LAPORAN PROYEK

IMPLEMENTASI SISTEM TEMU KEMBALI INFORMASI MENGGUNAKAN *BOOLEAN RETRIEVAL* DAN *VECTOR SPACE MODEL* (*TF-IDF*) BERBASIS STREAMLIT

Disusun Oleh:

NIM : A11.2023.14979

Nama : Suluh Yoga Pratama

Kelompok : A11.4703

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS ILMU KOMPUTER

2025

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

BAB I
PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi informasi telah menghasilkan jumlah data digital yang sangat besar, baik dalam bentuk dokumen teks, artikel berita, jurnal ilmiah, laporan organisasi, hingga posting media sosial. Informasi tersebut terus bertambah setiap detik, sehingga manusia memerlukan sistem yang dapat membantu menemukan informasi yang relevan secara cepat dan tepat. Bidang yang mempelajari bagaimana mesin melakukan pencarian informasi tersebut dikenal sebagai *Information Retrieval* (IR) atau Sistem Temu Kembali Informasi (STKI).

Berbeda dengan *database retrieval* yang hanya bekerja pada data terstruktur dan menuntut kecocokan query secara presisi (*exact matching*), pada IR pencarian dilakukan pada teks yang tidak terstruktur. Tujuan pencarian bukan hanya menemukan dokumen yang mengandung kata yang sama, tetapi dokumen yang relevan berdasarkan makna dan konteks. Oleh karena itu, IR menggunakan teknik seperti *indexing, ranking*, pembobotan kata (*term weighting*), serta model matematis untuk mengukur kesesuaian antara dokumen dan *query*. Dengan demikian, STKI menjadi komponen penting pada berbagai aplikasi modern, seperti *search engine*, rekomendasi akademik, chatbot, sistem tanya jawab, dan analisis teks.

1.2 Tujuan

Tujuan dari proyek ini antara lain:

1. Menyusun *corpus* berjumlah 50 dokumen dalam bahasa Indonesia.
2. Melakukan *text preprocessing* untuk membersihkan, menormalkan, dan menyederhanakan bentuk kata.
3. Membangun *Boolean Retrieval Model* untuk pencarian berdasarkan logika boolean.
4. Menerapkan *Vector Space Model (TF-IDF + Cosine Similarity)* untuk pencarian berbasis peringkat.
5. Mengembangkan sistem antarmuka pencarian berbasis web menggunakan Streamlit.
6. Mengevaluasi performa sistem menggunakan *precision, recall, P@k, AP,* dan *nDCG*.

1.3 Ruang Lingkup

Ruang lingkup proyek ini terbatas pada pemrosesan dokumen teks berbahasa Indonesia, tanpa tambahan fitur seperti *semantic search, synonym expansion*, atau *learning-to-rank.* Model yang digunakan memfokuskan pada dasar-dasar IR, yaitu *Boolean* dan *VSM*.

1.4 Keterkaitan dengan Sub-CPMK

Tabel 1. Keterkaitan dengan Sub-CPMK

Sub-CPMK	Capaian Pembelajaran	Implementasi di Proyek
10.1.2	Menerapkan *text preprocessing*	File preprocess.py
10.1.3	Mengimplementasikan *Boolean IR* & *VSM*	File boolean_ir.py dan vsm_ir.py
10.1.4	Melakukan evaluasi sistem IR	File eval.py

BAB II
DATA DAN PREPROCESSING

2.1 Sumber Data

Dataset yang digunakan dalam proyek ini terdiri dari 50 dokumen teks yang disusun secara mandiri. Setiap dokumen memiliki panjang 3 paragraf dan bertema mengenai Sistem Informasi, pencarian informasi, teknologi digital, serta dasar-dasar manajemen data. Tema dipilih agar konsisten dan mendukung pembentukan *corpus* yang homogen sehingga model dapat bekerja optimal. Dokumen awal disimpan dalam folder "data/raw/". Format yang digunakan adalah ".txt" untuk mempermudah pemrosesan berbasis string.

2.2 Peran & Tahapan *Preprocessing*

Teks mentah biasanya mengandung variasi bentuk kata, huruf kapital, tanda baca, dan kata-kata umum yang tidak memberikan makna signifikan. Jika langsung diproses, hal ini dapat menyebabkan:

1. Pembentukan indeks yang besar dan redundan
2. Perhitungan TF-IDF menjadi bias
3. Skor kemiripan antar dokumen tidak akurat

Karena itu, dilakukan *text preprocessing* untuk menyederhanakan representasi kata sehingga dokumen memiliki bentuk normal yang dapat dibandingkan secara adil.

Tabel 2. Tahap Preprocessing

Tahap	Deskripsi	Contoh
Case folding	Mengubah seluruh huruf menjadi huruf kecil	"Sistem" → "sistem"

Tokenization	Memecah teks menjadi unit kata	“sistem informasi modern” → [sistem, informasi, modern]
Stopword removal	Menghapus kata tidak bermakna penting	“dan”, “yang”, “adalah”, …
Stemming	Mengubah kata ke akar katanya	“berjalan”, “berjalannya” → “jalan”

Proses *stemming* dilakukan menggunakan library Sastrawi, yang memang dirancang untuk morfologi Bahasa Indonesia.

2.3 Contoh Sebelum & Sesudah Preprocessing

Tabel 3. Contoh Preprocessing

Before	***After***
“Sistem Informasi adalah bidang yang mempelajari teknologi komputer.”	sistem informasi bidang pelajar teknologi komputer

2.4 10 Token Paling Sering Muncul

Gambar 1. 10 Token Paling Sering Muncul

BAB III
METODE SISTEM TEMU KEMBALI

3.1 Pemodelan Dokumen

Setiap dokumen diubah menjadi representasi vektor. Ide utamanya adalah bahwa sebuah dokumen dapat dianggap sebagai kumpulan kata yang memiliki bobot tertentu. Semakin penting sebuah kata, semakin tinggi bobotnya dalam dokumen tersebut.

3.2 *Boolean Retrieval Model*

Model ini menggunakan logika boolean (AND, OR, NOT). Contoh interpretasi:

1. informasi *AND* sistem → dokumen yang mengandung kedua kata tersebut.

2. indeks *OR* vektor → dokumen yang mengandung salah satu kata.
3. *NOT* evaluasi → dokumen yang tidak mengandung kata tersebut.

Model ini menghasilkan output set dokumen, tanpa ranking.

3.3 *Vector Space Model (VSM)*

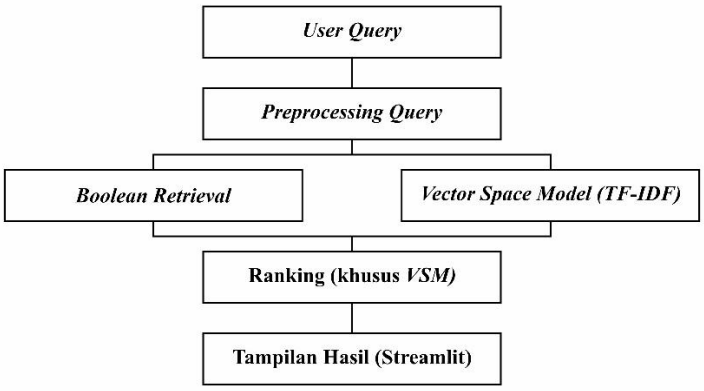
Pada VSM, dokumen direpresentasikan dalam bentuk vektor numerik dan output berupa ranking dokumen berdasarkan skor relevansi.

Tabel 4. Rumus Vector Space Model

Rumus TF	$TF = \frac{term\ count}{total\ terms}$
Rumus IDF (*Smoothed*)	$IDF = \log\left(\frac{\mathrm{N}+1}{df+1}\right)+1$
Rumus TF-IDF	$TF - IDF = TF \times IDF$
Rumus *Cosine Similarity*	$similarity = \frac{\sum(q_i\ .\ d_i)}{\sqrt{\sum(q_i^2)} \times \sqrt{\sum(d_i^2)}}$

BAB IV
ARSITEKTUR SISTEM

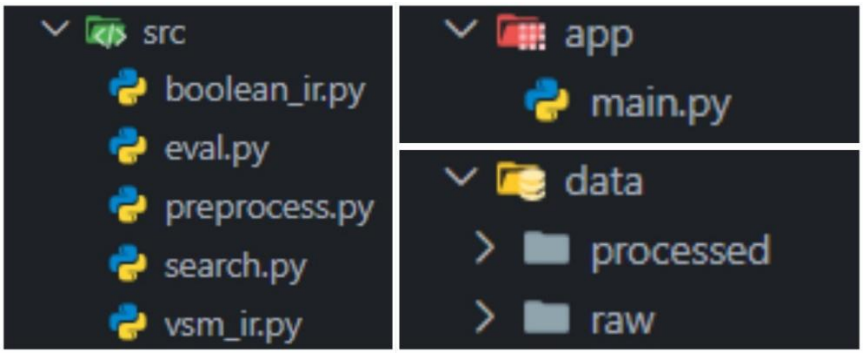
4.1 Diagram Alur Sistem



Gambar 2. Diagram Alur Sistem

Diagram alur menunjukkan bagaimana *query* diproses. Pertama, pengguna memasukkan teks pencarian. Query tersebut kemudian diproses melalui tahapan *preprocessing* yang sama dengan dokumen. Sistem kemudian memilih model pencarian sesuai pilihan pengguna (Boolean atau VSM). Jika menggunakan Boolean, sistem mengembalikan daftar dokumen yang cocok. Jika menggunakan VSM, sistem memberikan daftar peringkat berdasarkan skor *cosine similarity*. Hasil akhirnya ditampilkan melalui antarmuka Streamlit.

4.2 Struktur Folder



Gambar 3. Struktur Folder

Struktur folder menunjukkan pemisahan kode berdasarkan fungsinya:

1. src/ berisi logika utama sistem.
2. app/ berisi file antarmuka pengguna.
3. data/ menyimpan dokumen mentah (*raw*) dan hasil preprocessing (*processed*).

Pemilahan ini mendukung keterbacaan, perawatan kode, dan pengembangan lebih lanjut.

BAB V
EKSPERIMEN DAN EVALUASI

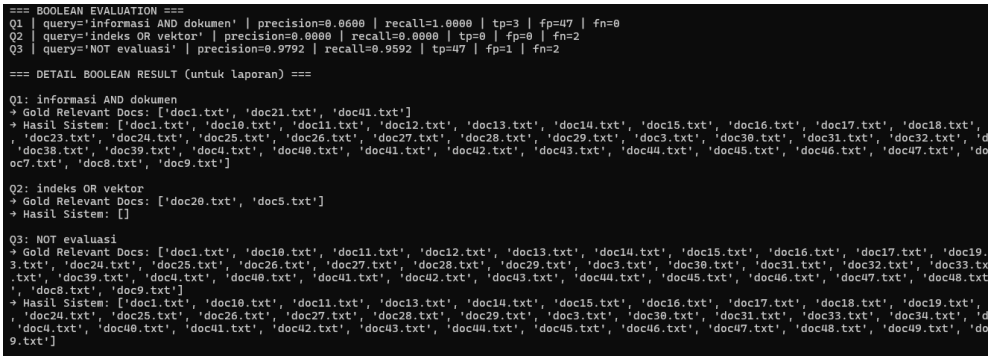
5.1 Skenario Pengujian dan Gold Relevant Set

Pada tahap pengujian, dilakukan evaluasi menggunakan tiga query utama yang telah dirancang untuk mewakili tiga operasi dasar dalam model Boolean, yaitu *AND*, *OR*, dan *NOT*. Selain itu, juga disusun *gold relevant set* (dokumen relevan menurut ahli/ground truth) untuk masing-masing query sebagai acuan pembandingan hasil sistem.

Tabel 5. Query Pengujian

Kode	Query	Gold Relevant Docs	Tujuan
Q1	informasi *AND* dokumen	{doc1.txt, doc21.txt, doc41.txt}	Menguji operasi *AND*
Q2	indeks *OR* vektor	{doc20.txt, doc5.txt}	Menguji operasi *OR*
Q3	*NOT* evaluasi	Semua dokumen kecuali dokumen yang mengandung kata evaluasi	Menguji operasi *NOT*

5.2 Evaluasi *Model Boolean*



Gambar 4. Pengujian Boolean Model

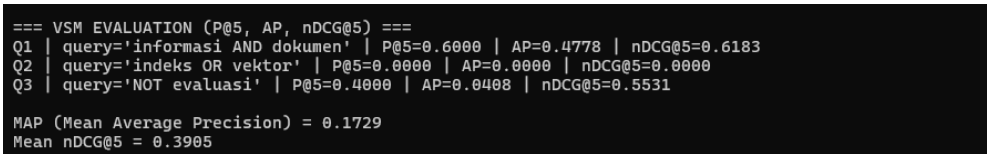
Tabel 6. Hasil Evaluasi Boolean Model

Query	***Precision***	***Recall***	**TP**	**FP**	**FN**
Q1	0.06	1.00	3	47	0
Q2	0.00	0.00	0	0	2
Q3	0.9792	0.9592	47	1	2

Analisis:

1. Pada Q1, *recall* yang tinggi (1.0) menunjukkan bahwa seluruh dokumen relevan berhasil ditemukan, namun *precision* rendah karena sistem mengembalikan banyak dokumen tidak relevan. Ini menunjukkan model boolean cenderung menghasilkan *over-retrieval*.
2. Pada Q2, baik *precision* maupun *recall* bernilai 0, menandakan istilah yang dicari tidak kuat termuat dalam koleksi dokumen atau hasil pencarian tidak tepat.
3. Pada Q3, *precision* sangat tinggi menunjukkan hasil pencarian NOT evaluasi sangat tepat, karena sistem mampu mengecualikan dokumen dengan benar.

5.3 Evaluasi *Vector Space Model (VSM)*



Gambar 5. Pengujian VSM

Tabel 7. Hasil Evaluasi VSM

Query	***P@5***	***AP***	***nDCG@5***
Q1	0.60	0.4778	0.6183
Q2	0.00	0.00	0.00

Q3	0.40	0.408	0.5531

Interpretasi:

1. Pada Q1, sistem mampu menempatkan dokumen relevan pada posisi atas hasil pencarian (dibuktikan *P@5* dan *nDCG@5* cukup baik).
2. Pada Q2, hasil serupa dengan boolean model kesulitan memahami makna kata indeks dan vektor, kemungkinan karena frekuensi kemunculan kata rendah.
3. Pada Q3, VSM berhasil mendeteksi kemiripan konten secara cukup baik sehingga dokumen relevan tetap muncul di peringkat atas.

5.4 Analisis Perbandingan Model

Tabel 8. Perbandingan Boolean Model dan VSM

Aspek	***Boolean Model***	***VSM***
Prinsip Kerja	Cocok/Tidak cocok (*exact match*)	Kesamaan bobot kata (*cosine similarity*)
Kelebihan	Sederhana dan cepat	Relevansi dokumen lebih terukur
Kekurangan	Sensitif terhadap kata yang tidak sesuai	Perlu proses bobot & komputasi lebih tinggi
Cocok Untuk	Pencarian deterministik yang jelas	Pencarian dokumen yang beragam & kontekstual

Kesimpulan:

1. Boolean baik dalam filtering, tetapi kualitas hasil sangat bergantung pada kata kunci.
2. VSM lebih baik dalam ranking, sehingga lebih sesuai untuk pencarian informasi yang kompleks.

BAB VI
DISKUSI

6.1 Kelebihan

1. Sistem mendukung dua paradigma *Information Retrieval*, yaitu *Boolean Retrieval* dan *Vector Space Model*, sehingga memungkinkan pengguna melakukan pencarian baik secara *exact matching* maupun berbasis tingkat relevansi.
2. Tahap *preprocessing* telah disesuaikan dengan karakteristik Bahasa Indonesia, termasuk *stopword removal* dan *stemming* menggunakan *Sastrawi*, yang meningkatkan kualitas representasi dokumen.

3. Sistem dirancang dengan struktur kode yang modular dan sederhana, sehingga mudah dipahami serta dapat dikembangkan lebih lanjut sesuai kebutuhan penelitian atau implementasi sistem nyata.

6.2 Keterbatasan

1. Sistem masih bekerja pada tingkat kesamaan kata (*lexical matching*) sehingga tidak mampu memahami sinonim, variasi kata, atau konteks semantik.
2. Sistem belum menyediakan mekanisme relevance *feedback*, sehingga pengguna tidak dapat memberi penilaian untuk meningkatkan kualitas hasil pencarian.
3. Model pembobotan TF-IDF tidak mempertimbangkan hubungan antar kata dan struktur kalimat sehingga ranking dokumen masih bersifat permukaan (*surface-level relevance*).

6.3 Saran Pengembangan

1. Mengganti metode pembobotan dari TF-IDF menjadi *BM25* untuk meningkatkan ketepatan skor relevansi.
2. Mengintegrasikan model word embeddings seperti *FastText, Word2Vec,* atau *BERT* untuk menangkap makna semantik antar kata.
3. Menambahkan fitur *highlighting* pada hasil pencarian agar pengguna dapat langsung melihat bagian kalimat yang relevan dalam dokumen.

BAB VII
KESIMPULAN

Sistem temu kembali informasi berbasis *Boolean Retrieval* dan *Vector Space Model* berhasil dibangun dan diimplementasikan dengan menggunakan Streamlit sebagai antarmuka pencarian. Proses *preprocessing* yang meliputi *case folding, tokenization, stopword removal,* dan *stemming* telah berhasil meningkatkan kualitas representasi teks sehingga dokumen dapat diolah dalam bentuk yang lebih terstruktur dan sesuai untuk proses pencarian. Model Boolean mampu melakukan pencarian berbasis kecocokan logika, sementara VSM melakukan pemeringkatan berdasarkan kemiripan vektor antar dokumen dan *query*.

Evaluasi kinerja sistem menggunakan metrik *precision, recall, Precision@k, Average Precision,* dan *nDCG* menunjukkan bahwa sistem mampu menemukan dokumen relevan meskipun hasil pemeringkatan belum sepenuhnya optimal. Rendahnya nilai beberapa metrik menunjukkan bahwa sistem masih terbatas dalam memahami variasi bahasa dan konteks semantik kata. Oleh karena itu, proses temu kembali masih bersifat berbasis kecocokan kata (*bag-of-words*) tanpa mempertimbangkan kedekatan makna antar kata.

Melalui rangkaian implementasi dan pengujian ini, seluruh capaian pembelajaran mata kuliah telah terpenuhi. Sub-CPMK 10.1.1 tercapai melalui perumusan konsep dasar STKI dan perancangan arsitektur sistem temu kembali informasi. Sub-CPMK 10.1.2 tercapai melalui penerapan tahapan *preprocessing*

dokumen menggunakan metode pembersihan teks, tokenisasi, penghilangan *stopword*, dan *stemming*. Selanjutnya, Sub-CPMK 10.1.3 tercapai melalui implementasi dua model temu kembali (Boolean dan VSM) serta pelaksanaan evaluasi performa sistem menggunakan berbagai metrik pengukuran relevansi. Dengan demikian, proyek ini berhasil mengintegrasikan pemahaman konseptual dan keterampilan teknis dalam membangun sistem temu kembali informasi berbasis teks.

Dengan demikian, proyek ini tidak hanya memberikan pengalaman langsung dalam mengimplementasikan Sistem Temu Kembali Informasi, tetapi juga memperkuat pemahaman mengenai bagaimana model pencarian bekerja di dunia nyata, khususnya dalam memproses dan menemukan informasi yang relevan dari kumpulan dokumen yang besar. Sistem ini dapat menjadi dasar untuk pengembangan lebih lanjut ke arah pencarian berbasis semantik atau integrasi dengan teknologi NLP modern.