UJIAN AKHIR SEMESTER
SISTEM REKOMENDASI SITASI DAN DOSEN PEMBIMBING UNIVERSITAS DIAN NUSWANTORO MENGGUNAKAN ALGORITMA TF-IDF, COSINE SIMILARITY, DAN BM25

Disusun Oleh :

Sabrina Aska Amalina (A11.2023.15264)

TEKNIK INFORMATIKA

FAKULTAS ILMU KOMPUTER

UNIVERSITAS DIAN NUSWANTORO

DAFTAR ISI

DAFTAR TABEL

DAFTAR GAMBAR

BAB I
PENDAHULUAN

1.1. Latar Belakang

Penelitian merupakan salah satu tahap penting yang harus dilalui oleh mahasiswa tingkat akhir sebagai syarat untuk menyelesaikan studi di perguruan tinggi. Pada tahap awal penelitian, mahasiswa di wajibkan untuk menentukan topik penelitian, mencari referensi ilmiah yang relevan, serta mengajukan dosen pembimbing yang sesuai dengan topik penelitian yang di pilih. Namun, proses awal ini sering menjadi tantangan tersendiri bagi mahasiswa, terutama bagi mereka yang baru pertama kali melakukan penelitian ilmiah. Salah satu kendala utamanya adalah sulitnya menemukan referensi sitasi yang relevan dan menentukan dosen pembimbing yang memiliki kepakaran yang sesuai. Menurut penelitian (Syarif & Wiguna, 2023), ketidak sesuaian antara topik mahasiswa dengan kompetensi dosen dapat menghambat durasi penyelesaian tugas akhir dan menurunkan kualitas karya ilmiah.

Secara ilmiah, penelitian terkini menunjukkan bahwa *scholarly recommendation system* berperan penting untuk membantu peneliti/mahasiswa menemukan sumber relevan secara lebih cepat, serta meminimalkan risiko terlewatnya literatur penting dalam tahap perumusan masalah dan penyusunan landasan teori. Sistem rekomendasi ilmiah banyak dikembangkan untuk literatur dan sumber riset lain dengan pendekatan *content-based* maupun *hybrid*, karena terbukti membantu efisiensi pencarian dan relevansi hasil (Zitong Zhang, 2023). Selain itu, kebutuhan rekomendasi yang lebih "berbasis bukti" juga sejalan dengan perkembangan *knowledge-based recommender systems*, yang menekankan penggunaan atribut/pengetahuan domain agar rekomendasi lebih terarah dan dapat dipertanggungjawabkan (Mathias Uta, 2024).

Berdasarkan permasalahan tersebut, diperlukan solusi berupa sistem rekomendasi terintegrasi di lingkungan UDINUS yang mampu merekomendasikan referensi sitasi yang relevan terhadap topik penelitian mahasiswa, dan merekomendasikan dosen pembimbing berdasarkan kesesuaian topik dengan profil kepakaran yang dibangun dari publikasi dosen. Gagasan ini didukung oleh penelitian terbaru terkait rekomendasi pembimbing. Adapun penelitian milik (Dasri Dasria, 2025) yang mengembangkan sistem rekomendasi pembimbing/tema penelitian menggunakan data akademik dan informasi topik untuk menghasilkan rekomendasi yang lebih objektif.

Dalam rancangan solusi, sistem dapat dibangun dengan pendekatan *content-based* dan *expert finding* berbasis korpus publikasi dicocokkan dengan metadata publikasi menggunakan

teknik *Information Retrieval* dan kemiripan teks. Secara operasional, metode seperti BM25 dapat digunakan untuk penyaringan awal yang relevan secara leksikal, kemudian dilakukan pemeringkatan lanjutan menggunakan TF-IDF dan *cosine similarity* agar kecocokan konteks topik lebih stabil. Dengan mekanisme ini, sistem diharapkan mampu mengurangi dominasi kata kunci umum, meningkatkan ketepatan rekomendasi sitasi, serta membantu mahasiswa memilih dosen pembimbing yang selaras dengan kepakaran dan rekam publikasi yang terukur.

1.2. Masalah Penelitian

Masalah penelitian ini disusun untuk menegaskan fokus pengembangan sistem rekomendasi penelitian di lingkungan UDINUS. Adapun masalah penelitian yaitu sebagai berikut :

1. Proses penentuan referensi ilmiah pada tahap awal penelitian masih menjadi kendala bagi mahasiswa karena hasil penelusuran sering terlalu luas, tidak spesifik, dan tidak seluruhnya relevan dengan konteks topik yang dipilih.
2. Penentuan dosen pembimbing cenderung dilakukan berdasarkan informasi terbatas, sehingga berpotensi terjadi ketidaksesuaian antara topik penelitian mahasiswa dan kepakaran dosen yang seharusnya tercermin dari rekam jejak publikasi.
3. Belum tersedianya sistem terintegrasi di lingkungan UDINUS yang secara objektif dapat merekomendasikan referensi sitasi yang relevan dan dosen pembimbing yang sesuai berdasarkan kesesuaian topik dengan korpus publikasi.

1.3. Tujuan Penelitian

Adapun tujuan penelitian ini disusun sebagai arah dan capaian yang ingin diwujudkan. Berikut tujuan dari penelitian :

1. Merancang dan membangun sistem rekomendasi penelitian di UDINUS yang dapat membantu mahasiswa menemukan referensi sitasi yang relevan dengan topik penelitian.
2. Mengembangkan modul rekomendasi dosen pembimbing berdasarkan kesesuaian topik penelitian mahasiswa dengan profil kepakaran dosen yang dibentuk dari data publikasi.
3. Menerapkan pendekatan *Information Retrieval* dan pemodelan kemiripan teks untuk proses penyaringan dan pemeringkatan rekomendasi, sehingga hasil yang diberikan lebih tepat sasaran dan mengurangi bias kata kunci generik.

BAB II
LANDASAN TEORI

2.1. Sistem Rekomendasi

Sistem rekomendasi adalah sistem yang membantu pengguna menemukan konten yang diminati dalam sistem informasi modern (Robin Burke, 2025). Pada kasus akademik (khususnya penentuan sitasi dan pembimbing), *content-based* sering lebih sesuai karena dapat memanfaatkan informasi teks tanpa bergantung pada histori rating/klik yang sering tidak tersedia.

2.2. *Text Mining* dan *Preprocessing Text*

Text mining bertujuan mengekstraksi informasi bermakna dari teks tidak terstruktur agar dapat diolah secara komputasional. Tahap praproses diperlukan untuk meningkatkan konsistensi data teks dan mengurangi *noise*, misalnya *case folding*, pembersihan tanda baca, tokenisasi, penghapusan stopword, normalisasi, serta stemming. Studi pada pemrosesan teks bahasa Indonesia menunjukkan bahwa skenario praproses yang lebih komprehensif dapat meningkatkan kinerja metode berbasis kemiripan karena mengurangi variasi bentuk kata dan kata umum yang tidak informatif (Andri Setiawan, 2025).

2.3. *Information Retrieval*

Information Retrieval (IR) mempelajari cara menemukan dokumen yang relevan terhadap query pengguna. Dalam sistem rekomendasi sitasi, *query* dapat berupa judul/abstrak ringkas/topik mahasiswa, sedangkan dokumen adalah korpus publikasi. IR sering digunakan sebagai fondasi karena mampu melakukan pemeringkatan dokumen secara efisien pada koleksi besar serta menyediakan baseline kuat sebelum pendekatan yang lebih kompleks diterapkan (Nandan Thakur, 2021).

2.4. Algoritma TF-IDF

TF-IDF (*Term Frequency-Inverse Document Frequency*) adalah metode statistik yang banyak digunakan dalam *natural language processing* dan *information retrieval* untuk mengevaluasi seberapa penting sebuah kata terhadap dokumen dalam kaitannya dengan koleksi dokumen yang lebih besar (GeeksforGeeks, 2025).

2.5. Algoritma BM25

BM25 (*Best Matching* 25) adalah algoritma pemeringkatan yang digunakan dalam sistem penelusuran informasi untuk menentukan seberapa relevan suatu dokumen dengan kueri pencarian tertentu. Ini adalah versi yang lebih baik dari pendekatan TF-IDF (*Term Frequency–Inverse Document Frequency*) tradisional dan banyak digunakan dalam mesin pencari dan basis data modern (GeeksforGeeks, 2025).

2.6. *Cosine Similarity*

Cosine Similarity adalah metrik yang digunakan untuk mengukur kesamaan antara dua vektor bukan nol dengan menghitung kosinus sudut di antara keduanya. Metode ini banyak digunakan dalam pembelajaran mesin dan analisis data, terutama dalam analisis teks, perbandingan dokumen, kueri pencarian, dan sistem rekomendasi (GeeksforGeeks, 2015).

BAB III
HASIL DAN PEMBAHASAN

3.1. Dataset

Data yang digunakan dalam penelitian ini berasal dari *corpus* publikasi yang merepresentasikan karya ilmiah di lingkungan Universitas Dian Nuswantoro, yang secara umum dapat dikelompokkan sebagai berikut:

1. Publikasi Kampus

   Publikasi kampus diperoleh dari repository resmi Universitas Dian Nuswantoro yang menyediakan berbagai dokumen ilmiah, seperti tugas akhir mahasiswa, laporan penelitian, dan materi publikasi lainnya. Data dari repository ini dapat diakses dan diekstraksi metadatanya untuk mendukung proses analisis. Kanal repository UDINUS dapat diakses melalui laman https://perpustakaan.dinus.ac.id/resource/publikasi-udinus/.

2. Publikasi Dosen Pembimbing

   Publikasi dosen pembimbing merupakan kumpulan karya ilmiah yang ditulis atau melibatkan dosen Universitas Dian Nuswantoro. Data ini digunakan untuk membentuk profil keilmuan dosen berdasarkan rekam jejak publikasinya. Informasi tersebut memiliki peran penting dalam proses rekomendasi dosen pembimbing, karena tingkat kesesuaian topik penelitian mahasiswa akan dihitung berdasarkan representasi teks dari publikasi dosen, seperti judul, abstrak, dan kata kunci.

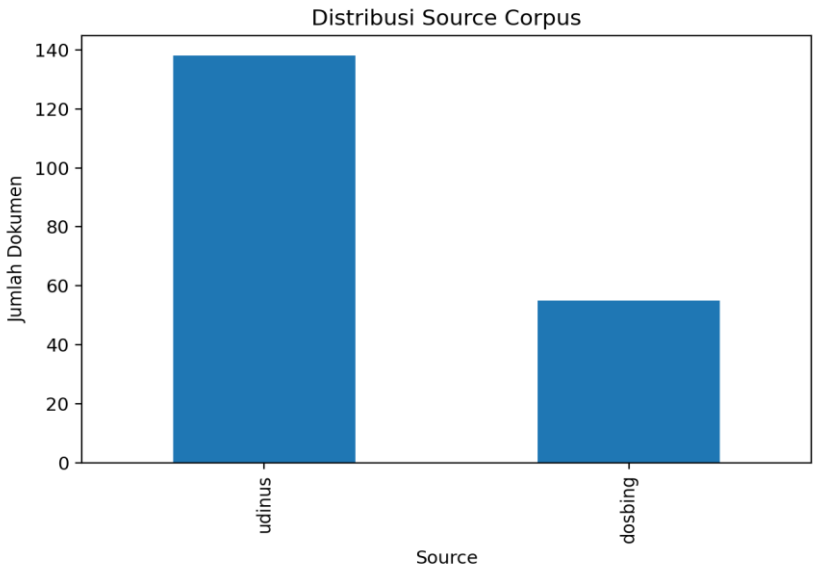
Dataset ini memiliki 11 kolom fitur dengan 193 baris. 138 data berasal dari *corpus* publikasi kampus dan 55 sisanya adalah *corpus* penelitian dosen pembimbing. Adapun fitur dari dataset tersebut adalah sebagai berikut.

Tabel 1 Deskripsi Fitur

Fitur	Deskripsi
doc_id	Identitas unik untuk setiap dokumen publikasi yang digunakan sebagai penanda data.
source	Sumber asal data publikasi (UDINUS, Dosbing)

dosen	Nama dosen pembimbing atau dosen terkait dengan publikasi. Pada beberapa data, atribut ini dapat bernilai kosong apabila belum tersedia.
path	Lokasi penyimpanan berkas dokumen publikasi pada sistem atau direktori lokal.
url	Alamat tautan asli publikasi yang dapat diakses secara daring.
tanggal	Tanggal publikasi dokumen ilmiah, digunakan untuk informasi waktu dan analisis temporal jika diperlukan.
judul	Judul karya ilmiah yang merepresentasikan topik utama penelitian.
keyword	Kata kunci yang menggambarkan fokus dan ruang lingkup penelitian.
abstrak	Ringkasan isi penelitian yang menjelaskan tujuan, metode, dan hasil penelitian, serta menjadi sumber utama dalam proses analisis teks.
peneliti	Nama penulis atau peneliti yang terlibat dalam publikasi ilmiah.
misc	Informasi tambahan di luar atribut utama.

Adapun grafik distribusi sumber dokumen menunjukkan komposisi corpus berdasarkan asal publikasi. Analisis ini diperlukan untuk memastikan proporsi data mencukupi bagi masing-masing fungsi sistem, yakni rekomendasi sitasi dan pembentukan profil dosen pembimbing.



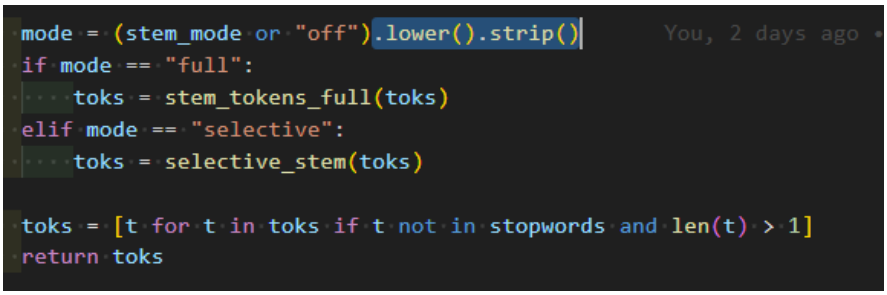
Gambar 3. 1 Grafik Distribusi Sumber Dokumen

3.2. Data *Preprocessing*

Tahap ini dilakukan untuk membentuk *corpus* menjadi teks indeks dan membentuk token hasil *preprocessing* agar siap digunakan untuk *modelling* BM25 dan *profilling* dosen. Adapun tahap yang dilakukan yaitu sebagai berikut.

1. *Case Folding* (*Lowercasing*)

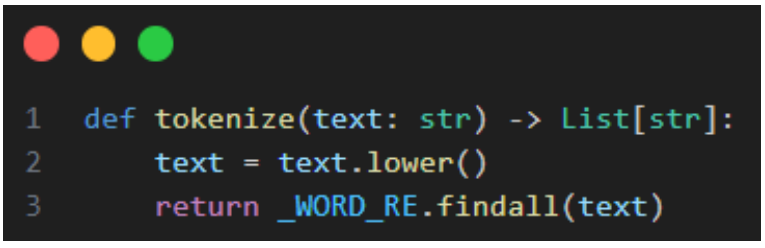
   Seluruh teks pada *corpus* diubah menjadi huruf kecil (*lowercase*). Adapun tujuannya adalah untuk mengurangi variasi token akibat perbedaan kapitalisasi, sehingga kata yang sama tidak dianggap sebagai token yang berbeda. Misal "Prediksi", "prediksi", dan "PREDIKSI" akan diperlakukan sebagai token yang sama yaitu "prediksi".



Gambar 3. 2 *Case Folding*

2. Tokenisasi

   Tokenisasi adalah proses memecah teks menjadi unit kata (token) yang akan diproses lebih lanjut. Pada penelitian ini, tokenisasi menggunakan pola *regex* yang dimana token yang diambil adalah rangkaian karakter huruf dan angka. Hal ini dilakukan karena istilah teknis sering memuat angka (misalnya "4g", "5g", "web3", "industri4", "3d"), mengurangi *noise* dari simbol atau tanda baca yang tidak membawa makna topik, menjaga konsistensi token untuk proses pencocokan antara *query* pengguna dan dokumen.



Gambar 3. 3 Tokenisasi

3. *Stopword Removal* (*Stopwords* Umum dan Kata Akademik)

   Setelah teks menjadi daftar token, dilakukan penghapusan stopwords yaitu kata-kata yang sangat sering muncul namun umumnya tidak membantu membedakan topik dokumen. Stopword removal dilakukan dalam dua lapisan yaitu stopwords bahasa

Indonesia dasar misal “dan”, “yang”, “dari”, “dengan”, “untuk”, “pada”, dan kata fungsi lainnya. Adapun kata generik akademik (generic research words) misal “penelitian”, “analisis”, “metode”, “hasil”, “studi”, “pendekatan”, “data”.

```python
try:
    from Sastrawi.Stemmer.StemmerFactory import StemmerFactory
    from Sastrawi.StopWordRemover.StopWordRemoverFactory import StopWordRemoverFactory
    _HAS_SASTRAWI = True
except ModuleNotFoundError:
    StemmerFactory = None
    StopWordRemoverFactory = None
    _HAS_SASTRAWI = False

_WORD_RE = re.compile(r"[a-zA-Z0-9_]+", re.UNICODE)

_STEMMER = StemmerFactory().create_stemmer() if _HAS_SASTRAWI else None
_SASTRAWI_STOPWORDS = set(StopWordRemoverFactory().get_stop_words()) if _HAS_SASTRAWI else set()

_TECH_WHITELIST = {
    "ai","ml","nlp","cnn","rnn","lstm","gru","svm","knn","rf","xgboost","bert","gpt",
    "embedding","transformer","token","tokenizer","dataset","benchmark","accuracy",
    "precision","recall","f1","roc","auc","api","sql","mysql","nosql","json","xml",
    "http","https","tcp","udp","gpu","cpu","ram","iot","ui","ux","devops","docker",
    "kubernetes","k8s","linux","windows","android","ios","react","nextjs","node",
    "python","java","golang","rust","c","cpp","csharp","php","javascript","typescript"
}
```

Gambar 3. 4 *Stopword Removal*

4. *Stemming*

Stemming merupakan proses mengubah setiap token ke bentuk dasarnya (*root word*) dengan tujuan mengurangi variasi morfologi pada kata berimbuhan. Dalam bahasa Indonesia, satu konsep dapat muncul dalam banyak bentuk, misalnya *“mengembangkan”, “pengembangan”, “dikembangkan”* yang pada dasarnya merujuk pada kata *“kembang”*.

```python
def selective_stem(tokens: List[str]) -> List[str]:
    out: List[str] = []
    for t in tokens:
        if _looks_indonesianish(t):
            if _STEMMER is None:
                out.append(t)
            else:
                out.append(_STEMMER.stem(t))
        else:
            out.append(t)
    return out
```

Gambar 3. 5 *Stemming*

3.3. Implementasi Rekomendasi Sitasi Menggunakan BM25

Implementasi BM25 pada penelitian diperlukan karena sistem harus melakukan pencarian terarah pada corpus internal (publikasi UDINUS) dengan cara memberi skor relevansi untuk setiap dokumen terhadap query ide penelitian.

3.3.1. Pembentukan Koleksi Dokumen untuk Indexing

Pada tahap ini, corpus publikasi yang telah melalui proses *data preparation* digunakan sebagai koleksi dokumen untuk membangun indeks BM25. Setiap dokumen diwakili oleh token hasil preprocessing. Dengan representasi token tersebut, indeks BM25 mampu membandingkan query dan dokumen dalam ruang kosakata yang konsisten. Agar sistem rekomendasi sitasi fokus pada publikasi yang relevan, dokumen yang digunakan untuk indeks dapat disaring berdasarkan atribut source (misalnya hanya publikasi kampus/paper), sedangkan publikasi dosen diproses terpisah untuk rekomendasi dosbing.

3.3.2. Proses Pembentukan Indeks BM25

Pembentukan indeks BM25 dilakukan dengan langkah berikut:

1. Koleksi token dokumen disusun menjadi list dokumen, di mana setiap dokumen adalah list token hasil preprocessing.
2. Sistem menghitung komponen statistik corpus yang dibutuhkan BM25, yaitu:
    - N: jumlah dokumen dalam corpus,
    - df(t): jumlah dokumen yang mengandung term t,
    - IDF(t): bobot term berdasarkan kelangkaannya dalam corpus,
    - dl: panjang dokumen (jumlah token),
    - avgdl: rata-rata panjang dokumen pada corpus.
3. Parameter BM25 yang digunakan:
    - k1: mengontrol sensitivitas terhadap frekuensi term dalam dokumen,
    - b: mengontrol normalisasi panjang dokumen.

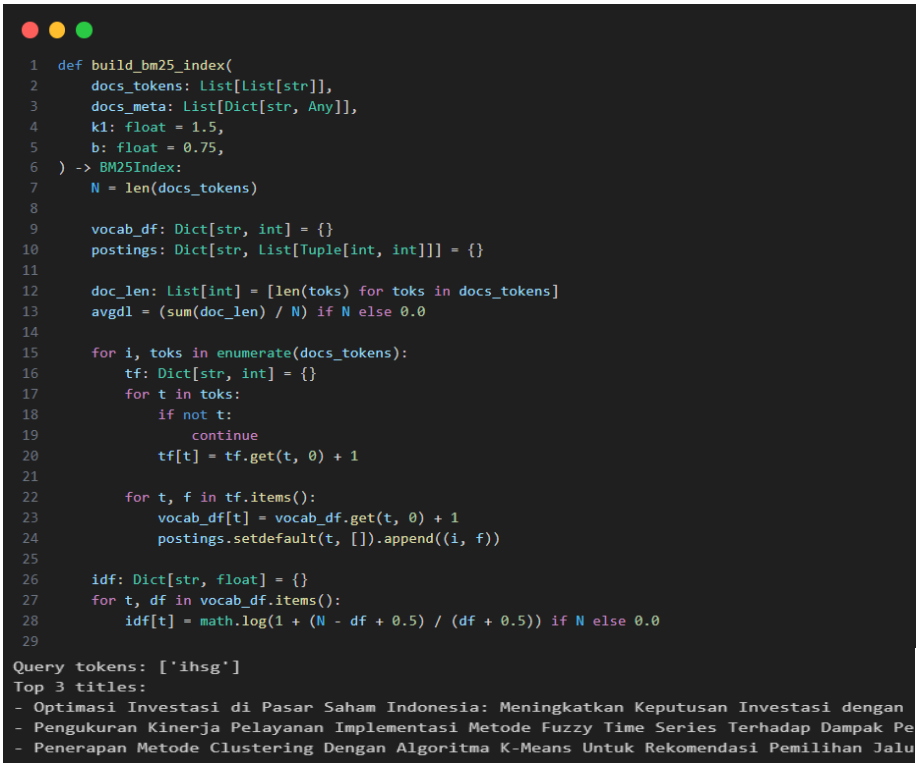
Pada implementasi ini digunakan nilai parameter yang umum dipakai sebagai *baseline*, misalnya k1 = 1.5 dan b = 0.75, kemudian parameter tersebut dapat disesuaikan apabila dilakukan *tuning* berdasarkan hasil evaluasi.

3.3.3. Perhitungan Skor Relevansi BM25

Setelah query diproses menjadi token, sistem menghitung skor relevansi BM25 untuk setiap dokumen. Secara konsep, BM25 menghitung kontribusi setiap term query terhadap dokumen menggunakan gabungan:

- TF (term frequency): seberapa sering term muncul dalam dokumen,
- IDF (inverse document frequency): seberapa penting term tersebut dalam corpus,
- Normalisasi panjang dokumen: agar dokumen panjang tidak selalu unggul hanya karena memiliki banyak token.

Skor total sebuah dokumen diperoleh dengan menjumlahkan kontribusi semua term query. Dokumen dengan skor tertinggi dianggap paling relevan terhadap query.



```python
def build_bm25_index(
    docs_tokens: List[List[str]],
    docs_meta: List[Dict[str, Any]],
    k1: float = 1.5,
    b: float = 0.75,
) -> BM25Index:
    N = len(docs_tokens)

    vocab_df: Dict[str, int] = {}
    postings: Dict[str, List[Tuple[int, int]]] = {}

    doc_len: List[int] = [len(toks) for toks in docs_tokens]
    avgdl = (sum(doc_len) / N) if N else 0.0

    for i, toks in enumerate(docs_tokens):
        tf: Dict[str, int] = {}
        for t in toks:
            if not t:
                continue
            tf[t] = tf.get(t, 0) + 1

        for t, f in tf.items():
            vocab_df[t] = vocab_df.get(t, 0) + 1
            postings.setdefault(t, []).append((i, f))

    idf: Dict[str, float] = {}
    for t, df in vocab_df.items():
        idf[t] = math.log(1 + (N - df + 0.5) / (df + 0.5)) if N else 0.0
```

```
Query tokens: ['ihsg']
Top 3 titles:
- Optimasi Investasi di Pasar Saham Indonesia: Meningkatkan Keputusan Investasi dengan
- Pengukuran Kinerja Pelayanan Implementasi Metode Fuzzy Time Series Terhadap Dampak Pe
- Penerapan Metode Clustering Dengan Algoritma K-Means Untuk Rekomendasi Pemilihan Jalu
```

Gambar 3. 6 Proses BM25

3.4. Rekomendasi Dosen Pembimbing Menggunakan TF-IDF + Cosine Similarity

Implementasi rekomendasi dosen pembimbing (dosbing) berbasis kemiripan teks menggunakan TF-IDF sebagai representasi vektor dokumen dan *Cosine Similarity* sebagai ukuran kedekatan antara topik penelitian mahasiswa dan profil keilmuan dosen. Pendekatan ini digunakan karena sistem perlu memetakan *query* ide penelitian yang biasanya berupa deskripsi singkat ke kumpulan publikasi dosen yang cenderung panjang, bervariasi, dan tersebar pada banyak dokumen. Dengan TF-IDF, istilah yang khas pada suatu bidang akan memperoleh bobot lebih besar, sedangkan istilah yang terlalu umum di seluruh corpus akan diredam.

3.4.1. Penyaringan Data untuk Modul Dosbing

Tahap awal adalah menentukan subset data yang digunakan untuk membangun profil dosen. Dokumen yang dipakai pada modul ini adalah dokumen dengan kategori sumber yang berkaitan dengan publikasi dosen, misalnya:

- source = "dosbing" atau label lain yang menandai publikasi dosen
- memiliki atribut dosen (misalnya lecturer_name / dosen) yang valid

3.4.2. Normalisasi Identitas Dosen

Konsistensi identitas dosen menjadi aspek penting agar satu dosen tidak terpecah menjadi beberapa entitas berbeda. Variasi penulisan nama dapat terjadi akibat penggunaan gelar (S.Kom., M.Kom., Dr., Prof.). Normalisasi dilakukan dengan menyamakan huruf menjadi lowercase, menghapus spasi ganda dan tanda baca tertentu, menyimpan nama asli untuk tampilan (*display name*), namun menggunakan nama ternormalisasi untuk kunci pengelompokan.

3.4.3. Pembentukan Profil Dosen

Karena setiap dosen dapat memiliki lebih dari satu publikasi, sistem membangun profil dosen dengan cara menggabungkan seluruh teks dari publikasi yang dimiliki dosen tersebut menjadi satu dokumen representatif. Proses ini dilakukan dengan mengambil teks dari setiap publikasi yang terdiri atas judul, abstrak, dan kata kunci, kemudian mengelompokkan publikasi berdasarkan identitas dosen yang telah dinormalisasi. Seluruh teks publikasi milik dosen selanjutnya digabungkan menjadi satu teks profil dosen, sehingga terbentuk gambaran tema keilmuan dosen secara menyeluruh.

3.4.4. Vektorisasi TF-IDF pada Profil Dosen

Setelah preprocessing, sistem membangun matriks TF-IDF dari seluruh profil dosen. Proses vektorisasi meliputi:

- Fitting TF-IDF Vectorizer pada kumpulan profile_text dosen.
- Menghasilkan matriks vektor berukuran: (jumlah_dosen × jumlah_fitur/kosakata)
- Konfigurasi vektorisasi yang umum digunakan untuk meningkatkan kualitas perhitungan kemiripan meliputi penggunaan unigram dan bigram (*ngram_range* = (1,2)) agar frasa penting seperti "*machine learning*", "sistem rekomendasi", dan "*computer vision*" tetap terjaga maknanya, serta pembatasan kemunculan term yang terlalu sering (*max_df*) agar kata-kata yang muncul pada hampir seluruh profil tidak mendominasi hasil analisis.

Tujuan TF-IDF adalah memberi bobot lebih tinggi pada istilah yang khas pada dosen tertentu dan memberi bobot rendah pada istilah yang umum pada semua dosen.

3.4.5. Representasi *Query* dan Perhitungan *Cosine Similarity*

Query mahasiswa yang berisi ide/topik penelitian diproses dengan *preprocessing* yang sama, kemudian ditransformasikan ke ruang TF-IDF menggunakan *vectorizer* yang sama (tanpa *fit* ulang). Selanjutnya dihitung *Cosine Similarity* antara vektor *query* dan setiap vektor profil dosen. *Cosine Similarity* digunakan karena:

- Mengukur kemiripan berdasarkan arah vektor (pola istilah), bukan besarnya,
- Relatif stabil untuk teks dengan panjang berbeda (query pendek vs profil panjang),
- Nilai similarity berada pada rentang 0–1 (untuk vektor non-negatif), sehingga mudah diinterpretasikan.

Skor similarity dihitung untuk semua dosen dan diurutkan menurun untuk menghasilkan rekomendasi Top-K dosen.

3.4.6. Mekanisme Penyaringan dan Stabilitas Rekomendasi (Threshold & Gating)

Dalam praktiknya, TF-IDF + *Cosine* dapat menghasilkan skor kecil namun tetap memberi ranking. Agar hasil rekomendasi tidak memunculkan dosen yang sebenarnya tidak relevan, diterapkan mekanisme penyaringannya menggunakan ambang batas *minimum similarity* (*threshold*). Sehingga dosen dengan similarity di bawah threshold dihapus dari kandidat. Selain itu digunakan juga *relative cutoff*, sehingga jika skor tertinggi sangat kecil, sistem dapat menampilkan pesan bahwa query terlalu umum dan menyarankan kata kunci yang lebih spesifik.

	dosen	pub_count	doc_text
0	ABU SALAM, M.Kom	5	peningkatan kesadaran kanker usus siswa smp ibu kartini melalui mobilekanker usus merupakan salah satu penyakit dice...
1	ARDYTHA LUTHFIARTA, M.Kom	5	pengenalan pola aksara jawa korelasi template matchingpada era modern dampak globalisasi semakin masuk meluas sebagi...
2	CINANTYA PARAMITA, S.Kom., M.Eng	5	klasifikasi jeruk nipis tingkat kematangan buah berdasarkan fitur warna nearest neighborpada proses klasifikasi buah...
3	Dr. Amiq Fahmi, S. Kom., M.Kom	5	pendekatan klasterisasi pemetaan rencana kontinjensi denguetujuan epidemi dengue menunjukkan peningkatan jumlah pend...
4	FAUZI ADI RAFRASTARA, M.CS	5	optimasi investasi pasar saham indonesia meningkatkan keputusan investasi prediksi ihsg decision treepasar saham ind...

	dosen	pub_count	similarity	matched_terms
0	FAUZI ADI RAFRASTARA, M.CS	5	0.108510	[ihsg, prediksi]
1	CINANTYA PARAMITA, S.Kom., M.Eng	5	0.086572	[ihsg, prediksi]
2	Gustina Alfa Trisnapradika, M.Kom	5	0.047411	[prediksi]
3	Prof. Dr. RINDRA YUSIANTO, S.Kom, MT	5	0.016046	[prediksi]
4	ARDYTHA LUTHFIARTA, M.Kom	5	0.008841	[prediksi]

Gambar 3. 7 TF-IDF + Cosine

3.5. Evaluasi

Proses evaluasi dilakukan untuk menilai kinerja sistem rekomendasi sitasi (berbasis BM25) dan sistem rekomendasi dosen pembimbing (berbasis TF-IDF + Cosine Similarity).

	query	P@10	nDCG@10	n_rel
0	aplikasi prediksi harga saham atau IHSG	0.3	0.946902	3
1	rekomendasi sitasi untuk proposal PKM desa wisata	0.4	0.943866	4
2	klasifikasi sentimen bahasa indonesia menggunakan SVM	0.3	0.901013	3
3	deteksi malware berbasis machine learning	0.3	0.967468	3
4	information retrieval bm25 tf-idf untuk pencarian paper	0.3	0.967468	3

Gambar 3. 8 Hasil Evaluasi Precision@K dan nDCG@K

Berdasarkan evaluasi pada lima query uji, nilai Precision@10 berada pada rentang 0,3–0,4. Hal ini menunjukkan bahwa dari 10 rekomendasi teratas, rata-rata terdapat 3–4 dokumen yang dinilai relevan. Selain itu, nilai nDCG@10 yang tinggi (0,90–0,97) mengindikasikan bahwa dokumen relevan umumnya muncul pada peringkat yang lebih atas, sehingga kualitas ranking model BM25 tergolong baik untuk kebutuhan pencarian sitasi.

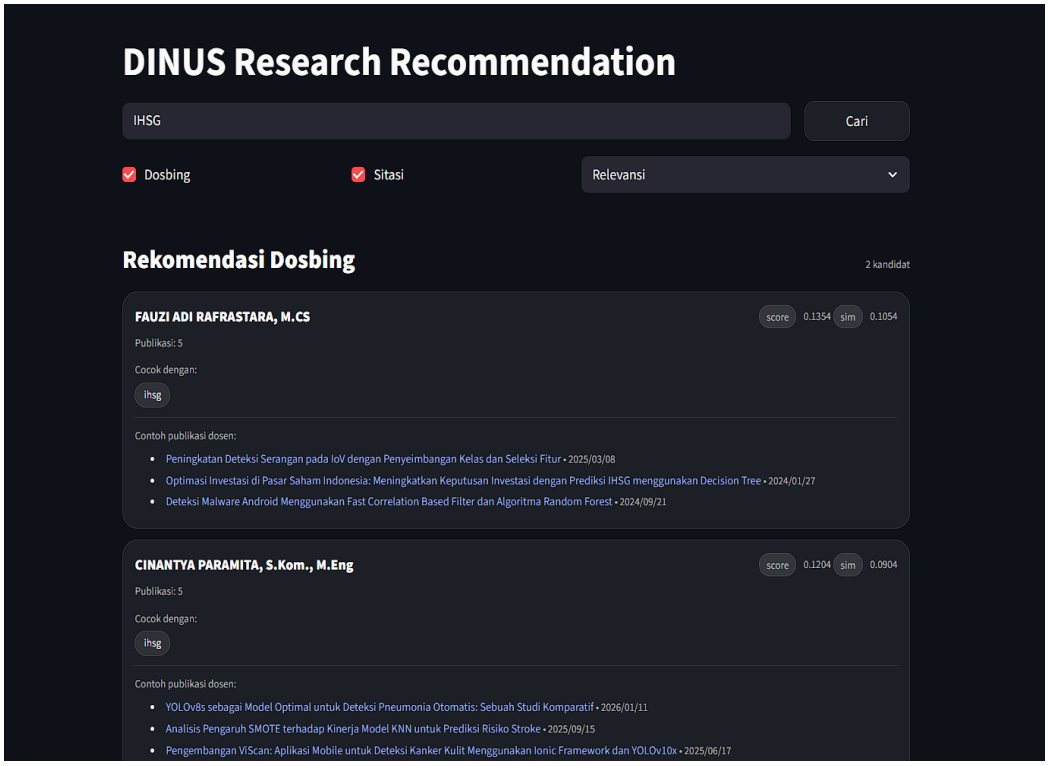
	doc_id	judul	tanggal	source	bm25_score	matched_terms	evidence
0	4968857.kemdiktisaintek	Model Hybrid Random Forest dan Information Gain untuk meningkatkan Performa Algoritma Machine Learning pada Deteksi ...	2024/09/30	dosbing	15.364607	learning, machine, malware	Oleh karena itu, diperlukan antivirus cerdas berbasis machine learning yang mampu mendeteksi malware berdasarkan per...
1	38334.ijeecs	Deteksi Malware Android Menggunakan Fast Correlation Based Filter dan Algoritma Random Forest	2024/09/21	dosbing	14.143186	learning, machine, malware	Penelitian ini membahas tantangan deteksi malware Android dengan memanfaatkan teknik machine learning tingkat lanjut...

Gambar 3. 9 Hasil Evaluasi Search

Tabel hasil rekomendasi menampilkan daftar dokumen teratas berdasarkan skor BM25. Kolom *matched_terms* menunjukkan token query yang ditemukan pada dokumen, sedangkan kolom *evidence* menampilkan potongan kalimat abstrak dengan kecocokan tertinggi sebagai justifikasi relevansi. Informasi ini digunakan untuk meningkatkan transparansi sistem dan membantu pengguna memvalidasi hasil rekomendasi secara cepat.

3.6. *Deployment*

Tahap *deployment* bertujuan memastikan sistem rekomendasi yang telah dibangun dapat diakses melalui antarmuka aplikasi, sehingga pengguna (mahasiswa) dapat memasukkan *query* topik penelitian dan memperoleh keluaran berupa rekomendasi sitasi serta rekomendasi dosen pembimbing. Pada penelitian ini, deployment dilakukan menggunakan Streamlit dengan link (https://dinusresearchrecommendationsystem.streamlit.app/). Setelah melakukan *deploy* ke Streamlit, maka sistem sudah dapat dijalankan. Berikut adalah sistem yang sudah di *deploy*.



Gambar 3. 10 Hasil *Deployment*

BAB IV
DISKUSI DAN KESIMPULAN

4.1. Diskusi

Berdasarkan hasil implementasi dan evaluasi, sistem rekomendasi penelitian UDINUS mampu memberikan dua keluaran utama, yaitu rekomendasi sitasi dan rekomendasi dosen pembimbing, dengan memanfaatkan pendekatan *Information Retrieval* dan kemiripan teks. Penerapan BM25 efektif sebagai tahap penyaringan awal untuk menemukan dokumen yang memiliki keterkaitan *term* dengan *query*, sedangkan kombinasi TF-IDF dan *cosine similarity* membantu menstabilkan pemeringkatan pada rekomendasi dosen melalui pembentukan profil publikasi. Mekanisme praproses (*case folding*, tokenisasi, *stopword removal*, serta *stemming* selektif) berperan dalam meningkatkan konsistensi representasi teks, sehingga mengurangi bias akibat variasi penulisan dan memperkecil dominasi kata-kata generik. Meski demikian, kualitas rekomendasi tetap dipengaruhi oleh kelengkapan metadata (misalnya abstrak/*keyword* yang kosong) dan cakupan korpus publikasi, sehingga pada kasus tertentu hasil rekomendasi dapat menurun ketika informasi dokumen tidak memadai atau istilah topik terlalu umum.

4.2. Kesimpulan

Penelitian ini menghasilkan sistem rekomendasi yang terintegrasi untuk membantu mahasiswa UDINUS pada tahap awal penelitian, khususnya dalam memperoleh referensi sitasi yang relevan serta menentukan dosen pembimbing yang sesuai dengan topik. Sistem dibangun menggunakan BM25 sebagai *baseline retrieval* untuk rekomendasi sitasi dan pendekatan hybrid untuk rekomendasi dosen melalui pemodelan TF-IDF dan cosine similarity berbasis profil publikasi, serta didukung oleh tahapan praproses teks yang terstruktur. Evaluasi menggunakan metrik berbasis peringkat menunjukkan bahwa sistem dapat menyajikan kandidat yang relevan pada urutan teratas, sementara deployment menggunakan Streamlit memungkinkan sistem diakses secara interaktif sebagai prototipe yang siap digunakan.

Referensi

Andri Setiawan, Z. A. (2025). Impact of Preprocessing on Indonesian Extractive Summarization Using LexRank, TextRank, DivRank, and Cosine Similarity. *G-Tech*, 2311-2321.

Dasri Dasria, A. A. (2025). Two-Way Thesis Supervisor Recommendation System Using MapReduce K-Skyband View Queries. *JOIV : International Journal on Informatics Visualization*, 163.

Mathias Uta, A. F.-M. (2024). Knowledge-Based Recommender Systems: Overview And Research Directions. *Frontiers in Big Data*, 1304439.

Nandan Thakur, N. R. (2021). BEIR: A Heterogenous Benchmark for Zero-shot Evaluation of Information Retrieval Models.

Robin Burke, A. F. (2025). Recommender Systems: An Overview. *AAAI*, 13-18.

Syarif, M., & Wiguna, W. (2023). Rekomendasi Dosen Pembimbing Skripsi menggunakan Metode Cosine Similarity. *Sistemasi: Jurnal Sistem Informasi*, 224-234.

Zitong Zhang, B. G. (2023). Scholarly Recommendation Systems: A Literature Survey. *Knowledge and Information Systems*, 4433-4478.

GeeksforGeeks. (2015, Juli 15). *Cosine Similarity*. Diambil kembali dari geeksforgeeks.org: https://www.geeksforgeeks.org/dbms/cosine-similarity/

GeeksforGeeks. (2025, Desember 17). *Understanding TF-IDF (Term Frequency-Inverse Document Frequency)*. Diambil kembali dari geeksforgeeks.org: https://www.geeksforgeeks.org/machine-learning/understanding-tf-idf-term-frequency-inverse-document-frequency/

GeeksforGeeks. (2025, Desember 16). *What is BM25 (Best Matching 25) Algorithm*. Diambil kembali dari geeksforgeeks.org: https://www.geeksforgeeks.org/nlp/what-is-bm25-best-matching-25-algorithm/