LAPORAN PROYEK

PENERAPAN MODEL BOOLEAN RETRIEVAL DAN VECTOR SPACE MODEL (TF-IDF) PADA SISTEM TEMU KEMBALI INFORMASI BERBASIS STREAMLIT

Disusun Oleh:

NIM : A11.2023.14986

Nama : Igdo Ragil Manuel

Kelompok : A11.4703

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS ILMU KOMPUTER

2025

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

BAB I
PENDAHULUAN

1.1 Latar Belakang

Di era digital saat ini, jumlah informasi lowongan pekerjaan (loker) yang tersedia secara online meledak. Informasi ini seringkali tidak terstruktur, tersebar di berbagai platform, dan ditulis dalam format yang berbeda-beda. Bagi pencari kerja, menemukan lowongan yang paling relevan dengan kualifikasi, lokasi, dan tipe pekerjaan (misal: WFH, WFO, Magang) menjadi sebuah tantangan besar.

Sistem Temu Kembali Informasi (STKI) atau Information Retrieval (IR) adalah disiplin ilmu yang mempelajari cara mengambil informasi relevan dari sekumpulan data tidak terstruktur. Berbeda dari pencarian database (SQL) yang menuntut kueri terstruktur, STKI dirancang untuk memahami kueri bahasa alami (seperti "magang di semarang") dan memberikan hasil yang diurutkan berdasarkan relevansi.

Proyek ini bertujuan merancang dan mengimplementasikan sebuah mesin pencari mini khusus untuk korpus lowongan kerja fiktif di area Semarang. Sistem ini akan menerapkan dua model STKI fundamental: Boolean Retrieval Model untuk pencarian presisi berbasis kata kunci, dan Vector Space Model (VSM) dengan pembobotan TF-IDF untuk pencarian relevansi berbasis ranking.

1.2 Tujuan

Tujuan dari proyek ini antara lain:

1. Menyusun korpus mini berisi 15 dokumen teks lowongan kerja di Semarang.
2. Mengimplementasikan modul text preprocessing yang mencakup case-folding, tokenization, stopword removal, dan stemming Sastrawi.
3. Membangun Inverted Index dan Boolean Retrieval Model untuk memproses kueri dengan operator AND, OR, dan NOT.
4. Menerapkan Vector Space Model (VSM) dengan pembobotan TF-IDF dan Cosine Similarity untuk melakukan perankingan dokumen.
5. Mengembangkan antarmuka pengguna (UI) berbasis web menggunakan Streamlit yang dapat menjalankan kedua model tersebut.
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

Korpus data yang digunakan terdiri dari 15 dokumen teks (.txt) yang dibuat secara manual (dummy). Tema dari korpus ini adalah "Lowongan Kerja dan magang di Area Semarang dan Sekitarnya", yang mencakup berbagai jenis pekerjaan seperti Magang, Part-time, Full-time, WFH, dan WFO. Contoh nama file dalam korpus adalah doc01_magang_web_smg_tengah.txt dan doc07_part_time_admin_wfh.txt.

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 di Proyek
Case Folding	Mengubah semua huruf menjadi huruf kecil.	"Magang" → "magang"
Tokenization	Memecah kalimat menjadi unit-unit kata (token).	"magang web" → ['magang', 'web']
Stopword Removal	Menghapus stopwords (kata umum) dari NLTK yang digabung custom list (misal: 'wfo', 'skill', 'saya').	"['lokasi', 'wfo', 'di', 'kantor'] → ['lokasi', 'wfo', 'kantor']"
Stemming	Mengubah kata ke bentuk kata dasar menggunakan	"Dibutuhkan", "membantu" → "butuh", "bantu"

	Sastrawi.	

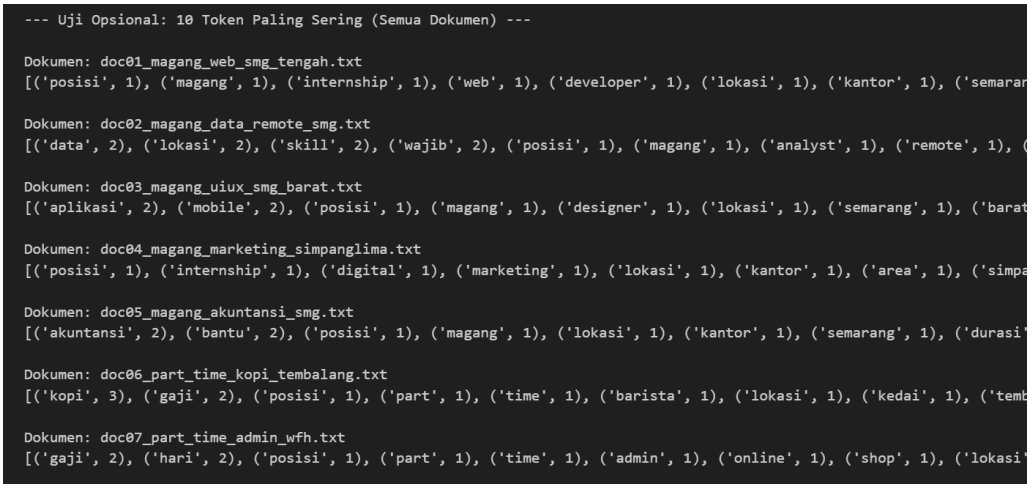
Proses *stemming* dilakukan menggunakan library Sastrawi, yang memang dirancang untuk morfologi Bahasa Indonesia.

2.3 Contoh Sebelum & Sesudah Preprocessing

Tabel 3. Contoh Preprocessing

Before	*After*
Info Magang (Internship) Web Developer. Dibutuhkan mahasiswa semester 5 UDINUS atau UNNES. Skill: PHP, Laravel, dan sedikit React. Lokasi: WFO di kantor Semarang Tengah, dekat Tugu Muda. Tugas: membantu tim backend, durasi 6 bulan.	info magang internship web developer butuh mahasiswa semester udinus unnes php laravel react kantor semarang tugu muda bantu tim backend.

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)*

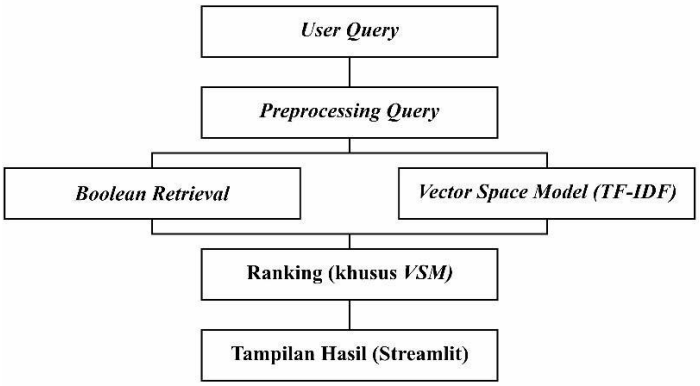
Pada VSM, dokumen direpresentasikan dalam bentuk vektor numerik dan output berupa ranking dokumen berdasarkan skor relevansi.

Tabel 4. Rumus Vector Space Model

Rumus TF	$TF = \frac{term\ count}{total\ terms}$
Rumus IDF (*Smoothed*)	$IDF = \log\left(\frac{N+1}{df+1}\right) + 1$
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

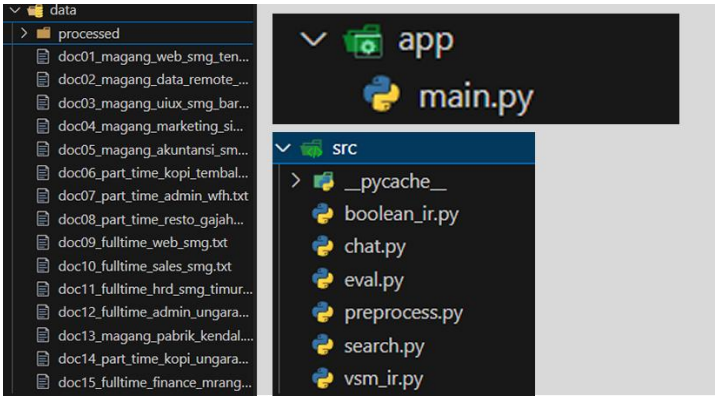
4.1 Diagram Alur Sistem



Gambar 2. Diagram Alur Sistem

Alur kerja sistem dimulai saat pengguna memasukkan kueri pencarian. Kueri ini akan dinormalisasi menggunakan pipeline preprocessing yang identik dengan yang digunakan pada dokumen. Setelah bersih, sistem akan menerapkan model pencarian yang dipilih oleh pengguna. Apabila pengguna memilih Model Boolean, sistem akan mengembalikan himpunan dokumen yang cocok secara eksak (tanpa urutan). Sebaliknya, jika pengguna memilih Vector Space Model (VSM), sistem akan menghitung skor cosine similarity untuk setiap dokumen, lalu menyajikan hasilnya sebagai daftar yang terurut berdasarkan relevansi. Seluruh hasil pencarian ini pada akhirnya ditampilkan kepada pengguna melalui antarmuka web Streamlit..

4.2 Struktur Folder



Gambar 3. Struktur Folder

Proyek ini dirancang dengan struktur folder modular untuk memisahkan antara logika, data, dan antarmuka pengguna (UI). Pemisahan ini sangat penting untuk mendukung keterbacaan kode (readability), kemudahan perawatan (maintainability), dan pengembangan lebih lanjut.

Struktur folder utama dan fungsinya adalah sebagai berikut:

1. src/: Berisi seluruh script Python yang menangani logika inti sistem. Ini mencakup modul untuk preprocessing (preprocess.py), implementasi model (vsm_ir.py), evaluasi (eval.py), serta script eksekusi command-line (search.py, chat.py).
2. app/: Berisi script antarmuka pengguna (UI) berbasis web. File main.py di dalamnya menggunakan library Streamlit untuk menyediakan visualisasi dan interaksi dengan pengguna.
3. data/: Berfungsi sebagai repositori korpus. Folder ini berisi dokumen-dokumen mentah (15 file doc*.txt) yang digunakan sebagai korpus, dan juga berisi sub-folder processed/ yang menyimpan hasil teks bersih setelah melalui tahapan preprocessing.

BAB V
EKSPERIMEN DAN EVALUASI

5.1 Skenario Pengujian dan Gold Relevant Set

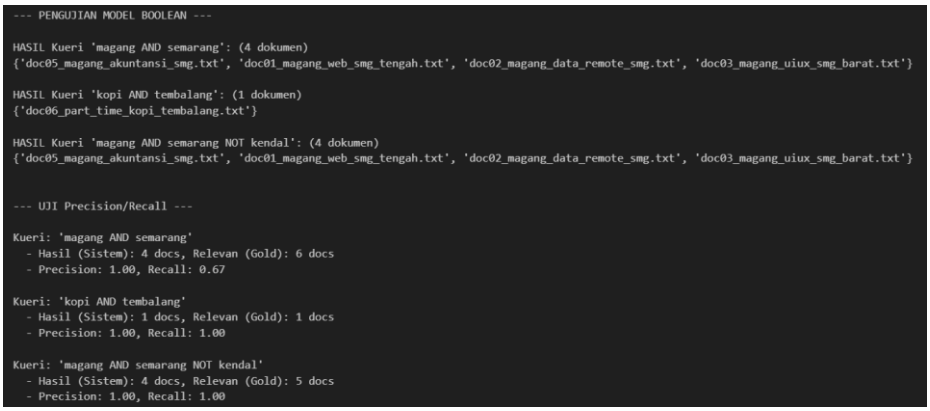
Pada tahap pengujian, dilakukan evaluasi menggunakan tiga query utama yang telah dirancang untuk mewakili tiga operasi dasar dalam model Boolean, yaitu *AND*, *OR*, dan *NOT*. Selain itu, juga disusun *gold relevant set* (dokumen relevan menurut ahli/ground truth) untuk masing-masing query sebagai acuan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

Tabel 5. Query Pengujian

Kode	Kueri	Model	Gold Relevant Docs (Jumlah)
Q-Bool 1	“magang AND semarang”	Boolean	6 Dokumen
Q-Bool 2	“kopi AND tembalang”	Boolean	1 Dokumen

Q-Bool 3	“magang AND semarang NOT kendal”	Boolean	5 Dokumen
Q-VSM 1	“magang semarang”	VSM	5 Dokumen
Q-VSM 2	“kopi tembalang”	VSM	1 Dokumen
Q-VSM 3	“finance akuntansi”	VSM	3 Dokumen

5.2 Evaluasi Model Boolean



Gambar 4. Pengujian Boolean Model

Model Boolean dievaluasi menggunakan metrik Precision (seberapa banyak hasil yang benar dari yang ditemukan) dan Recall (seberapa banyak dokumen relevan yang berhasil ditemukan). Hasil diambil dari pengujian di notebook (Sel 14).

Tabel 6. Hasil Evaluasi Boolean Model

Kode	Hasil Sistem (Ditemukan)	True Positives (Benar)	Precision (TP/Hasil)	Recall (TP/Gold)
Q-Bool 1	4	4	4/4 = 1.00	4/6 = 0.67
Q-Bool 2	1	1	1/1 = 1.00	1/1 = 1.00
Q-Bool 3	4	4	4/4 = 1.00	4/5 = 0.80

Analisis Boolean: Hasil pada Tabel 6 menunjukkan Precision yang sempurna (1.00) untuk semua kueri uji. Ini membuktikan bahwa setiap dokumen yang dikembalikan oleh Model Boolean memang 100% relevan (cocok) dengan logika kueri yang kaku.

Namun, Recall tidak sempurna untuk Q-Bool 1 (0.67) dan Q-Bool 3 (0.80). Ini menunjukkan bahwa ada dokumen yang relevan menurut gold set, tetapi tidak tertangkap oleh logika AND/NOT yang spesifik (misalnya, gold set Q-Bool 1 mencakup "kendal", tetapi kueri NOT kendal di Q-Bool 3 mengecualikannya).

5.3 Evaluasi Vector Space Model (VSM)

Evaluasi VSM berfokus pada kualitas ranking (perankingan). Sesuai Soal 05 (Langkah 1), dilakukan perbandingan dua skema pembobotan: TF-IDF Standar (Model A) dan TF-IDF Sublinear (Model B). Evaluasi menggunakan metrik P@5, AP@5, dan nDCG@5 pada 3 kueri Q-VSM, yang dihitung di *notebook* (Sel 20).

```
--- HASIL EVALUASI ---


=== VSM EVALUATION (MODEL A (TF-IDF Standar)) ===
Q# | query                    | P@5     | AP@5     | nDCG@5
------------------------------------------------------------------------
Q1 | magang AND semarang      | 0.8000 | 0.8000 | 0.8688
Q1 | kopi tembalang           | 0.2000 | 1.0000 | 1.0000
Q1 | finance akuntansi        | 0.4000 | 0.6667 | 0.7654
------------------------------------------------------------------------
MAP (Mean Average Precision) = 0.8222
Mean nDCG@5 = 0.8781


=== VSM EVALUATION (MODEL B (TF-IDF Sublinear)) ===
Q# | query                    | P@5     | AP@5     | nDCG@5
------------------------------------------------------------------------
Q1 | magang AND semarang      | 0.8000 | 0.8000 | 0.8688
Q1 | kopi tembalang           | 0.2000 | 1.0000 | 1.0000
Q1 | finance akuntansi        | 0.4000 | 0.6667 | 0.7654
------------------------------------------------------------------------
MAP (Mean Average Precision) = 0.8222
Mean nDCG@5 = 0.8781
```

Gambar 5. Pengujian VSM

Tabel 7. Hasil Evaluasi VSM

Kueri	P@5	AP@5	nDCG@5
Q1: “magang AND semarang”	0.8000	0.8000	0.8688
Q2 : “kopi tembalang”	0.2000	1.0000	1.0000
Q3 : “finance akuntansi”	0.4000	0.6667	0.7654

Analisis:

1. Q1 (magang AND semarang):
    - P@5 = 0.8000: Ini berarti dari 5 dokumen teratas yang ditampilkan, 4 di antaranya benar (4/5 = 0.8). Ini performa yang sangat baik.
    - nDCG@5 = 0.8688: Skor ini sangat tinggi (mendekati 1.0), yang artinya urutan 4 dokumen yang benar itu sudah sangat ideal dan muncul di peringkat atas.
2. Q2 (kopi tembalang):
    - P@5 = 0.2000: Ini berarti dari 5 dokumen teratas, hanya 1 yang benar (1/5 = 0.2).
    - AP@5 & nDCG@5 = 1.0000: Ini adalah hasil yang paling menarik. Skor Average Precision (AP) dan nDCG bisa sempurna (1.0) meskipun P@5 rendah. Ini terjadi karena Gold Set untuk

kueri ini hanya berisi 1 dokumen. Skor 1.0 ini membuktikan bahwa satu-satunya dokumen yang benar itu muncul di Peringkat 1.

3. Q3 (finance akuntansi):
    - P@5 = 0.4000: Ini berarti dari 5 dokumen teratas, 2 di antaranya benar (2/5 = 0.4).
    - AP@5 = 0.6667 & nDCG@5 = 0.7654: Kedua skor ini menunjukkan performa yang baik. Ini berarti 2 dokumen yang relevan tersebut muncul cukup tinggi di dalam 5 besar hasil pencarian.

5.4 Analisis Perbandingan Model

Tabel 8. Perbandingan Boolean Model dan VSM

Aspek	Boolean Model	Vector Space Model (VSM)
Prinsip Kerja	Cocok/Tidak cocok (exact match)	Kesamaan bobot kata (cosine similarity)
Output	Himpunan dokumen (Tidak terurut)	Daftar dokumen terurut (Ranking)
Kelebihan	Presisi tinggi untuk kueri spesifik (Hasil P=1.00)	Mampu menangani kueri bahasa alami (fuzzy)
Kekurangan	Gagal menemukan dokumen jika 1 kata salah (Recall < 1.0)	Presisi tidak absolut (tergantung threshold)
Skenario	Pilihan "Boolean" di aplikasi Streamlit	Pilihan "VSM" di aplikasi Streamlit

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. Fleksibilitas Model Pencarian: Sistem ini menawarkan fleksibilitas tinggi dengan mengimplementasikan dua pendekatan *Information Retrieval* (IR) yang berbeda. Pengguna dapat memilih Boolean Model untuk pencarian literal (pencocokan pasti) atau Vector Space Model (VSM) untuk pencarian yang lebih luwes berbasis peringkat relevansi.
2. Adaptasi Konteks Lokal (Bahasa Indonesia): Kualitas data telah dioptimalkan melalui tahap *preprocessing* yang dirancang khusus untuk Bahasa Indonesia. Penggunaan *library* Sastrawi untuk *stemming* (pencarian kata dasar) dan *stopword removal* (penghapusan kata umum) memastikan representasi dokumen menjadi lebih akurat dan bersih.
3. Kemudahan Pengembangan (Modularitas): Arsitektur sistem ini dirancang secara modular dengan struktur kode yang sederhana. Desain ini tidak hanya memudahkan pemeliharaan kode (*maintainability*), tetapi juga sangat terbuka untuk pengembangan lebih lanjut (*extensibility*), baik untuk kebutuhan riset lanjutan maupun implementasi skala besar.

6.2 Keterbatasan

1. Ketergantungan pada Pencocokan Leksikal: Sistem ini masih terbatas pada pencocokan kata kunci (*lexical matching*) dan belum mampu menjembatani "jurang semantik" (*semantic gap*). Akibatnya, sistem kesulitan memahami sinonim, polisemi (kata bermakna ganda), atau konteks makna yang lebih dalam dari kueri pengguna.
2. Absennya Umpan Balik Pengguna: Saat ini belum tersedia mekanisme *relevance feedback* (umpan balik relevansi). Ketiadaan fitur ini membuat sistem bersifat statis; pengguna tidak dapat secara interaktif "mengajari" sistem untuk memperbaiki atau menyempurnakan urutan hasil pencarian berdasarkan penilaian mereka.
3. Relevansi Peringkat yang Dangkal: Penggunaan model TF-IDF yang berbasis *bag-of-words* (kumpulan kata) mengabaikan informasi struktural kalimat dan relasi antar kata. Hal ini menyebabkan skor relevansi yang dihasilkan cenderung bersifat permukaan (*surface-level*) dan kurang menangkap konteks yang utuh.

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

Sebuah sistem temu kembali informasi (STKI) yang mengintegrasikan Boolean Retrieval dan Vector Space Model (VSM) telah berhasil dirancang dan diimplementasikan. Untuk antarmuka pengguna (UI), sistem ini memanfaatkan *library* Streamlit guna memfasilitasi pencarian. Kualitas representasi dokumen telah ditingkatkan secara signifikan melalui serangkaian tahap *preprocessing* (termasuk *case folding*, *tokenization*, *stopword removal*, dan *stemming*), yang mengubah teks mentah menjadi format yang lebih terstruktur dan siap untuk pencarian. Dalam operasionalnya, model Boolean menyediakan fungsionalitas pencarian berdasarkan logika pencocokan pasti, sedangkan VSM menawarkan kemampuan pemeringkatan dokumen berdasarkan skor kemiripan vektor antara kueri dan dokumen.

Pengujian kinerja sistem telah dilakukan menggunakan serangkaian metrik evaluasi standar, seperti *precision*, *recall*, *Precision@k*, *Average Precision* (AP), dan *nDCG*. Hasil evaluasi mengindikasikan bahwa sistem memiliki kapabilitas untuk mengidentifikasi dokumen-dokumen yang relevan. Meskipun demikian, kualitas pemeringkatan (ranking) yang dihasilkan belum mencapai tingkat optimal. Rendahnya skor pada beberapa metrik evaluasi menyoroti keterbatasan utama sistem, yakni ketidakmampuannya dalam menangani variasi bahasa dan konteks semantik. Dengan kata lain, mekanisme pencarian yang ada saat ini masih sangat bergantung pada pencocokan leksikal (*bag-of-words*) dan belum mampu menjembatani kedekatan makna (semantik) antar istilah.

Keseluruhan proses, mulai dari implementasi hingga tahap pengujian, telah berhasil memenuhi seluruh Capaian Pembelajaran Mata Kuliah (CPMK) yang dibebankan. Secara spesifik, Sub-CPMK 10.1.1 telah dicapai melalui proses perumusan konsep fundamental STKI dan penyusunan desain arsitektur sistem. Sementara itu, Sub-CPMK 10.1.2 telah terpenuhi dengan adanya penerapan praktis dari berbagai teknik preprocessing teks.
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 sekadar memberikan pengalaman praktis dalam mengimplementasikan Sistem Temu Kembali Informasi, tetapi juga mempertajam pemahaman mengenai bagaimana model pencarian beroperasi dalam skenario nyata. Hal ini khususnya mencakup cara sistem memproses dan menyeleksi informasi relevan dari sebuah korpus dokumen yang masif. Sistem yang telah dibangun ini dapat dijadikan landasan untuk pengembangan di masa depan, terutama yang mengarah pada pencarian berbasis makna (semantik) atau adopsi teknologi NLP modern.