Anak-anaknya Dibunuh di Gurun
21. Rumah Sakit Rujukan COVID 19 di Rumania Kebakaran dan Tewaskan 10 Orang
22. Selama 27 Tahun Keluarga Ini Tak Tahu Kerabatnya yang Hilang Tewas Kecelakaan
23. Suami Bunuh Istri dan Putrinya, Lalu Tidur Dengan Jenazah Korban Selama 7 Hari

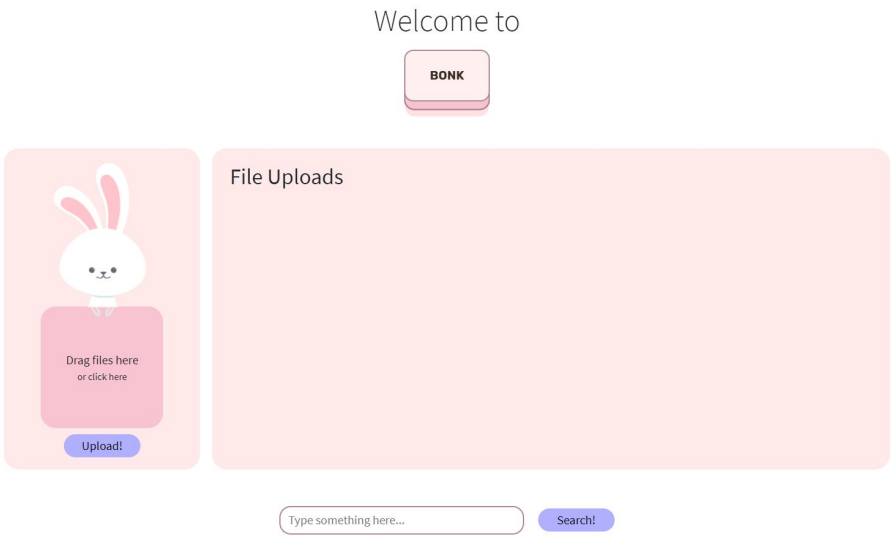
24. Turis Beri Review Jelek Di Penjara 2 Hari, Hotel Thailand Diperingatkan TripAdvisor

25. Unik! Masjid di Malaysia Dilengkapi Tempat Fitness, Ada Pelatihnya Juga

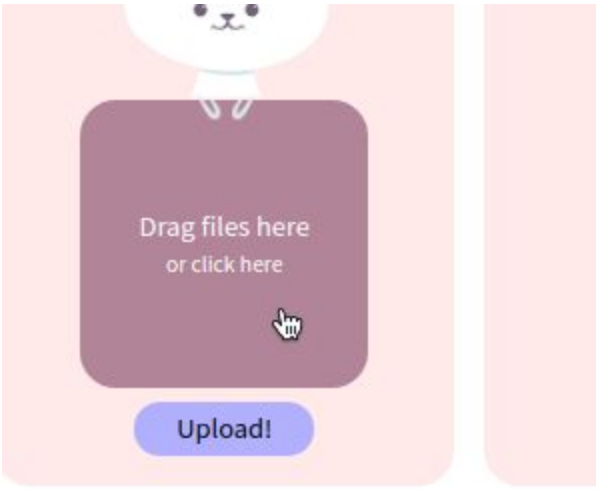
26. Video Viral Pria di China Bawa Harimau Jalan-jalan Rupanya Seekor Anjing yang Dicat

Semua dokumen tersebut diambil dari KOMPAS.com.

Dari semua dokumen di atas, akan dicari *query* "Joe Trump Pembuat Roti Bawa Anjing Jalan jalan Rupanya Kursi Presiden Putih".

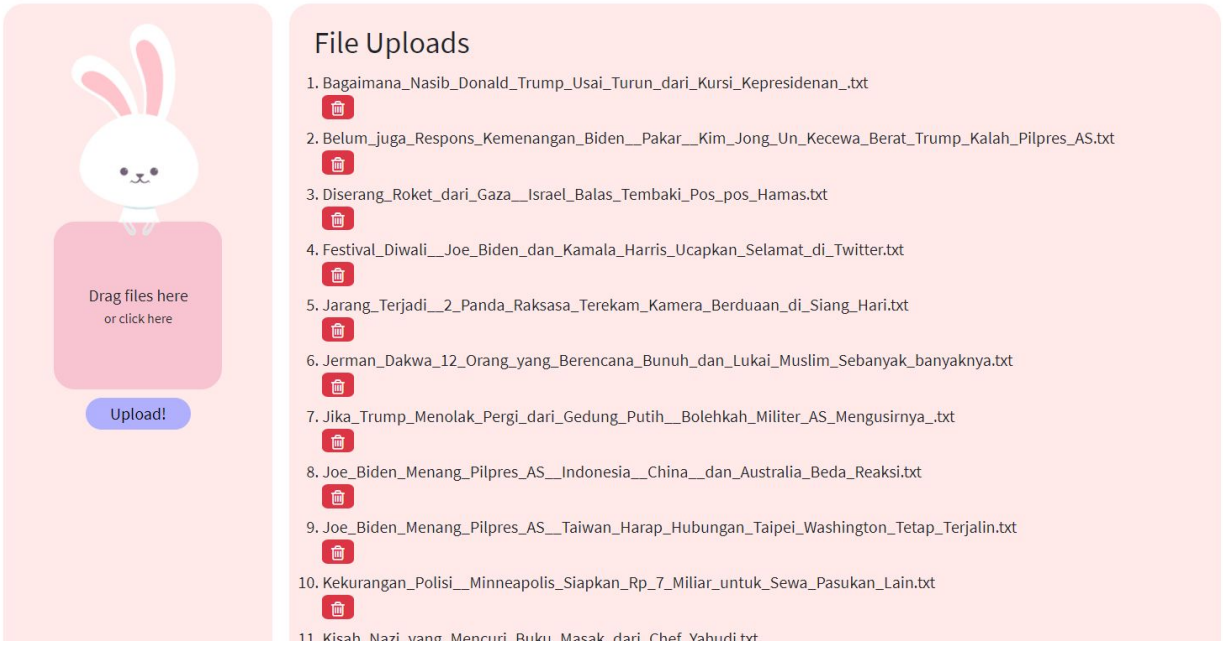


Gambar 1. Tangkap layar tampilan index.html



Gambar 2. *Cursor* sedang *hovering*

Pertama, semua dokumen diunggah dengan menekan tombol '*Drag files here or click here*'



Gambar 3. Tangkap layar index.html setelah mengunggah *file*

Gambar 4. Tangkap layar *query* yang sudah terisi

Kemudian masukkan *query* di kolom pencarian kemudian tekan tombol *Search!*.

Gambar 5. Tangkap layar hasil *query* (*Query Result*)

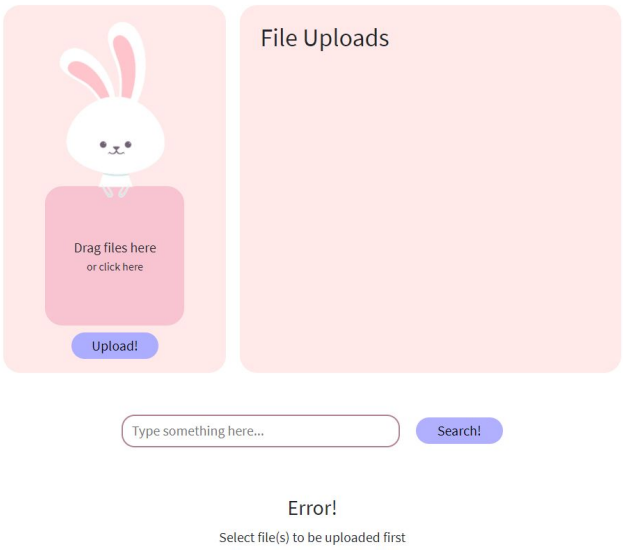
Hasilnya dokumen berjudul “Bagaimana Nasib Donald Trump Usai Turun dari Kursi Kepresidenan” merupakan dokumen yang paling mirip dengan tingkat kemiripan 32.07%.

Gambar 6. Tangkap layar *Query Result* dengan tingkat kemiripan tertinggi

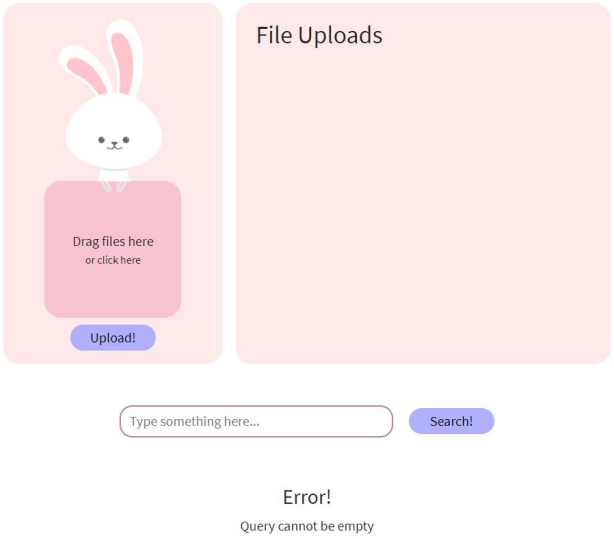
Term	query	Bagaimana_Nasib_Donald_Trump_...	Belum_juga_Respons_Kemenangan...	Diserang_Roket_dari_Gaza__Israel_..
presiden	1	8	7	0
trump	1	6	8	1
putih	1	3	1	0
joe	1	1	2	0
buat	1	1	0	0
jalan	2	0	0	0
roti	1	0	0	0
bawa	1	0	0	0
anjing	1	0	0	0
kursi	1	0	0	0

Gambar 7. Tangkap layar tabel yang berisi jumlah kata dari *query*, dan semua dokumen unggahan

Dari hasil percobaan menggunakan *query* “Joe Trump Pembuat Roti Bawa Anjing Jalan jalan Rupanya Kursi Presiden Putih” menghasilkan vektor berdasarkan *query [1, 1, 1, 1, 1, 2, 1, 1, 1, 1]* dan dokumen ”Bagaimana Nasib Donald Trump …” menghasilkan vektor *[8, 6, 3, 1, 1, 0, 0, 0, 0, 0]* sedangkan dokumen “Belum juga Respons Kemenangan …” menghasilkan vektor *[7, 8, 1, 2, 0, 0, 0, 0, 0, 0]*. Jika dihitung antara vektor *query* dengan vektor dokumen pertama menggunakan cosine similarity akan didapat nilai kemiripan 32.07% sedangkan penghitungan vektor *query* dengan dokumen kedua akan didapat nilai kemiripan 31.45%. Sehingga dokumen pertama lebih mirip dengan *query* yang dicari dibandingkan dokumen lainnya.



Gambar 8. Tangkap layar situs apabila tidak ada *file* yang diunggah



Gambar 9. Tangkap layar situs apabila *query* tidak diisi

Kedua gambar tersebut menampilkan layar pengguna apabila pengguna tidak mengunggah berkas dan/atau mengisi *query*.

BAB V
SIMPULAN, SARAN, DAN REFLEKSI

Hasil dari pengerjaan tugas ini adalah suatu program berbasis web yang dapat menghitung *cosine similarity* dokumen dengan kata kunci pencarian pengguna. Program dapat menerima *file* dokumen berupa berkas teks (.txt) serta *query* dari pengguna dan mengembalikan kemiripan *query* dengan berkas-berkas yang diberikan. Berkas-berkas dan *query* diolah terlebih dahulu, yaitu dengan melakukan *stemming* dan membersihkan tanda baca. Program memberikan jumlah kata dari setiap dokumen serta kemiripan *query* dengan masing-masing dokumen. Hasil dari *query* pengguna ditampilkan secara berurut berdasarkan tingkat kemiripannya. Selain itu, tabel frekuensi perkata juga ditampilkan di bawah hasil *query*.

Saran untuk pengembangan sistem temu balik informasi ini adalah membuat sistem yang juga dapat memberikan hasil *query* meskipun pengetikan kata-kata dalam *query* terdapat kesalahan penulisan atau *typological error*. Jika pengguna salah menuliskan kata-kata ke dalam *query* dapat ditampilkan kata yang lebih tepat atau saran kata yang berhubungan. Dengan demikian mesin pencari ini dapat lebih efektif dalam melakukan pencarian. Saran lain adalah membuat kode situs dengan rapi sehingga lebih mudah dibaca dan dimengerti. Jika ada *programmer* lain yang ingin mengembangkan situs lebih lanjut, *programmer* tersebut dapat melakukannya dengan lebih mudah. Saran lain adalah untuk membuat situs dengan UI (*User Interface*) yang lebih modern dan *user friendly* sehingga UX (*User Experience*) lebih baik.

Refleksi dari tugas ini adalah betapa banyaknya manfaat dari vektor. Meskipun di dalam dunia analog, vektor di atas ruang tiga tidak dapat digambarkan, nyatanya vektor di ruang-n dapat dimanfaatkan untuk berbagai hal. Selain itu, sebaiknya kami mulai mempersiapkan dan melatih diri menggunakan GitHub (untuk menghindari dan mengatasi *conflict*), dan mencoba *Operating System* (OS) lain karena salah satu anggota kelompok kami ketika meng-*compile* server menggunakan OS yang berbeda, waktu yang diperlukan sangat lama.

BAB VI
DAFTAR REFERENSI

Karyono, G., & Utomo, F. S. (2012). Temu Balik Informasi pada Dokumen Teks Berbahasa Indonesia dengan Metode Vector Space Retrieval Model. *Semantik*, *2*(1).

Manning, C. D., Schütze, H., & Raghavan, P. (2008). *Introduction to information retrieval*. Cambridge university press.