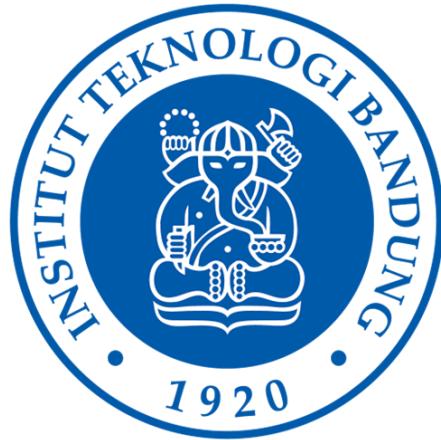


Tugas IF4042 Sistem Temu Balik Informasi

Sistem Temu Balik Informasi dan

Ekspansi *Query* dengan Word2Vec



Disusun oleh:

Kevin John Wesley Hutabarat	NIM. 13521042
Arleen Chrysantha Gunardi	NIM. 13521059
Moh. Aghna Maysan Abyan	NIM. 13521076
Austin Gabriel Pardosi	NIM. 13521084
Ryan Samuel Chandra	NIM. 13521140

TEKNIK INFORMATIKA
INSTITUT TEKNOLOGI BANDUNG
2025

1. Deskripsi Aplikasi

Ekspansi kueri (*query expansion*) adalah teknik yang umumnya digunakan pada proses temu balik informasi untuk meningkatkan kinerja temu balik melalui pengubahan bentuk kueri yang diberikan oleh pengguna, seperti menambahkan istilah baru atau memberikan bobot pada kueri awal (Vechtomova & Wang, 2006). Sementara itu, Word2Vec adalah teknik yang digunakan pada pemrosesan bahasa alami (NLP) untuk memperoleh vektor representasi dari sebuah kata berdasarkan konteks kata tersebut terhadap kata-kata disekitarnya (Mikolov et al., 2013).

Aplikasi ini adalah aplikasi yang dapat melakukan ekspansi kueri dengan teknik Word2Vec. Pada aplikasi ini, *tech-stack* yang digunakan adalah Next.js (didukung Typescript, Tailwind CSS, dan ShadCN UI) untuk *front-end* dan FastAPI (dengan bahasa pemrograman Python) untuk *back-end*. Aplikasi ini memungkinkan pengguna untuk:

- 1) Melakukan *stemming*
- 2) Melakukan *stop-word elimination*
- 3) Memilih metode pembobotan:
 - TF (*Term Frequency*): *logarithmic*, *binary*, *augmented*, atau *raw*
 - IDF (*Inverse Document Frequency*)
 - TF-IDF
 - TF-IDF-*Cosine normalization*
- 4) Memasukkan beberapa kata/*term* yang ingin ditambahkan atau seluruh kata ditambahkan

Selain itu, program yang telah dibangun untuk aplikasi ini dapat melakukan:

- 1) Menerima kueri secara interaktif maupun *batch*
- 2) Menampilkan kueri hasil ekspansi beserta bobot dan hasil pemeringkatannya
- 3) Menampilkan *inverted file* sebuah dokumen
- 4) Menampilkan hasil temu balik (pemeringkatan dokumen serta nilai *similarity measure*) dan MAP (*mean average precision*), baik tiap kueri maupun seluruh kueri untuk kueri yang belum melalui proses ekspansi maupun kueri yang sudah melalui proses ekspansi

2. Deskripsi Program

A. Deskripsi *Function*

1. Class QueryExpansionService

Kelas ini menangani kebutuhan *query expansion* dari sistem temu balik. Pada aplikasi ini, metode *query expansion* yang digunakan adalah *expansion* dengan Word2Vec. Koleksi dokumen yang dimiliki dilatih dengan model Word2Vec untuk dicari kemiripan antar katanya. Kelas ini memiliki fungsi-fungsi sebagai berikut.

a. `create`

Fungsi ini digunakan untuk menginisialisasi dan melatih model Word2Vec. Masukan dari fungsi ini adalah `document_path` yang berbentuk string. Format file yang dapat ditangani adalah dua jenis, yaitu CISI *file* dan JSON. Fungsi ini mengembalikan model Word2Vec yang akan digunakan untuk *query expansion*.

b. `ensure_model_trained`

Fungsi ini dibuat untuk memastikan model sudah dilatih sebelum digunakan. Jika model belum dilatih dan `document_path` diberikan, model akan dilatih terlebih dahulu. Fungsi ini menerima `document_path`, yaitu path ke dokumen untuk dilatih.

c. `train_word2vec_model`

Fungsi ini digunakan untuk melatih model Word2Vec dari dokumen. Fungsi ini menerima masukan `documents`, yang memiliki format Dict[str,str]. *Key* dari `documents` adalah id dokumen, sementara *value* nya adalah konten dokumen. Untuk setiap dokumen, kontennya akan dilakukan pra-pemrosesan terlebih dahulu sesuai dengan masukan pengguna. Dokumen yang sudah melalui pra-pemrosesan kemudian dilatih pada model Word2Vec dengan ukuran vektor 100, window berukuran 5 kata, frekuensi term minimum 5, 4 *thread*, dan model skip-gram.

d. **load_pretrained_model**

Fungsi ini digunakan untuk memuat model *pretrained* Word2Vec. Fungsi ini menerima masukan model_path berbentuk string. model_path adalah path yang mengandung model *pretrained*.

e. **expand_query**

Fungsi ini adalah fungsi utama dari ekspansi *query*. Fungsi ini melakukan ekspansi query dengan model Word2Vec. Masukan yang digunakan fungsi ini adalah query, threshold, dan limit. Query yang akan dicocokkan ke setiap dokumen disimpan di variabel query, sedangkan threshold digunakan sebagai ambang batas kemiripan dari pasangan kata untuk dianggap cukup mirip, dan limit adalah banyak kata yang ditambahkan ke dalam *query*. Untuk setiap term di dalam query, akan dicari term dari dokumen yang mirip dengan term tersebut. Jika argumen limit tidak kosong, term yang diambil hanyalah *term* teratas berdasarkan kemiripan sesuai dengan besarnya limit masukan. Fungsi ini menghasilkan *dictionary* berisi *query* asli dan *expanded*, serta term-term yang ditambahkan.

f. **get_similar_terms**

Fungsi ini digunakan untuk mendapatkan term-term yang mirip dengan term yang diberikan. Fungsi ini menerima argumen berupa term, yaitu kata yang ingin dicari kata-kata yang mirip dengannya. Selain itu, terdapat juga threshold yang berupa ambang batas nilai kemiripan untuk sebuah pasangan term dianggap mirip. Fungsi ini menghasilkan list dari term yang dianggap mirip, juga nilai kemiripannya.

g. **read_cisi_collection**

Fungsi ini digunakan untuk membaca koleksi CISI dan mengembalikan dokumen dalam bentuk *dictionary*. Fungsi ini menerima masukan berupa file_path, yaitu path letak dokumen berada. Fungsi ini membaca *file* dokumen dan mengubahnya ke dalam bentuk *dictionary* agar dapat digunakan oleh aplikasi.

h. `read_json_collection`

Fungsi ini digunakan untuk membaca dokumen berformat JSON dan mengembalikan dokumen dalam bentuk *dictionary*. Fungsi ini menerima masukan berupa `file_path`, yaitu path letak dokumen berada. Fungsi ini membaca *file* dokumen dan mengubahnya ke dalam bentuk *dictionary* agar dapat digunakan oleh aplikasi.

2. Class RetrievalService

Kelas ini menjadi inti dari sistem temu balik informasi, yaitu menangani kebutuhan pengembalian informasi (dokumen) berdasarkan kueri, baik yang dimasukkan secara interaktif maupun *batch*. Kelas ini akan digunakan untuk mengembalikan informasi berdasarkan kueri semula, serta kueri hasil *expansion* menggunakan *Word2Vec*. Kelas ini juga memanfaatkan fungsi *parsing* kueri berbentuk *batch* serta *relevance judgement* dari *file* “func_parser.py” pada folder “parsing”. Kelas ini memiliki fungsi-fungsi sebagai berikut.

a. `create_inverted_file`

➤ Masukan:

- `documents: Dict[str, Any]`: merupakan hasil *parsing* koleksi dokumen dengan fungsi `parser_docs` yang dilakukan oleh front-end, berformat kamus ID dokumen terhadap isi teksnya.
- `use_stemming: bool`: merupakan keputusan apakah akan menggunakan *porter stemming* terhadap setiap *term* atau tidak, dengan format *True/False*.
- `use_stopword_removal: bool`, merupakan keputusan apakah akan mengeliminasi setiap *stop word* pada dokumen atau tidak, dengan format *True/False*.
- `document_weighting_method: Dict[str, bool]`: kamus berisi metode-metode pembobotan yang tersedia (TF: *raw*, *log*, *binary*, *augmented*; IDF, *cosine normalization*) yang akan diterapkan ke dokumen terhadap *True/False* terkait digunakan atau tidak.

➤ Keluaran:

- `inverted_file: Dict[str, Dict[str, float]]`: kamus berisi *term* terhadap kamus lagi yang berisi ID dokumen terhadap bobot *term* tersebut dalam dokumen tersebut.

➤ Deskripsi/Proses:

1. Setiap *value* dari kamus masukan di-*preprocess* dengan menggunakan fungsi `preprocess_text()` agar *stemming* dan *stop word removal* dapat dilakukan jika masukannya *True*.
2. Perhitungan jumlah kemunculan masing-masing token dalam masing-masing dokumen dilakukan, menghasilkan *dictionary freq_file* sebagai masukan untuk `calculate_tf_idf()`.
3. `freq_file` diiterasi dan setiap *value*-nya (berisi kamus kata terhadap jumlahnya) diiterasi lagi untuk menghitung bobotnya menggunakan `calculate_tf_idf()`.
4. Kamus *inverted file* diciptakan dengan membalik `freq_file` dari formasi ID dokumen-*term*-jumlah kemunculan menjadi *term*-ID dokumen-bobot.

b. `calculate_tf_idf`

➤ Masukan:

- `term: str`: merupakan *term*/kata yang akan dihitung bobotnya pada dokumen tertentu.
- `doc: str`: merupakan ID dokumen tempat term tersebut berada.
- `freq_file: Dict[str, Any]`: merupakan *dictionary* yang berisi frekuensi kemunculan setiap term pada tiap dokumen. Format:
`{doc_id: Counter({term: freq, ...}), ...}`
- `weighting_method: Dict[str, bool]`: merupakan konfigurasi metode pembobotan yang digunakan (misalnya: `tf_raw`, `tf_log`, `tf_binary`, `tf_augmented`, `use_idf`, `use_normalization`).

➤ Keluaran:

- `Dict[str, Any]`: merupakan *dictionary* yang berisi term, dokumen, dan bobot hasil perhitungan. Format: {"term": term, "doc": doc, "weight": weight}.

➤ Deskripsi/Proses:

- Fungsi ini menghitung bobot sebuah *term* pada dokumen tertentu menggunakan metode pembobotan yang dipilih. Bobot-bobot tersebut digunakan dalam membentuk *inverted file*.
1. Proses dimulai dengan menghitung TF (*Term Frequency*) sesuai metode (*raw, log, binary, augmented*).
 2. Jika `use_idf` diaktifkan, maka dihitung juga IDF (*Inverse Document Frequency*) untuk *term* tersebut.
 3. Jika normalisasi diaktifkan, bobot akan dinormalisasi berdasarkan panjang dokumen.
 4. Hasil akhirnya adalah bobot akhir *term* pada dokumen yang akan digunakan dalam *inverted file*.

c. `calculate_query_weight`

➤ Masukan:

- `query: str`: merupakan isi (teks) *query* yang akan diproses, berupa *string*.
- `weighting_method: Dict[str, bool]`: merupakan konfigurasi metode pembobotan yang digunakan untuk *query*.
- `inverted_file: Dict[str, Dict[str, float]]`: merupakan *inverted file* yang sudah dibangun dari dokumen, alias keluaran dari fungsi `create_inverted_file()`.

➤ Keluaran:

- `query_vector: Dict[str, float]`: kamus berisi setiap *term* pada *query* terhadap bobotnya.

➤ Deskripsi/Proses:

- Mirip dengan `calculate_tf_idf()`, hanya saja khusus *query*.

d. `calculate_similarity`

➤ Masukan:

- `query_vector: Dict[str, float]`: keluaran dari fungsi `calculate_query_weight()`
- `document_vectors: Dict[str, Dict[str, float]]`: *inverted file*, alias keluaran dari fungsi `create_inverted_file()`.

➤ Keluaran:

- `query_docs_similarities: Dict[str, Any]`: kamus berisi nilai kemiripan dari *query* dengan setiap dokumen.

➤ Deskripsi/Proses:

- Fungsi ini menghitung nilai kemiripan antara vektor *query* dengan setiap vektor dokumen dengan melakukan operasi *dot product*.

e. `retrieve_document_single_query`

➤ Masukan

- `query: str`: merupakan *query* yang dimasukkan oleh *user*.
- `inverted_file: Dict[str, Dict[str, float]]`: merupakan *inverted file* hasil pembobotan dokumen.
- `weighting_method: Dict[str, bool]`: merupakan konfigurasi metode pembobotan untuk *query*.
- `relevant_doc: List[str]`: merupakan daftar ID dokumen relevan (untuk evaluasi, bisa kosong jika kueri interaktif).

➤ Keluaran

- `Tuple[Dict[str, float], float]`: merupakan *tuple* yang berisi *dictionary* dokumen terurut berdasarkan similaritasnya dengan *query*, dan nilai *average precision* (AP) jika tersedia.

➤ Deskripsi/Proses

- Fungsi ini melakukan proses *retrieval* untuk satu *query*.

1. *Query* diubah menjadi vektor, dihitung similaritasnya dengan dokumen, lalu dokumen diurutkan berdasarkan nilai similaritas.
2. Jika tersedia daftar dokumen relevan, maka dihitung juga *average precision* (AP) sebagai metrik evaluasi.

f. **retrieve_document_batch_query**

➤ Masukan

- **filename:** str: merupakan nama/lokasi file *batch query* mentah yang berisi beberapa *query*.
- **inverted_file:** Dict[str, Dict[str, float]]: merupakan *inverted file* hasil pembobotan dokumen.
- **weighting_method:** Dict[str, bool]: merupakan konfigurasi metode pembobotan untuk *query*.
- **relevant_doc_filename:** str: merupakan nama/lokasi *file* daftar dokumen relevan (mentah) untuk setiap *query*.

➤ Keluaran

- Tuple[List[Tuple[Dict[str, float], Tuple[str, str], float]], float]: merupakan *list* dokumen hasil *retrieval* untuk setiap *query* (berisi *similarity*, informasi ID dan konten *query*, dan AP untuk hasil *retrieval* tersebut), serta nilai *mean average precision* (MAP) untuk keseluruhan *retrieval* (semua kueri).

➤ Deskripsi/Proses

- Fungsi ini melakukan proses *retrieval* untuk sekumpulan *query* (*batch*).
- **Query yang tidak memiliki dokumen relevan di-skip.**
 1. File mentah *batch query* dan *relevant document* di-parse menggunakan fungsi `parser_query()` dan `parser_qrels()`.
 2. Kumpulan *query* hasil *parsing* diiterasi, dihitung similaritasnya dengan semua dokumen serta *average precision*-nya meng-

gunakan fungsi `retrieve_document_single_query()`, lalu hasilnya dikumpulkan.

3. Terakhir, *mean average precision* (MAP) dihitung sebagai metrik evaluasi keseluruhan *batch*.

g. **`retrieve_document_by_id`**

➤ Masukan

- `id: str`: merupakan ID dokumen yang ingin diambil.
- `documents: List[Dict[str, Any]]`: merupakan *list* dokumen yang tersedia.

➤ Keluaran

- `Dict[str, Any]`: merupakan *dictionary* berisi informasi dokumen (*author, title, content*) jika ditemukan, atau *dictionary* kosong jika tidak ditemukan.

➤ Deskripsi/Proses

- Fungsi ini mencari dokumen berdasarkan ID yang diberikan.
 1. Jika ditemukan, akan mengembalikan informasi dokumen (*author, title, content*).
 2. Jika tidak ditemukan, mengembalikan *dictionary* kosong.

h. **`retrieve_document_ids`**

➤ Masukan

- `documents: List[Dict[str, Any]]`: merupakan *list* dokumen yang tersedia.
- `ids: List[str]`: merupakan list ID dokumen yang ingin diambil.

➤ Keluaran

- `List[Dict[str, Any]]`: merupakan *list* informasi dokumen yang ditemukan berdasarkan ID yang diberikan.

➤ Deskripsi/Proses

- Fungsi ini mencari beberapa dokumen berdasarkan *list* ID yang diberikan.

1. Untuk setiap ID, fungsi akan memanggil *retrieve_document_by_id*.
2. Hasilnya adalah list dokumen yang ditemukan.

i. **get_weight_by_document_id**

➤ Masukan

- **document_id: str**: merupakan ID dokumen yang ingin diambil bobot *term*-nya.
- **inverted_file: Dict[str, Dict[str, float]]**: merupakan *inverted file* hasil pembobotan dokumen.

➤ Keluaran

- **Dict[str, float]**: merupakan *dictionary* yang memetakan setiap *term* pada dokumen dengan ID tersebut ke bobotnya. Format: *{term: weight, ...}*.

➤ Deskripsi/Proses

- Fungsi ini mengambil bobot setiap *term* yang terdapat pada dokumen tertentu dari *inverted file*.
- Setiap *term* yang memiliki *entry* untuk dokumen tersebut akan dimasukkan ke *dictionary* hasil.

B. Deskripsi *Endpoint*

Endpoint digunakan untuk merutekan semua permintaan yang masuk dari *front-end* ke *back-end* agar dapat ditangani dengan fungsi dan cara yang tepat. *Library* utama yang digunakan adalah FastAPI.

1. documents.py

Menampung *endpoint* dengan rute yang diawali oleh “/documents”. Semua *endpoint* berkaitan dengan operasi dokumen. Terdapat 4 rute sebagai berikut.

a. **GET "/list"**

Mengembalikan daftar ID dokumen beserta isinya dari *file* “parsing_docs.json” yang merupakan hasil *parsing* dari *endpoint POST* (“/parse”). *Exception* yang

digunakan adalah *file* tidak ditemukan (404), terjadi eror pada saat *parsing* (500), dan *internal server error* (500).

b. **POST "/upload"**

Mengunggah dokumen dengan masukan harus bertipe “file” dengan tipe input *form-data* (*HTTP multipart upload*). Respons yang dikembalikan adalah nama *file* yang dipilih.

c. **POST "/parse"**

Melakukan *parsing* dokumen pada direktori yang namanya (*string*) menjadi input bertipe *application/x-www-form-urlencoded*. Kembaliannya yaitu nama direktori.

d. **POST "/retrieve-by-ids"**

Mengembalikan dokumen dari *file* “parsing_docs_with_field.json” berdasarkan daftar ID yang dikirim oleh klien. Dokumen akan dikonversi menjadi *list of dictionaries*, lalu dilakukan pencocokan ID berdasarkan *request*. *Exception* yang digunakan adalah *file* tidak ditemukan (404), terjadi eror pada saat *parsing* (500), dan *internal server error* (500).

2. query.py

Menampung *endpoint* dengan rute yang diawali oleh “/query”. Terdapat 3 rute sebagai berikut.

a. **POST "/query/expand"**

Melakukan ekspansi kueri pada sebuah *string* kueri dengan Word2vec. *Exception* yang digunakan adalah *internal server error* (500).

b. **POST "/query/expand-batch"**

Melakukan ekspansi kueri pada beberapa kueri dalam *file* JSON. *Exception* yang digunakan adalah *file* tidak ditemukan (400) dan terjadi eror saat parsing (500).

c. **GET "/query/model-status"**

Melakukan pengecekan status model Word2vec. Status yang diberikan adalah *ready* (model sudah tersedia), *not_ready* (model belum tersedia), atau *error* (terdapat kesalahan saat memeriksa status model).

d. **POST "/query/retrain-model"**

Melatih ulang model Word2Vec secara dinamis tanpa perlu *restart* aplikasi, dengan parameter *preprocessing* (*stemming* dan *stopword removal*) yang dapat dikustomisasi. *Endpoint* ini akan mengembalikan detail konfigurasi sebelum dan sesudah *retraining*, serta status keberhasilan proses *retraining*.

e. **GET "/query/preprocessing-config"**

Mengembalikan konfigurasi *preprocessing* (seperti penggunaan *stemming* dan *stopword removal*) yang sedang digunakan oleh model Word2Vec saat ini. *Endpoint* ini memudahkan pengecekan konsistensi *preprocessing* antara proses pelatihan dan penggunaan model.

3. retrieval.py

Menampung *endpoint* dengan rute yang diawali oleh “/retrieval”. Terdapat 10 rute sebagai berikut.

a. **POST "/query/interactive"**

Endpoint ini adalah *placeholder* untuk *interactive query*. Mengembalikan pesan “Interactive query placeholder”.

b. **POST "/query/batch"**

Endpoint ini adalah *placeholder* untuk *batch query*. Mengembalikan pesan “Batch query placeholder”.

c. **POST "/retrieve"**

Endpoint ini digunakan untuk melakukan *retrieval* dokumen menggunakan *cached inverted file*. *Endpoint* ini dapat dipanggil jika sudah memanggil GET /inverted-file terlebih dahulu. Exception yang digunakan adalah *file* tidak tersedia (400), yaitu ketika *inverted file* belum dibuat. *Endpoint* ini akan menghasilkan peringkat dokumen, *average precision*, *total retrieved*, dan query yang digunakan.

d. **GET "/inverted-file"**

Membuat dan menyimpan sebuah *inverted file* berdasarkan dokumen “parsing_docs.json” yang telah di-parsing. Keluarannya mencakup *inverted file* yang telah disebutkan, serta jumlah dokumen, *term*, dan status *cache*. Ketika sudah menghasilkan *inverted file*, *file* tersebut disimpan di *cache*.

e. **GET "/cache/status"**

Memeriksa status *cache* untuk *inverted file*. Apabila *inverted file* masih terdapat di *cache*, kembaliannya adalah status tersedia jumlah *terms*, dan parameteranya. Sedangkan jika tidak ada di *cache*, kembaliannya adalah status tidak tersedia, dengan catatan agar *user* mengunjungi *endpoint* GET /inverted-file dahulu.

f. **DELETE "/inverted-file/cache"**

Menghapus semua hal yang berhubungan dengan *inverted file* dari *cache*. Kembaliannya adalah status berhasil.

g. **GET "/model-status"**

Memeriksa status model *Word2Vec*. Kembalian *default*-nya adalah status siap dengan tambahan *vocabulary size*. Ada dua jenis *exception* yang digunakan, yaitu *HTTPException* untuk menangkap eror dari FastAPI dan *Exception* biasa untuk menangkap eror yang lebih umum.

h. **POST "/retrieve-batch"**

Melakukan *batch retrieval* dengan melibatkan *cached inverted file*. Hanya bisa digunakan ketika `GET /inverted-file` sudah dipanggil dahulu. Membuat *instance* dari *class* “*RetrievalService*” dan menggunakan fungsi `retrieve_document_batch_query()`, lalu memformat ulang keluaran dari fungsi tersebut sehingga menjadi JSON / *dictionary* sesuai dengan yang diharapkan untuk ditampilkan di *front-end*. *Exception* yang digunakan adalah *internal server error* (500).

- i. **`GET "/document-weights/{document_id}"`**

Mengembalikan bobot (*weights*) seluruh *term* dalam sebuah dokumen dari *inverted file* yang telah dibuat sebelumnya. Untuk mengeksekusi *endpoint* ini, diperlukan pengecekan ketersediaan *inverted file*. Apabila tidak ditemukan *inverted file* dari sebuah dokumen, proses eksekusi dihentikan dengan mengembalikan pesan eror. Sedangkan, apabila ditemukan *inverted file* dari sebuah dokumen, proses eksekusi tetap berjalan. Oleh karena itu, *endpoint* ini bergantung pada *endpoint* `GET "/inverted-file"`.

- j. **`POST "/calculate-query-weight"`**

Endpoint ini digunakan untuk menghitung bobot setiap term pada query. Endpoint ini dapat ditembak jika sudah memanggil `GET /inverted-file` terlebih dahulu. *Endpoint* ini memiliki body request yang berisi *query* dan *weighting_method*. *Query* adalah *query text* yang akan dihitung bobotnya, sementara *weighting_method* adalah metode pembobotan yang digunakan untuk *query*. *Endpoint* ini mengembalikan *query*, vektor *query*, banyak kata, dan metode pembobotan yang digunakan.

C. Deskripsi *Library*

Library digunakan untuk membangun sistem *Information Retrieval* yang komprehensif dengan fitur *query expansion*, *document retrieval*, dan *text preprocessing*. Setiap library memiliki peran khusus dalam mendukung fungsionalitas yang optimal.

Backend Libraries:

1. FastAPI

Framework web modern untuk membangun REST API yang cepat dan efisien. FastAPI menyediakan *automatic API documentation, type hints validation, dan asynchronous request handling* untuk endpoint *retrieval* dan *query expansion*.

2. Uvicorn

ASGI server untuk menjalankan aplikasi FastAPI secara *production-ready*. Uvicorn mendukung *high-performance async/await operations* dan *concurrent request processing*.

3. Pydantic

Library untuk *data validation* dan *settings management* menggunakan Python *type annotations*. Pydantic memastikan *input/output* data sesuai dengan *schema* yang didefinisikan pada model *request/response*.

4. Python-multipart

Library untuk *handling multipart form data*, khususnya untuk file *upload operations*. Digunakan dalam endpoint *upload* dokumen dan *batch processing*.

5. NumPy

Library fundamental untuk *scientific computing* dan *numerical operations*. NumPy mendukung operasi matematis pada *vector* dan *matrix* untuk perhitungan *similarity score* dan *term weighting*.

6. SciPy

Library untuk *advanced mathematical functions* dan *scientific computing*. SciPy menyediakan algoritma optimisasi dan *statistical operations* untuk *information retrieval metrics*.

7. Gensim

Specialized library untuk topic modeling dan *document similarity analysis*. Gensim digunakan untuk implementasi Word2Vec model dalam *query expansion* dan *semantic similarity calculation*.

8. Scikit-learn

Machine learning library untuk data *preprocessing*, *feature extraction*, dan *evaluation metrics*. Scikit-learn mendukung TF-IDF *calculation*, *cosine similarity*, dan *performance evaluation metrics* seperti MAP (*Mean Average Precision*).

9. NLTK

Natural Language Processing toolkit untuk *text preprocessing operations*. NLTK menyediakan *tokenization*, *stemming* (Porter Stemmer), *stopword removal*, dan *corpus management* untuk bahasa Indonesia dan Inggris.

10. Pandas

Data manipulation and analysis library untuk *handling structured data*. Pandas digunakan untuk *document parsing*, *batch query processing*, dan *data formatting* dalam operasi *retrieval*.

Frontend Libraries:

1. Next.js

Framework React full-stack untuk membangun *interface pengguna* sistem *Information Retrieval*. Next.js menyediakan *server-side rendering*, *optimized bundling*, dan *routing system* untuk performa aplikasi *web* yang optimal dalam menampilkan hasil *retrieval* dan *query interface*.

2. Axios

HTTP client library untuk komunikasi antara *frontend* dan *backend API*. Axios menangani semua *request* ke *endpoint* FastAPI *including document upload*, *query submission*, *retrieval requests*, dan *real-time communication* untuk operasi *Information Retrieval*.

3. Radix UI Components

Collection of accessible UI components (Dialog, Select, Tabs, Tooltip, dll.) untuk membangun *interface* yang *user-friendly*. Radix UI mendukung *advanced search interface*, *filter options*, *document preview modals*, dan *accessibility features* untuk sistem *retrieval* yang *inclusive*.

4. Tailwind CSS

Utility-first CSS framework untuk *styling* yang *consistent* dan *responsive*. Tailwind CSS memungkinkan *rapid development* dari *search interface*, *result layouts*, dan *responsive design* yang optimal untuk berbagai *device* dalam mengakses sistem *Information Retrieval*.

3. Hasil Eksperimen

Tabel 3.1 Eksperimen 1 (interaktif)

Query asal	<i>What is the need for information consolidation, evaluation, and retrieval in scientific research?</i>
Stemming	False
Penghapusan stop word	False
Batas query expansion	10 kata
Metode pembobotan	TF (augmented) x IDF
Hasil query expansion	<i>what is the need for information consolidation evaluation and retrieval in scientific research features testing interface components evaluating acquisition smart mechanized synthesis experimental</i> { "evaluation": [{ "term": "features", "similarity": 0.9539157748222351 }, { "term": "testing", "similarity": 0.9484084248542786 }, { "term": "interface", "similarity": 0.9435595273971558 }] }

	<pre> }, { "term": "components", "similarity": 0.9430817365646362 }, { "term": "evaluating", "similarity": 0.9415867328643799 }, { "term": "acquisition", "similarity": 0.9389102458953857 }, { "term": "smart", "similarity": 0.934873104095459 }, { "term": "mechanized", "similarity": 0.9342643022537231 }, { "term": "synthesis", "similarity": 0.9291064143180847 }, { "term": "experimental", "similarity": 0.9278411865234375 }] } </pre>
Hasil retrieval dengan query asli	<p>1458 documents retrieved</p> <p>Document ID: 1358</p> <ul style="list-style-type: none"> • Rank #1 • Similarity: 16.26049 <p>System Design, Evaluation, and Costing by Herner, Saul</p> <p>The word "system" as applied to information programs and activities is one which is very vaguely defined. The purpose of this paper is to help clarify the concept and discuss it in the context of the librarian's conventional planning and administrative activities. This is done through a narration of the step-by-step procedures followed in the conceptualization and design of an actual library</p>

	<p>and information program. The steps involved are the following: definition of the purpose of the program, and financial and administrative constraints</p>
Hasil retrieval dengan expanded query	<p>1458 documents retrieved</p> <p>Document ID: 608</p> <ul style="list-style-type: none"> Rank #1 Similarity: 49.59661 <p>A new comparison Between Conventional Indexing (MEDLARS) and Automatic Text Processing (SMART) by Salton, G.</p> <p>A new testing process is described designed to compare conventional retrieval(MEDLARS) and automatic text analysis methods (SMART).. The results obtained with a collection of documents chosen independently of either SMART or MEDLARS indicate that a simple automatic extraction of keywords from document abstracts produces a 30 to 40 percent loss compared with MEDLARS indexing.. A replacement of the unranked Boolean searches used in MEDLARS by the standard ranked output normally provided by SMART reduces the loss</p>
Screenshot	<p>The screenshot shows the 'Information Retrieval' interface. On the left, there's a sidebar with 'Information Retrieval' and a 'Document Collection' section. The main area displays search results for the query 'What is the need for information consolidation, evaluation, and retrieval in scientific research?'. It includes tabs for 'Original Query Result', 'Expanded Query Result', and 'Inverted File'. Below the tabs, it shows '1458 documents retrieved' and two specific document entries:</p> <ul style="list-style-type: none"> Document ID: 120 • Rank #2 • Similarity: 15.83436 An Assessment of Quality in Graduate Education by Carter, A.M. Before this study was begun in the spring of 1964, serious deliberation was given to the question of American Council of Education sponsorship of evaluation of selected graduate programs of major universities that comprise an important segment of the Council's membership. There was no question about the need for doing in a systematic and objective way what necessarily goes on continually in any event, though usually in a piecemeal and more impressionistic way. Our Commission on Standards Objectives for Higher... Document ID: 120 • Rank #3 • Similarity: 14.68956 Design and Evaluation of Information Systems by Swanson, R.W. The co-jointing of "design" with "evaluation" that is called for by this chapter posed organizational and inclusion-exclusion problems for the author. In part, "design" and "evaluation"

The screenshot displays a search interface with three panels:

- Original query weights:** Shows the original query terms with weights: 'need': 1, 'inform': 1, 'consolid': 1, 'evalu': 1, 'rettriev': 1, 'inscientif': 1, 'research': 1.
- Expanded query weights:** Shows the expanded query terms with weights: 'evaluation': 0.75, 'features': 0.75, 'similarity': 0.75, 'testing': 0.75, 'interface': 0.75, 'acquisition': 0.75, 'smart': 0.75, 'mechan': 0.75, 'synthesis': 0.75, 'experiment': 0.75.
- Expanded query similarity:** Shows the expanded query terms with similarity scores: 'evaluation': [{ 'term': 'features', 'similarity': 0.953915774822351 }, { 'term': 'testing', 'similarity': 0.9484064248542786 }, { 'term': 'interface', 'similarity': 0.9435595273971558 }, { 'term': 'components', 'similarity': 0.9438617365646362 }, { 'term': 'evaluating', 'similarity': 0.9415867328643799 }, { 'term': 'acquisition', 'similarity': 0.9389192458653857 }]

Below these panels is an **Inverted File** panel titled "All Documents" which lists document IDs and their term frequencies:

```
{
  "1": {
    "*24": 2.3,
    "*30": 2.2,
    "*41": 2.1,
    "*43": 2.1,
    "*45": 2.2,
    "*47": 2.1,
    "*61": 2.1,
    "*89": 2.0,
    "*97": 2.0,
    "*100": 2.0,
    "*180": 2.1,
    "*111": 2.2,
    "*119": 2.4,
    "*135": 2.1,
    "*145": 2.0,
    "*188": 2.0,
    "*198": 2.0,
    "*228": 2.0,
    "*229": 2.1,
    "*265": 2.0,
    "*269": 2.2,
    "*274": 2.1,
    "*318": 2.1
  }
}
```

Tabel 3.2 Eksperimen 2 (interaktif)

Query asal	<i>Two Kinds of Power An Essay on Bibliographic Control</i>
Stemming	True
Penghapusan stop word	False
Batas query expansion	10 kata
Metode pembobotan	TF (logarithmic) x IDF

Hasil query expansion	<p><i>two kind of power an essay on bibliograph control theme outsid unambigu jump mesh uncrit discret commentari line-not curv</i></p> <pre>{ "essay": [{ "term": "theme", "similarity": 0.9790260195732117 }, { "term": "outsid", "similarity": 0.9787817001342773 }], "power": [{ "term": "unambigu", "similarity": 0.9753784537315369 }, { "term": "jump", "similarity": 0.9736353158950806 }, { "term": "mesh", "similarity": 0.9733381271362305 }, { "term": "uncrit", "similarity": 0.9729182720184326 }, { "term": "discret", "similarity": 0.9728000164031982 }, { "term": "commentari", "similarity": 0.9725228548049927 }, { "term": "line-not", "similarity": 0.9725039005279541 }, { "term": "curv", "similarity": 0.9724660515785217 }] }</pre>
------------------------------	---

	<p>]</p> <p>}</p>
Hasil retrieval dengan query asli	<p>1453 documents retrieved</p> <p>Document ID: 3</p> <ul style="list-style-type: none"> • Rank #1 • Similarity: 0.18350 <p>Two Kinds of PowerAn Essay on Bibliographic Control by Wilson, P.</p> <p>The relationships between the organization and control of writingsand the organization and control of knowledge and information willinevitably enter our story, for writings contain, along with much else, a great deal of mankind's stock of knowledge and information. Bibliographicalcontrol is a form of power, and if knowledge itself is a form of power,as the familiar slogan claims, bibliographical control is in a certain sensepower over power, power to obtain the knowledge recorded in writtenform. As writings are not simply, and not in</p>
Hasil retrieval dengan expanded query	<p>1453 documents retrieved</p> <p>Document ID: 669</p> <ul style="list-style-type: none"> • Rank #1 • Similarity: 248.99508 <p>Rapid Structure Searches via Permuted Chemical Line-Notations by Sorter, P.F.Granito, C.E.Gilmer, J.C.Gelberg, A.Metcalf, E.A.</p> <p>The Wiswesser chemical line-notation is an uniqueand unambiguous method of representing chemicalstructures by a linear series of letters, numbers,ampersands, and hyphens. These symbols are meaningful tochemists familiar with the notation and can be processedby automatic data processing (ADP) equipment.The uniqueness of the line-notation permits the useof alphanumerically arranged lists of notations fordictionary-type searches. This ordered arrangementpermits the rapid location of a specific compound or a specific</p>

Screenshot

Results for Two Kinds of Power An Essay on Bibliographic Control

Original Query Result Expanded Query Result Inverted File

1453 documents retrieved

Document ID: 403 • Rank #1 • Similarity: 193.41728
Two Kinds of PowerAn Essay on Bibliographic Control
by Wilson, R
The relationships between the organization and control of writingsand the organization and control of knowledge and information willinevitably enter our story. As writings contain, along with much else, a great deal of mankind's stock of knowledge and information. Bibliographicalcontrol is a form of power, and if knowledge itself is a form of poweras the familiar slogan claims, bibliographical control is in a certain sensepower over power, power to obtain the knowledge recorded in writtenform. As writings are not simply, and not in...

Document ID: 403 • Rank #2 • Similarity: 58.37476
Government Publications: A Guide to Bibliographic Tools
by Palic, V.M.
The organization of government at all levels - international, national,provincial or state, and local - has resulted in increasing governmentinfluence on the life of each citizen. Concomitant with this proliferation of published directives, regulations, reports,technical studies, and other informational issuances in such volume that noone engaged in a business or profession, no financial tycoon, educator/researcher, farmer, housewife, welfare recipient, or unemployed person canfunction without some reference to...

Document ID: 901 • Rank #3 • Similarity: 54.81527
Language and Information Selected Essays on their Theory and Application
by Barilek, Y.
At one time or another many authors must have faced the dilemma whether to gather their articles published on a certain topic and republishthem as a collection of essays or whether to rework them into an anthology book. I decided in favor of the first course with regard to the articles had written during the last fifteen years on language and informationin particular on the more technical and applied aspects, leaving for somenonre occasion my papers on the philosophy of language.

Results for Two Kinds of Power An Essay on Bibliographic Control

Original Query Result Expanded Query Result Inverted File

1453 documents retrieved

Document ID: 669 • Rank #1 • Similarity: 248.99508
Rapid Structure Searches via Permutated Chemical Line-Notations
by Sorter, P.F.Granito, C.E.Gallina, J.C.Gelberg, A.Merlo, A.E.A.
In Wiessner, chemical line-notation is an unstructured, non-alphanumeric method of representing chemical structures by a linear series of letters, numbers,underscores, and hyphens. These symbols are meaningful to chemists familiar with the notation and can be processedby automatic data processing (ADP) equipment. The uniqueness of the line notation permits the use of alphanumerically arranged lists of notations forstructure-type searches. This ordered arrangementpermits the rapid location of a specific compound or specific...

Document ID: 3 • Rank #2 • Similarity: 193.41728
Two Kinds of PowerAn Essay on Bibliographic Control
by Wilson, R
The relationships between the organization and control of writingsand the organization and control of knowledge and information willinevitably enter our story. As writings contain, along with much else, a great deal of mankind's stock of knowledge and information. Bibliographicalcontrol is a form of power, and if knowledge itself is a form of poweras the familiar slogan claims, bibliographical control is in a certain sensepower over power, power to obtain the knowledge recorded in writtenform. As writings are not simply, and not in...

Document ID: 516 • Rank #3 • Similarity: 172.49241
Problems in Information Retrieval:Logical Jumps in the Expression of Information
by Faradane, J.Russell, J.M.Yates-Mercer, R.A.
In a structured data base, such as that obtained when information is indexedin a format including explicit relations, retrieval of all relevant items inresponse to a question may, in some cases, be affected by techniquesofne structure. Considerations in the form of logical jumps, or the omission ofa concept with one relation out of a string of three concepts with twoentwined relations, have been investigated by two different methods, inorder to overcome one ofthese techniques. Thirty-two rules are proposedwhich could permit th...

Original query weights Expanded query weights Expanded query similarity

```
{
  "two": 1,
  "kind": 1,
  "power": 1,
  "mess": 1,
  "bibliograph": 1,
  "control": 1
}
```

```
{
  "two": 1,
  "kind": 1,
  "power": 1,
  "mess": 1,
  "bibliograph": 1,
  "control": 1,
  "theme": 1,
  "outside": 1,
  "unambigu": 1,
  "jump": 1,
  "mesh": 1,
  "uncrit": 1,
  "valasse": 1,
  "commentari": 1,
  "line-not": 1,
  "curv": 1
}
```

```
{
  "essay": [
    {
      "term": "theme",
      "similarity": 0.979268195732117
    },
    {
      "term": "outside",
      "similarity": 0.9787817001342773
    }
  ],
  "power": [
    {
      "term": "unambigu",
      "similarity": 0.9753784537316369
    },
    {
      "term": "jump",
      "similarity": 0.9736353158960806
    },
    {
      "term": "mess",
      "similarity": 0.9733361271362305
    }
  ]
}
```

Tabel 3.3 Eksperimen 3 (interaktif)

Query asal	<i>What is information science? Give definitions where possible</i>
Stemming	True
Penghapusan stop word	True
Batas query expansion	8 kata
Metode pembobotan	TF (raw) only
Hasil query expansion	<i>inform scienc give definit possibl notion therefor believ insight conceptu algebra automata suggest</i>

	<pre>{ "definit": [{ "term": "notion", "similarity": 0.9737148284912109 }], "give": [{ "term": "therefor", "similarity": 0.9699913859367371 }, { "term": "believ", "similarity": 0.9690499305725098 }, { "term": "insight", "similarity": 0.965623676776886 }, { "term": "conceptu", "similarity": 0.9652296900749207 }, { "term": "algebra", "similarity": 0.9650352597236633 }, { "term": "automata", "similarity": 0.9638199806213379 }], "possibl": [{ "term": "suggest", "similarity": 0.9624667763710022 }] }</pre>
Hasil retrieval dengan query asli	<p>888 documents retrieved</p> <p>Document ID: 85</p> <ul style="list-style-type: none"> • Rank #1 • Similarity: 17.00000

	<p>Information Science: Toward the Development of a True Scientific Discipline by Yovits, M. C.</p> <p>It is pointed out that if information science is to be considered a "true" science similar to physics or chemistry then it must have a set of concepts and an analytical expression which apply to the flow of information in a general way.. In several previous papers, the author and a colleague have described a model of a generalized information system which has wide, and perhaps universal applicability.. This paper elaborates on this model and indicates the range of its applicability.. Several fundamental quantities are defined specifically in a way which allows for quantification.. It is pointed out in this paper that this model can be the basis for the development of a "true" science of information with all of the necessary requirements for a science.. By the use of this model and the definition of a "true" science, the goals and requirements for a curriculum in information science are thus established.. Within this context, information is defined as data of value in decision making.. Quantitative measures of information can be obtained by relating information to specific observable actions which can be measured physically..</p>
Hasil retrieval dengan expanded query	<p>986 documents retrieved</p> <p>Document ID: 85</p> <ul style="list-style-type: none"> • Rank #1 • Similarity: 17.00000 <p>Information Science: Toward the Development of a True Scientific Discipline by Yovits, M. C.</p> <p>It is pointed out that if information science is to be considered a "true" science similar to physics or chemistry then it must have a set of concepts and an analytical expression which apply to the flow of information in a general way.. In several previous papers, the author and a colleague have described a model of a generalized information system which has wide, and perhaps universal applicability.. This paper elaborates on this model and indicates the range of its applicability.. Several fundamental quantities are defined specifically in a way</p>

Screenshot

The screenshot displays a search interface with two separate queries and their results.

Query 1: "What is information science? Give definitions where possible"

Results for Query 1:

- Original Query Result: Document ID: 468 • Rank #1 • Similarity: 17.00000
- Expanded Query Result: Information Science: Toward the Development of a True Scientific Discipline
- Inverted File

Query 2: "What is information science?"

Results for Query 2:

- Original Query Result: Document ID: 85 • Rank #1 • Similarity: 17.00000
- Expanded Query Result: Information Science: Toward the Development of a True Scientific Discipline
- Inverted File

Comparison View:

Original query weights	Expanded query weights	Expanded query similarity
<pre>{ "inform": 1, "science": 1, "what": 1, "definit": 1, "possible": 1 }</pre>	<pre>{ "inform": 1, "science": 1, "give": 1, "definit": 1, "possible": 1, "notion": 1, "theory": 1, "believe": 1, "insight": 1, "concept": 1, "algebra": 1, "automata": 1, "suggest": 1 }</pre>	<pre>[{ "definit": [{ "term": "notion", "similarity": 0.9737148284912109 }], "give": [{ "term": "theifor", "similarity": 0.9699913859367371 }, { "term": "believe", "similarity": 0.9698499385725898 }, { "term": "insight", "similarity": 0.965623676776886 }, { "term": "conceptu", "similarity": 0.9652296908749207 }] }]</pre>

Tabel 3.4 Eksperimen 4 (interaktif)

Query asal	<i>Representative examples of successful architectural solutions</i>
Stemming	True
Penghapusan stop word	True
Batas query expansion	4 kata
Metode pembobotan	IDF only
Hasil query expansion	<i>repres exempl success architectur solut harvard information-handl weapon pont</i> {

	<pre> "architectur": [{ "term": "harvard", "similarity": 0.9976713061332703 }, { "term": "information-handl", "similarity": 0.9976140856742859 }, { "term": "weapon", "similarity": 0.996739387512207 }, { "term": "pont", "similarity": 0.9966989159584045 }] } </pre>
Hasil retrieval dengan query asli	<p>275 documents retrieved</p> <p>Document ID: 7</p> <ul style="list-style-type: none"> • Rank #1 • Similarity: 203.97436 <p>Academic Library BuildingsA Guide to Architectural Issues and Solutions by Ellsworth, R.E.</p> <p>This book attempts to present representative examples of successful architectural solutions to the important problems librarians and architects face in planning new college and university library buildings or in remodeling and enlarging existing structures. It does not attempt to make case study evaluations, as was done by Ellsworth Mason for Brown and Yale. Nor does it present examples of unsuccessful solutions except to show how to avoid mistakes, and in these cases the libraries will not be identified.</p>
Hasil retrieval dengan expanded query	<p>208 documents retrieved</p> <p>Document ID: 1449</p> <ul style="list-style-type: none"> • Rank #1 • Similarity: 438.97986 <p>The Weapons Acquisition Process: An Economic Analysis by Peck, M.J.</p>

A distinctive feature of American weapons development and production is the use of private firms to carry forward most of the effort. This volume is primarily concerned with the government-business relationships within which these activities take place. Our title reflects our emphasis. Weapons Acquisition is defined to include the conception, development, and production of technically advanced weapons for ultimate use by the armed forces. Process emphasizes the flow of decisions and activities during weapons programs, including the

Screenshot

The screenshots illustrate a search interface with various configuration options and results for different queries.

Query Type:

- Interactive (selected)
- Batch

Query: Representative examples of success

Checkboxes:

- Apply Stemming (checked)
- Remove Stop Words (checked)

Query Expansion:

Word Limit: 4

Word limit value: 1 means no limitation for the expansion word counts

Similarity Threshold: 0.7

Document Weighting Method:

TF Method: raw

Query Weighting Method:

TF Method: raw

Results for Representative examples of successful architectural solutions

Expanded query terms: repres exempl success architectur solut harvard information-handl weapon pont

[View query details](#)

Original Query Result Expanded Query Result Inverted File

275 documents retrieved

Document ID: 7 • Rank #1 • Similarity: 203.97436
Academic Library BuildingsA Guide to Architectural Issues and Solutions
by Ellsworth, R.E.
The book attempts to present representative examples of successfularchitectural solutions to the important problems librarians andarchitects face in planning new college and university librarybuildings or in remodeling and enlarging existing structures. It doesnot attempt to make case study evaluations, as was done byEllsworth Mason for Brown and Yale. Nor does it present examplesof unsuccessful solutions except to show how to avoid mistakes, and in these cases the libraries will not be identified.

Document ID: 350 • Rank #2 • Similarity: 110.47945
The Design of Inquiring Systems
by Churchman, C.W.
"Design" is used throughout in its most generic sense to include planning, operations research, engineering design,architectural design, programming, budgeting, and all the otheractivities in which we consciously attempt to change ourselves and ourenvironment to improve the quality of our lives. So the book could beread as a philosophy of organization theory, or of architectural engineering design or of operations research, or of planning.The word "inquiry" suggests that the audience includes personsinterested in...

Document ID: 427 • Rank #3 • Similarity: 78.42186
The Validity of Subjective Probability of Success Forecastsby R & D Project Managers
by Sudweeks, F.D.
Models and techniques to aid management in planningcontrolling R&D projects frequently use subjective probabilityofsuccess forecasts as one of the major inputs. An experiment wasconducted at the research laboratories of Monsanto Company toassess the predictive validity and consistency of such forecasts.The results indicate that the eventual success or failure of certaintypes of R&D projects can be predicted by measuring the time shaped polled probability of success forecasts. Probability offorecasts appear to yield...

Query Type:

Interactive (selected)

Query: Representative examples of success

Checkboxes:

- Apply Stemming (checked)
- Remove Stop Words (checked)

Query Expansion:

Word Limit: 4

Word limit value: 1 means no limitation for the expansion word counts

Similarity Threshold: 0.7

Document Weighting Method:

TF Method: raw

Query Weighting Method:

TF Method: raw

Results for Representative examples of successful architectural solutions

Expanded query terms: repres exempl success architectur solut harvard information-handl weapon pont

[View query details](#)

Original Query Result Expanded Query Result Inverted File

208 documents retrieved

Document ID: 1449 • Rank #1 • Similarity: 438.97996
The Weapons Acquisition Process: An Economic Analysis
by Peck, M.J.
A distinctive feature of American weapons development and production is the use of private firms to carry forward most of the effort. This volume is primarily concerned with the government-business relationships within which these activities take place. Our title reflects our emphasis. Weapons Acquisition is defined to include the conception, development, and production of technically advanced weapons for ultimate use by the armed forces. Process emphasizes the flow of decisions and activities during weapons programs, including the...

Document ID: 7 • Rank #2 • Similarity: 193.12027
Academic Library BuildingsA Guide to Architectural Issues and Solutions
by Ellsworth, R.E.
The book attempts to present representative examples of successfularchitectural solutions to the important problems librarians andarchitects face in planning new college and university librarybuildings or in remodeling and enlarging existing structures. It doesnot attempt to make case study evaluations, as was done byEllsworth Mason for Brown and Yale. Nor does it present examplesof unsuccessful solutions except to show how to avoid mistakes, and in these cases the libraries will not be identified.

Document ID: 679 • Rank #3 • Similarity: 125.42292
A Chemical Structure Storage and Search System Developed at DuPont
by Gluck, D.J.
As early as 1961, we in the Engineering Department atDuPont recognized the need for a better system for storing chemical structure information for storage andsubsequent retrieval. We believed that current methodsand the then current development of notation systemswould not completely serve our chemists' long rangechemical identification needs. Accordingly, we studied and then developed a chemicalstructure storage and search system. Huber gave a goodreview of the various approaches and applications. To useit...

Original query weights

```
{
  "repres": 1,
  "exempl": 1,
  "success": 1,
  "architectur": 1,
  "solut": 1
}
```

Expanded query weights

```
{
  "repres": 1,
  "exempl": 1,
  "success": 1,
  "architectur": 1,
  "solut": 1,
  "harvard": 1,
  "information-handl": 1,
  "weapon": 1,
  "pont": 1
}
```

Expanded query similarity

```
[
  {
    "architectur": [
      {
        "term": "harvard",
        "similarity": 0.9976713861332703
      },
      {
        "term": "information-handl",
        "similarity": 0.9976149856742859
      }
    ],
    "term": "weapon",
    "similarity": 0.996739387512287
  },
  {
    "term": "pont",
    "similarity": 0.9966989159584045
  }
]
```

Tabel 3.5 Eksperimen 5 (interaktif)

Query asal	<i>Algorithms for Processing Partial Match Queries Using Word Fragments</i>
Stemming	False
Penghapusan stop word	True
Batas query expansion	6 kata
Metode pembobotan	TF (binary) x IDF
Hasil query expansion	<i>algorithms processing partial match queries using word fragments write exhibit symbol option regular suited</i> "partial": [{ "term": "write", "similarity": 0.9979393482208252 }, { "term": "exhibit", "similarity": 0.9976842999458313 }, { "term": "symbol", "similarity": 0.997544527053833 }, { "term": "option", "similarity": 0.9974923729896545 }, { "term": "regular", "similarity": 0.9974504113197327 } , "match": [{ "term": "suited", "similarity": 0.9973295331001282 }]]
Hasil retrieval dengan query asli	275 documents retrieved Document ID: 489 • Rank #1 • Similarity: 62.72565

	<p>Experiments in Book Indexing by Computer by Borko, Harold</p> <p>The most challenging task in preparing an index to a book is to select alland only those terms that are related to the text and are useful for relevancepurposes.. While a knowledgeable human can make the selection on an intuitivebasis, automatic indexing requires a precise operational criterion for definingand selecting good and useful index terms.. Two principles of selection areproposed:specification and selection of useful terms, and specification andexclusion of useless terms.. Because of the nebulous nature and</p>						
Hasil retrieval dengan expanded query	<p>302 documents retrieved</p> <p>Document ID: 712</p> <ul style="list-style-type: none"> • Rank #1 • Similarity:78.22364 <p>Technical-Abstracting Fundamentals.II. Writing Principles and Practices by Weil, B.H.Zarembra, I.Owen, H.</p> <p>Abstracts can serve their purpose best only if they arecarefully written to transmit important information toreaders quickly and accurately. This requires knowledgeof audience needs, habits, and desires; ability to identifythe key facts in the document; ability to organize thesefacts, to present them in the order best suited to theaudience; and ability to write the abstracts clearly,concisely, and in conformity with the style rules of themedium involved. Some of these abilities are inborn, butall can be learned by study, practice, and criticism.</p>						
Screenshot	<p>The screenshot shows a search interface with the following details:</p> <ul style="list-style-type: none"> Query Type: Interactive (radio button selected) Query: Algorithms for Processing Partial Word Limit: 6 Similarity Threshold: 0.7 Document Weighting Method: TF Method: binary Query Weighting Method: TF Method: binary <p>Results for Algorithms for Processing Partial Match Queries Using Word Fragments</p> <p>Expanded query terms: algorithms processing partial match queries using word fragments write exhibit symbol option regular suited</p> <table border="1"> <thead> <tr> <th>Original Query Result</th> <th>Expanded Query Result</th> <th>Inverted File</th> </tr> </thead> <tbody> <tr> <td>275 documents retrieved</td> <td>Document ID: 489 • Rank #1 • Similarity: 62.72565 Experiments in Book Indexing by Computer by Borko, Harold</td> <td></td> </tr> </tbody> </table> <p>The most challenging task in preparing an index to a book is to select alland only those terms that are related to the text and are useful for relevancepurposes.. While a knowledgeable human can make the selection on an intuitivebasis, automatic indexing requires a precise operational criterion for definingand selecting good and useful index terms.. Two principles of selection areproposed:specification and selection of useful terms, and specification andexclusion of useless terms.. Because of the nebulous nature and..</p> <p>Document ID: 62 • Rank #2 • Similarity: 62.71141 A Literature Search and File Organization Model by Leimkuhler, Ferdinand R.</p> <p>A principle of sequential optimization in search theory distributes thesearch effort at each stage so as to maximize the probability of targetdetection with the effort expended thus far. As an application of thisprinciple to the search of pertinent items in a literature file, the file itemsshould be arranged in decreasing order of the probability that an item willyield the information sought. Complete ordering in this manner may not befeasible, and it is proposed that the files be partially ordered in searchzones with some loss in search..</p> <p>Document ID: 581 • Rank #3 • Similarity: 62.71141 Structure and Effectiveness of The Citation Identifier, anOperational Computer Program for Automatic Identification of CaseCitations in Legal Literature by Borkowski, CasimirCepanic, LouisMartin, J. SperlingSallo, VirginiaTeu, Siegfried</p> <p>A computer program for automatic identification of "full form" citations in legal literature (e.g., Rutherford v. Geddes, 4 Wall. 220,18 L. Ed. 343; Southland Industries, Inc. v. Federal Communications Commission, 1938, 69 App. D.C. 82, 99 F.2d 117) has been developedby this group and is now operational..The level of performance of this program known as "full form" detection is 95 percent and the "cited documents" detection is 100 percent..The "cited documents" detection is 100 percent..</p>	Original Query Result	Expanded Query Result	Inverted File	275 documents retrieved	Document ID: 489 • Rank #1 • Similarity: 62.72565 Experiments in Book Indexing by Computer by Borko, Harold	
Original Query Result	Expanded Query Result	Inverted File					
275 documents retrieved	Document ID: 489 • Rank #1 • Similarity: 62.72565 Experiments in Book Indexing by Computer by Borko, Harold						

The screenshot shows a two-panel interface. The left panel is titled 'Information Retrieval' and contains a form for querying a document collection. It includes fields for 'Query Type' (set to 'Interactive'), 'Query' ('Algorithms for Processing Partial'), 'Apply Stemming' (unchecked), 'Remove Stop Words' (checked), 'Word Limit' (set to 6), 'Similarity Threshold' (set to 0.7), and 'Document Weighting Method' (set to 'TF Method: binary'). The right panel displays the results for the query 'algorithms processing partial match queries using word fragments write exhibit symbol option regular suited'. It shows the 'Original Query Result' and 'Expanded Query Result' (which includes terms like 'partial', 'write', 'symbol', 'option', 'regular', and 'exhibit') along with their similarity scores. Below this, a document summary for 'Technical Abstracting Fundamentals.II. Writing Principles and Practices' by West, E.M. is shown.

Tabel 3.6 Eksperimen 6 (batch)

File query	query.text
Query asal	<i>What problems and concerns are there in making up descriptive titles? What difficulties are involved in automatically retrieving articles from approximate titles? What is the usual relevance of the content of articles to their titles?</i>
Stemming	False
Penghapusan stop word	False
Batas query expansion	4 kata
Metode pembobotan	IDF only
Hasil query expansion	<i>what problems and concerns are there in making up descriptive titles what difficulties are involved in automatically retrieving articles from approximate titles what is the usual relevance of the content of articles to their titles hyperbolic clue nonresearch interrelated</i> { "retrieving": [{ "term": "hyperbolic",

	<pre> "similarity": 0.9933832287788391 }, { "term": "nonresearch", "similarity": 0.9921275973320007 }, { "term": "interrelated", "similarity": 0.9919673800468445 }], "usual": [{ "term": "clue", "similarity": 0.9921396374702454 }] } </pre>
Hasil retrieval dengan query asli	<p>MAP = 0.17900 1460 documents retrieved AP score: 0.32260</p> <p>Document ID: 589 • Rank #1 • Similarity: 451.05246</p> <p>Are Titles of Chemical Papers Becoming More Informative? by Tocatlian, Jacques J.</p>
Hasil retrieval dengan expanded query	<p>MAP = 0.05403 1460 documents retrieved AP score: 0.29576</p> <p>Document ID: 589 • Rank #1 • Similarity: 451.05246</p> <p>Are Titles of Chemical Papers Becoming More Informative? by Tocatlian, Jacques J.</p>

Screenshot

The screenshots show the 'Information Retrieval' interface. The left side of each screenshot displays the search parameters:

- Document Collection:** Choose File No file chosen
- Query Type:** Batch (selected)
- Query:** query.text
- Relevance judgement:** qrels.text
- Query Expansion:** Word Limit: 4
- Similarity Threshold:** 0.7
- Document Weighting Method:** TF Method: logarithmic

The right side shows the search results for the query "What problems and concerns are there in making up descriptive titles? What difficulties are involved in automatically retrieving articles from approximate titles? What is the usual relevance of the content of articles to their titles?".

Results (Top 10 documents):

Rank	Document ID	Title	Similarity
1	589	Are Titles of Chemical Papers Becoming More Informative?	451.05246
2	722	Information Transfer Limitations of Titles of Chemical Documents	342.73936
3	590	Information Transfer Limitations of Titles of Chemical Documents	342.73936
4	591	Information Transfer Limitations of Titles of Chemical Documents	342.73936
5	592	Information Transfer Limitations of Titles of Chemical Documents	342.73936
6	593	Information Transfer Limitations of Titles of Chemical Documents	342.73936
7	594	Information Transfer Limitations of Titles of Chemical Documents	342.73936
8	595	Information Transfer Limitations of Titles of Chemical Documents	342.73936
9	596	Information Transfer Limitations of Titles of Chemical Documents	342.73936
10	597	Information Transfer Limitations of Titles of Chemical Documents	342.73936

MAP score: original query 0.17900, expanded query 0.05403.

Tabel 3.7 Eksperimen 7 (batch)

File query	query.text
Query asal	<i>What problems and concerns are there in making up descriptive titles? What difficulties are involved in automatically retrieving articles from approximate titles? What is the usual relevance of the content of articles to their titles?</i>
Stemming	False
Penghapusan stop word	False
Batas query expansion	10 kata
Metode pembobotan	TF (augmented) x IDF x cosine normalization
Hasil query expansion	<i>what problems and concerns are there in making up descriptive titles what difficulties are involved in automatically retrieving articles from approximate titles what is the usual relevance of the content of articles to their titles hyperbolic clue nonresearch interrelated requester ends accordance proposes disseminated progresses</i> { "retrieving": [{

	<pre> "term": "hyperbolic", "similarity": 0.9933832287788391 }, { "term": "nonresearch", "similarity": 0.9921275973320007 }, { "term": "interrelated", "similarity": 0.9919673800468445 }, { "term": "requester", "similarity": 0.9919372797012329 }, { "term": "ends", "similarity": 0.9917370080947876 }, { "term": "accordance", "similarity": 0.9915167689323425 }, { "term": "proposes", "similarity": 0.9914995431900024 }, { "term": "progresses", "similarity": 0.9913545846939087 }], "usual": [{ "term": "clue", "similarity": 0.9921396374702454 }, { "term": "disseminated", "similarity": 0.9914236664772034 }] } </pre>
Hasil retrieval dengan query asli	<p>MAP = 0.12764</p> <p>1460 documents retrieved</p>

	<p>AP score: 0.10501</p> <p>Document ID: 856</p> <ul style="list-style-type: none"> • Rank #1 • Similarity: 0.04862 <p>Concept of an On-Line Computerized Library Catalog by Kilgour, Frederick G.</p>
Hasil retrieval dengan expanded query	<p>MAP = 0.03948</p> <p>1460 documents retrieved AP score: 0.08687</p> <p>Document ID: 1138</p> <ul style="list-style-type: none"> • Rank #1 • Similarity: 0.03921 <p>Relevance and Pertinence by Polushkin, V.A.</p>
Screenshot	<p>The screenshot displays two side-by-side search results pages from an information retrieval system. Both pages have identical header and sidebar sections, including a file selection dropdown, query type (Batch), relevance judgement, and other parameters like word limit and similarity threshold. The left page (Original Query Result) shows results for Document ID 856 with AP score 0.10501 and Rank #1. The right page (Expanded Query Result) shows results for Document ID 1138 with AP score 0.08687 and Rank #1. Both pages include a 'View query details' link and a 'Query 1 of 76' indicator. The main content area for each result page lists the document ID, rank, similarity, title, author, and a brief description of the paper's content.</p>

Original query weights	Expanded query weights	Expanded query similarity
<pre>{ "problem": 0.6666666666666666, "concern": 0.6666666666666666, "make": 0.6666666666666666, "describe": 0.6666666666666666, "title": 1, "shortcom": 0.6666666666666666, "involve": 0.6666666666666666, "automat": 0.6666666666666666, "retirev": 0.6666666666666666, "article": 0.8333333333333333, "from": 0.6666666666666666, "unusual": 0.6666666666666666, "relate": 0.6666666666666666, "content": 0.6666666666666666 }</pre>	<pre>{ "problem": 0.6666666666666666, "concern": 0.6666666666666666, "make": 0.6666666666666666, "describe": 0.6666666666666666, "title": 1, "shortcom": 0.6666666666666666, "involve": 0.6666666666666666, "automat": 0.6666666666666666, "retirev": 0.6666666666666666, "article": 0.8333333333333333, "from": 0.6666666666666666, "unusual": 0.6666666666666666, "relate": 0.6666666666666666, "content": 0.6666666666666666, "hyperbol": 0.6666666666666666, "clique": 0.6666666666666666, "nonrelated": 0.6666666666666666, "interval": 0.6666666666666666, "request": 0.6666666666666666, "end": 0.6666666666666666, "disseminate": 0.6666666666666666, "process": 0.6666666666666666, "dissimil": 0.6666666666666666, "nonreass": 0.6666666666666666 }</pre>	<pre>{ "retrieving": [{ "term": "hypotholic", "similarity": 0.9933832287788391 }, { "term": "nonresearch", "similarity": 0.9921275973320007 }, { "term": "interrelated", "similarity": 0.9919673800468445 }, { "term": "requester", "similarity": 0.9919372797812329 }, { "term": "end", "similarity": 0.99173700808947876 }, { "term": "accordance", "similarity": 0.9915167689323425 }] }</pre>

Tabel 3.8 Eksperimen 8 (batch)

File query	query.text
Query asal	<i>What problems and concerns are there in making up descriptive titles? What difficulties are involved in automatically retrieving articles from approximate titles? What is the usual relevance of the content of articles to their titles?</i>
Stemming	True
Penghapusan stop word	False
Batas query expansion	5 kata
Metode pembobotan	TF (logarithmic) x IDF
Hasil query expansion	<pre>{ "difficulti": [{ "term": "shortcom", "similarity": 0.9713085293769836 }, { "term": "ad", "similarity": 0.9629461765289307 }, { "term": "overlook", "similarity": 0.9626510739326477 }, { "term": "due", "similarity": 0.9622777104377747 }] }</pre>

	<pre>["usual": [{ "term": "narrow", "similarity": 0.9668431878089905 }, { "term": "equal", "similarity": 0.9618819952011108 }] }</pre>
Hasil retrieval dengan query asli	<p>MAP = 0.08170</p> <p>1460 documents retrieved</p> <p>AP score: 0.02883</p> <p>Document ID: 1295</p> <ul style="list-style-type: none"> • Rank #1 • Similarity: 0.49505 <p>Communication or Chaos? by Baker, D.B.</p>
Hasil retrieval dengan expanded query	<p>MAP = 0.06044</p> <p>1460 documents retrieved</p> <p>AP score: 0.27842</p> <p>Document ID: 17</p> <ul style="list-style-type: none"> • Rank #1 • Similarity: 176.89640 <p>Adventures in Librarianship by Voigt, M.J.</p>
Screenshot	<p>The screenshot shows a search interface with the following details:</p> <p>Query Type: Batch</p> <p>Query: query.text</p> <p>Relevance judgement: qrels.text</p> <p>Options: Apply Stemming (checked), Remove Stop Words (unchecked)</p> <p>Word Limit: 6</p> <p>Similarity Threshold: 0.7</p> <p>Document Weighting Method: logarithmic</p> <p>Query Weighting Method: TF only, IDF only, TF x IDF, TF x IDF x cosine normalization</p> <p>Results:</p> <p>Original Query Result:</p> <ul style="list-style-type: none"> 1460 documents retrieved • AP score: 0.02883 Document ID: 1295 • Rank #1 • Similarity: 0.49505 Communication or Chaos? by Baker, D.B. Effective transfer of scientific and technical information continues to be a pressing national problem. <p>Expanded Query Result:</p> <ul style="list-style-type: none"> MAP score original query 0.08170 expanded query 0.06044 Query 1 of 76 Document ID: 1295 • Rank #1 • Similarity: 0.49505 Communication or Chaos? by Baker, D.B. Effective transfer of scientific and technical information continues to be a pressing national problem. Document ID: 578 • Rank #2 • Similarity: 0.46840 Terse Literatures: I. Terse Conclusions by Bernier, C.L. Terse Conclusion: Prompt literatures of organized terseconclusions may increase ability to keep up in a subject,reduce need for translation, and make information availablepromptly. Document ID: 60 • Rank #3 • Similarity: 0.44874 Information Science: What Is It? by Borko, H. In seeking a new sense of identity, we ask, in this article, the question:What is information science? What does the information science do? Tentativeanswers to these questions are

Query Type

- Interactive
- Batch

Query query.text

Relevance judgment qrels.txt

Apply Stemming

Remove Stop Words

Query Expansion

Word Limit: 5

Word limit value: 1 means no limitation for the expansion word count

Similarity Threshold: 0.7

Document Weighting Method

TF Method: logarithmic

- TF only
- IDF only
- TF x IDF
- TF x IDF x cosine normalization

Query Weighting Method

Results for What problems and concerns are there in making up descriptive titles? What difficulties are involved in automatically retrieving articles from approximate titles? What is the usual relevance of the content of articles to their titles?

Expanded query terms: what problem and concern are there in make up descript titl what difficulti are involv ad overllok due equal

View query details

Original Query Result Expanded Query Result Inverted File

1460 documents retrieved • AP score: 0.27842

Document ID: 17 - Rank #1 • Similarity: 176.89640
Adventures in Librarianship by Voigt, M.J.

There has long been a need for a continuing series to provide scholarly reviews of the rapidly changing and advancing field of librarianship, a series which would select subjects with particular current significance to the profession and provide an analysis of the advances made through research and practice. Advances in librarianship is planned and designed to fill this need. It will present critical articles and surveys based on the published literature, research in progress, and developments in libraries of all types. Mechanization may appear to be...

Document ID: 1090 - Rank #2 • Similarity: 174.31291
Library Optimum by Sanderson, A.

Sir, In his recent article B.C. Brookes propounds an ingenious mathematical framework to determine which periodical volumes a library should hold. He is careful to point out that the selection will need regular review and revision to take the value of the aging factor or the contents of the Bradford set change from year to year. There is as yet very little experimental evidence on the consistency of either, such limited evidence as there is suggests that the aging factors reasonably constant. But the position of the Bradford set is less...

Original query weights

```
{
  "problem": 1,
  "concern": 1,
  "make": 1,
  "descript": 1,
  "titl": 3,
  "difficulti": 1,
  "involv": 1,
  "retriev": 1,
  "articl": 2,
  "fromapproxim": 1,
  "usual": 1,
  "relev": 1,
  "content": 1
}
```

Expanded query weights

```
{
  "problem": 1,
  "concern": 1,
  "make": 1,
  "descript": 1,
  "titl": 3,
  "difficulti": 1,
  "involv": 1,
  "retriev": 1,
  "articl": 2,
  "fromapproxim": 1,
  "usual": 1,
  "relev": 1,
  "content": 1,
  "shortcom": 1,
  "narrow": 1,
  "ad": 1,
  "overlook": 1,
  "due": 1,
  "equal": 1
}
```

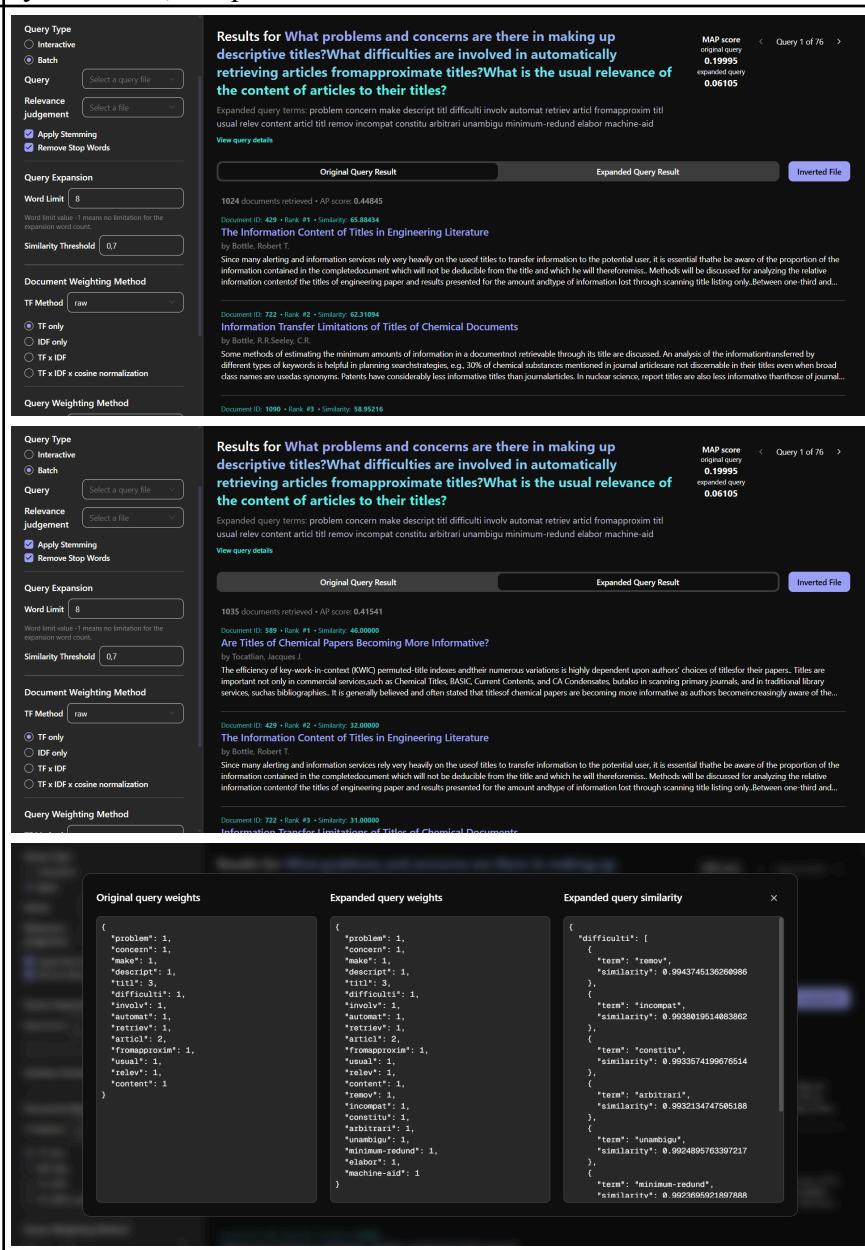
Expanded query similarity

```
{
  "difficulti": [
    {
      "term": "shortcom",
      "similarity": 0.9713085293769836
    },
    {
      "term": "ad",
      "similarity": 0.9629461765289397
    },
    {
      "term": "overlook",
      "similarity": 0.9626510739326477
    },
    {
      "term": "due",
      "similarity": 0.962277104377747
    }
  ],
  "usual": [
    {
      "term": "narrow",
      "similarity": 0.9668431878869985
    }
  ]
}
```

Tabel 3.9 Eksperimen 9 (batch)

File query	query.text
Query asal	<i>What problems and concerns are there in making up descriptive titles? What difficulties are involved in automatically retrieving articles from approximate titles? What is the usual relevance of the content of articles to their titles?</i>
Stemming	True
Penghapusan stop word	True
Batas query expansion	8 kata
Metode pembobotan	TF (raw) only
Hasil query expansion	<p><i>problem concern make descript titl difficulti involv automat retriev articl fromapproxim titl usual relev content articl titl remov incompat constitu arbitrari unambigu minimum-redund elabor machine-aid</i></p> <pre>{ "difficulti": [{ "term": "remov", "similarity": 0.9943745136260986 }] }</pre>

	<pre> }, { "term": "incompat", "similarity": 0.9938019514083862 }, { "term": "constitu", "similarity": 0.9933574199676514 }, { "term": "arbitrari", "similarity": 0.9932134747505188 }, { "term": "unambigu", "similarity": 0.9924895763397217 }, { "term": "minimum-redund", "similarity": 0.9923695921897888 }, { "term": "elabor", "similarity": 0.9923436641693115 }, { "term": "machine-aid", "similarity": 0.9920647144317627 }] } </pre>
Hasil retrieval dengan query asli	<p>MAP = 0.19995</p> <p>1024 documents retrieved</p> <p>AP score: 0.44845</p> <p>Document ID: 429</p> <ul style="list-style-type: none"> • Rank #1 • Similarity: 65.88434 <p>The Information Content of Titles in Engineering Literature by Bottle, Robert T.</p>
Hasil retrieval dengan expanded query	<p>MAP = 0.06105</p> <p>1035 documents retrieved</p> <p>AP score: 0.41541</p>

<p>Screenshot</p>	<p>Document ID: 589 • Rank #1 • Similarity: 46.00000</p> <p>Are Titles of Chemical Papers Becoming More Informative? by Tocatlian, Jacques J.</p>						
	 <p>The screenshot displays a search interface with three separate query results, each consisting of a query form, document details, and expanded query results. The results are as follows:</p> <ul style="list-style-type: none"> Result 1 (Document ID: 589): <ul style="list-style-type: none"> Query Form: Query Type (Batch), Query (Select a query file), Relevance judgement (Select a file), Apply Stemming (checked), Remove Stop Words (checked). Document Details: Document ID: 589 • Rank #1 • Similarity: 46.00000 Expanded Query Result: Results for What problems and concerns are there in making up descriptive titles? What difficulties are involved in automatically retrieving articles from approximate titles? What is the usual relevance of the content of articles to their titles? Similarity Metrics: MAP score original query 0.19995, expanded query 0.06105. Result 2 (Document ID: 722): <ul style="list-style-type: none"> Query Form: Query Type (Batch), Query (Select a query file), Relevance judgement (Select a file), Apply Stemming (checked), Remove Stop Words (checked). Document Details: Document ID: 722 • Rank #2 • Similarity: 62.31094 Expanded Query Result: The Information Content of Titles in Engineering Literature by Bottle, Robert T. Similarity Metrics: MAP score original query 0.19995, expanded query 0.06105. Result 3 (Document ID: 1090): <ul style="list-style-type: none"> Query Form: Query Type (Batch), Query (Select a query file), Relevance judgement (Select a file), Apply Stemming (checked), Remove Stop Words (checked). Document Details: Document ID: 1090 • Rank #3 • Similarity: 58.95216 Expanded Query Result: Information Transfer Limitations of Titles of Chemical Documents by Bottle, R.R.Seeley, C.R. Similarity Metrics: MAP score original query 0.19995, expanded query 0.06105. <p>Below the three results, there is a detailed view of the expanded query weights and similarity for Document ID 589:</p> <table border="1"> <thead> <tr> <th>Original query weights</th> <th>Expanded query weights</th> <th>Expanded query similarity</th> </tr> </thead> <tbody> <tr> <td> <pre>{ "problem": 1, "concern": 1, "make": 1, "descrip": 1, "title": 3, "difficult": 1, "involv": 1, "automat": 1, "retirev": 1, "usual": 2, "fromapproxim": 1, "usual": 1, "relev": 1, "content": 1 }</pre> </td> <td> <pre>{ "problem": 1, "concern": 1, "make": 1, "descrip": 1, "title": 3, "difficult": 1, "involv": 1, "automat": 1, "retirev": 1, "usual": 2, "fromapproxim": 1, "usual": 1, "relev": 1, "content": 1, "incompat": 1, "constitu": 1, "arbitrari": 1, "unambigu": 1, "minimum-redund": 1, "elabor": 1, "machine-aid": 1 }</pre> </td> <td> <pre>[{ "difficult": [{ "term": "remov", "similarity": 0.9943745136260966 }, { "term": "incompat", "similarity": 0.9938819514063862 }, { "term": "constitu", "similarity": 0.9933574199676514 }, { "term": "arbitrari", "similarity": 0.993213474565188 }, { "term": "unambigu", "similarity": 0.9924895763397217 }], "term": "minimum-redund", "similarity": 0.9923695921807868 }]</pre> </td> </tr> </tbody> </table>	Original query weights	Expanded query weights	Expanded query similarity	<pre>{ "problem": 1, "concern": 1, "make": 1, "descrip": 1, "title": 3, "difficult": 1, "involv": 1, "automat": 1, "retirev": 1, "usual": 2, "fromapproxim": 1, "usual": 1, "relev": 1, "content": 1 }</pre>	<pre>{ "problem": 1, "concern": 1, "make": 1, "descrip": 1, "title": 3, "difficult": 1, "involv": 1, "automat": 1, "retirev": 1, "usual": 2, "fromapproxim": 1, "usual": 1, "relev": 1, "content": 1, "incompat": 1, "constitu": 1, "arbitrari": 1, "unambigu": 1, "minimum-redund": 1, "elabor": 1, "machine-aid": 1 }</pre>	<pre>[{ "difficult": [{ "term": "remov", "similarity": 0.9943745136260966 }, { "term": "incompat", "similarity": 0.9938819514063862 }, { "term": "constitu", "similarity": 0.9933574199676514 }, { "term": "arbitrari", "similarity": 0.993213474565188 }, { "term": "unambigu", "similarity": 0.9924895763397217 }], "term": "minimum-redund", "similarity": 0.9923695921807868 }]</pre>
Original query weights	Expanded query weights	Expanded query similarity					
<pre>{ "problem": 1, "concern": 1, "make": 1, "descrip": 1, "title": 3, "difficult": 1, "involv": 1, "automat": 1, "retirev": 1, "usual": 2, "fromapproxim": 1, "usual": 1, "relev": 1, "content": 1 }</pre>	<pre>{ "problem": 1, "concern": 1, "make": 1, "descrip": 1, "title": 3, "difficult": 1, "involv": 1, "automat": 1, "retirev": 1, "usual": 2, "fromapproxim": 1, "usual": 1, "relev": 1, "content": 1, "incompat": 1, "constitu": 1, "arbitrari": 1, "unambigu": 1, "minimum-redund": 1, "elabor": 1, "machine-aid": 1 }</pre>	<pre>[{ "difficult": [{ "term": "remov", "similarity": 0.9943745136260966 }, { "term": "incompat", "similarity": 0.9938819514063862 }, { "term": "constitu", "similarity": 0.9933574199676514 }, { "term": "arbitrari", "similarity": 0.993213474565188 }, { "term": "unambigu", "similarity": 0.9924895763397217 }], "term": "minimum-redund", "similarity": 0.9923695921807868 }]</pre>					

Tabel 3.10 Eksperimen 10 (batch)

File query	query.text
-------------------	------------

Query asal	<i>Image recognition and any other methods of automatically transforming printed text into computer-ready form.</i>
Stemming	True
Penghapusan stop word	True
Batas query expansion	4 kata
Metode pembobotan	IDF only
Hasil query expansion	<pre> <i>imag recognit ani method automaticallytransform print text computer-readi form nonbook predomin self-contain coincid</i> { "imag": [{ "term": "nonbook", "similarity": 0.9971915483474731 }, { "term": "predomin", "similarity": 0.9971434473991394 }, { "term": "self-contain", "similarity": 0.9970644116401672 }, { "term": "coincid", "similarity": 0.9970524311065674 }] } </pre>
Hasil retrieval dengan query asli	<p>MAP = 0.19981 566 documents retrieved AP score: 0.06339</p> <p>Document ID: 320 • Rank #1 • Similarity: 145.07714</p> <p>The Teachable Language Comprehender: A Simulation Program and Theory of Language by Quillian, M.R.</p>
Hasil retrieval dengan expanded query	<p>MAP = 0.05823 575 documents retrieved AP score: 0.04457</p>

<p>Screenshot</p>	<p>Document ID: 320</p> <ul style="list-style-type: none"> Rank #1 Similarity: 145.07714 <p>The Teachable Language Comprehender:A Simulation Program and Theory of Language by Quillian, M.R.</p>
<div style="display: flex; justify-content: space-between;"> <div style="flex: 1;"> <p>Query Type <input type="radio"/> Interactive <input checked="" type="radio"/> Batch</p> <p>Query <input type="text" value="query.text"/></p> <p>Relevance judgement <input type="text" value="grels.text"/></p> <p><input checked="" type="checkbox"/> Apply Stemming <input checked="" type="checkbox"/> Remove Stop Words</p> <p>Query Expansion <input type="text" value="4"/> <small>Word limit value = 1 means no limitation for the expansion word count</small></p> <p>Similarity Threshold <input type="text" value="0.7"/></p> <p>Document Weighting Method <input type="radio"/> TF Method <input checked="" type="radio"/> raw</p> <p><input type="radio"/> TF only <input checked="" type="radio"/> IDF only <input type="radio"/> TF x IDF <input type="radio"/> TF x IDF x cosine normalization</p> <p>Query Weighting Method</p> </div> <div style="flex: 1;"> <p>Results for Image recognition and any other methods of automaticallytransforming printed text into computer-ready form.</p> <p>MAP score original query 0.19981 expanded query 0.05823</p> <p>Original Query Result Expanded Query Result Inverted File</p> <p>566 documents retrieved • AP score: 0.06339</p> <p>Document ID: 320 • Rank #1 • Similarity: 145.07714 The Teachable Language Comprehender:A Simulation Program and Theory of Language by Quillian, M.R.</p> <p>The Teachable Language Comprehender (TLC) is a programdesigned to be capable of being taught to "comprehend"English text. When text which the program has not seen before input to it, it comprehends that text by correctly relatingeach (explicit or implicit) assertion of the new text to a largememory. This memory is a "semantic network" representingfactual assertions about the world.The program also creates copies of the parts of its memorywhich have been found to relate to the new text, adaptingand combining...</p> <p>Document ID: 1224 • Rank #2 • Similarity: 91.05328 On one model of semantic information theory by Shneider, Y.A.</p> <p>Text processing problems (such as automatic translation and automaticabtracting) create a need for defining explicit concepts, which should becharacterized as the properties and quantity of semantic informationcontained in document texts.In fact, we need a formal model, which lets us describe the process ofsemantic text analysis.Semantic text analysis could be described from the point of view of something a different "conception of the world" - e.g. the text of very meaningfularticle does not contain, in fact, any information ...</p> <p>Document ID: 890 • Rank #3 • Similarity: 76.77459 Pattern Recognition and Structure-Activity Relationship Studies.Computer-Assisted Prediction of Antitumor Activity in Structurally DiverseDrugs in an Experimental Mouse Brain Tumor System by Chu, K., C.Feldman, R. J.Shapiro, M. B.Hazard, G. F. Jr.Geran, R. I.</p> <p>Results for Image recognition and any other methods of automaticallytransforming printed text into computer-ready form.</p> <p>MAP score original query 0.19981 expanded query 0.05823</p> <p>Original Query Result Expanded Query Result Inverted File</p> <p>575 documents retrieved • AP score: 0.04457</p> <p>Document ID: 320 • Rank #1 • Similarity: 145.07714 The Teachable Language Comprehender:A Simulation Program and Theory of Language by Quillian, M.R.</p> <p>The Teachable Language Comprehender (TLC) is a programdesigned to be capable of being taught to "comprehend"English text. When text which the program has not seen before input to it, it comprehends that text by correctly relatingeach (explicit or implicit) assertion of the new text to a largememory. This memory is a "semantic network" representingfactual assertions about the world.The program also creates copies of the parts of its memorywhich have been found to relate to the new text, adaptingand combining...</p> <p>Document ID: 320 • Rank #2 • Similarity: 114.65397 Developing Multi-Media Libraries by Hicks, W.B.</p> <p>This book presents the concept of the modern library as a comprehensiveresource center. The philosophy and objectives of the center are clarifiedand desirable practices in the selection and acquisition of nonbook oraudiovisual materials - interchangeably defined as those materials thatcommunicate primarily through aural and visual stimuli - are recommended,along with information pertinent to facilitating these tasks. Theirorganization in general is discussed, with emphasis on the necessity forbasic decisions and policies...</p> <p>Document ID: 992 • Rank #3 • Similarity: 110.47945 Cataloging Nonbook Materials: Mountain or Moleshill? by Massonneau, Suzanne</p> <p>The development of cataloging codes for nonbook materials in surveyed, withparticular attention devoted to the absence of stated objectives, the problem ofthe integrated catalog,</p> <div style="display: flex; justify-content: space-around; margin-top: 10px;"> <div style="width: 33%;"> <p>Original query weights</p> <pre>{ "img": 1, "recognit": 1, "anit": 1, "method": 1, "automaticallytransform": 1, "print": 1, "text": 1, "computer-readi": 1, "form": 1 }</pre> </div> <div style="width: 33%;"> <p>Expanded query weights</p> <pre>{ "img": 1, "recognit": 1, "anit": 1, "method": 1, "automaticallytransform": 1, "print": 1, "text": 1, "computer-readi": 1, "form": 1, "nonbook": 1, "predomin": 1, "self-contain": 1, "coincid": 1 }</pre> </div> <div style="width: 33%;"> <p>Expanded query similarity</p> <pre>[{ "term": ["img", "nonbook", "similarity": 0.9971915483474731], { "term": "predomin", "similarity": 0.9971434473991394 }, { "term": "self-contain", "similarity": 0.997064416461672 }, { "term": "coincid", "similarity": 0.9970524311865674 }] }</pre> </div> </div> </div> </div>	

4. Analisis Hasil Eksperimen

Berdasarkan hasil eksperimen pada bagian sebelumnya, dapat disimpulkan bahwa skor MAP untuk *expanded query* lebih rendah daripada skor MAP untuk *query* asli. Secara teoretis, *query expansion* seharusnya meningkatkan nilai skor MAP. Faktor yang menyebabkan penurunan nilai skor MAP adalah kurang akuratnya model Word2Vec yang digunakan untuk melakukan *query expansion*. Seperti yang dapat diamati pada hasil eksperimen, misalnya Eksperimen 6, kata-kata yang ditambahkan pada *query expansion* kurang memiliki kesinambungan secara makna dengan kata-kata pada *query* asli. Hal ini disebabkan oleh mekanisme Word2Vec yang melihat kemiripan kata berdasarkan kemunculan dari pasangan kata di dalam *window* yang berukuran tertentu. Konfigurasi dari ukuran *window* memengaruhi akurasi dari Word2Vec dalam menentukan kemiripan dari pasangan kata. Ada kemungkinan bahwa pemilihan ukuran *window* dari Word2Vec kurang tepat, sehingga kata-kata yang muncul bersamaan kurang menggambarkan hubungan antarkata tersebut. Kemungkinan lainnya adalah koleksi dokumen yang digunakan untuk melatih model Word2Vec kurang baik, sehingga kata-kata yang memperoleh kemiripan tinggi hanya menggambarkan kemunculan kata secara bersamaan yang sering muncul, bukan menunjukkan kemiripan kata secara arti.

LAMPIRAN

Lampiran A: Kode Program

- Repozitori *Front-End*: <https://github.com/arleenchr/IR-System-FE>
- Repozitori *Back-End*: <https://github.com/AustinPardosi/IR-System-BE>

Lampiran B: Panduan Penggunaan Aplikasi

Aplikasi terdiri dari dua komponen, yaitu *front-end* dan *back-end*. Keduanya harus dijalankan pada *localhost*.

Panduan untuk menjalankan *back-end*

1. Menggunakan *virtual environment*

- Jalankan perintah berikut untuk membuat *virtual environment*.

```
# Windows  
python -m venv venv  
source venv/Scripts/activate  
  
# Linux/Mac  
python3 -m venv venv  
source venv/bin/activate
```

- Jalankan perintah berikut untuk melakukan instalasi *dependency*.

```
pip install -r requirements.txt
```

2. Menggunakan *docker*

- Jalankan perintah berikut untuk memulai layanan *back-end*.

```
# Menjalankan docker  
docker-compose up  
  
# Menjalankan docker di background  
docker-compose up -d
```

- Jalankan perintah berikut untuk menghentikan layanan *back-end*.

```
docker-compose down
```

3. Setelah dijalankan, layanan *back-end* dapat diakses melalui <https://localhost:8080>.

Panduan untuk menjalankan *front-end*

1. Melakukan instalasi *library* dan *dependency* sebelum memulai dengan menjalankan perintah berikut.

```
npm install
```

2. Membuat konfigurasi *environment* dengan membuat *file* dengan nama `.env` pada direktori *root*. Adapun isi *file* adalah sebagai berikut.

```
API_BASE_URL=http://localhost:8080/api
```

3. Menjalankan aplikasi pada lingkungan pengembangan (*dev*) dengan perintah berikut.

```
npm run dev
```

4. Aplikasi web dapat diakses melalui <https://localhost:3000>.

Panduan untuk menggunakan aplikasi

Antarmuka aplikasi hanya terdiri dari satu halaman yang terdiri dari dua bagian utama, yaitu *toolbar* pada bagian kiri dan bagian untuk menampilkan hasil *retrieval* dokumen (lihat gambar C.1).

Toolbar menyediakan tempat untuk memasukkan *input*, seperti yang ditunjukkan Gambar C.2. Beberapa masukan yang dibutuhkan untuk melakukan *retrieval* adalah sebagai berikut.

1. Koleksi dokumen, berupa *file* yang memuat seluruh dokumen.
2. Jenis *query* (interaktif dan *batch*),
 - a. Teks *query* (jika interaktif)
 - b. *File query* beserta *file relevance judgement* (jika *batch*)
3. Pilihan untuk menerapkan *stemming* dan penghapusan *stop words*
4. Pengaturan untuk *query expansion*

- a. Batas (*limit*) banyaknya kata yang akan ditambahkan pada *query expansion*
 - b. Nilai ambang kemiripan (*similarity threshold*) untuk kata yang akan ditambahkan pada *query expansion*
5. Metode pembobotan dokumen
 - a. Metode TF, yaitu *logarithmic*, *raw*, *binary*, atau *augmented*
 - b. Pilihan kalkulasi pembobotan: TF saja, IDF saja, TF x IDF, atau TF x IDF x *cosine normalization*
 6. Metode pembobotan *query*
 - a. Metode TF, yaitu *logarithmic*, *raw*, *binary*, atau *augmented*
 - b. Pilihan kalkulasi pembobotan: TF saja, IDF saja, TF x IDF, atau TF x IDF x *cosine normalization*

Setelah memastikan semua *input field* pada *toolbar* terisi, klik tombol *Retrieve* untuk mulai melakukan *document retrieval*.

Hasil *retrieval* akan ditampilkan pada halaman yang sama. Beberapa komponen hasil yang ditampilkan adalah sebagai berikut (referensi lihat Gambar C.3).

1. *Query*
 - a. *Query* asli (lihat Gambar C.3 bagian (a))
 - b. *Expanded query* di bawah tulisan *query* asli (lihat Gambar C.3 bagian (b))
 - c. Tombol untuk menampilkan bobot *query* asli dan *expanded query* (lihat Gambar C.3 bagian (c)). Bobot *query* akan ditampilkan pada *pop-up* seperti pada Gambar C.9.
2. Hasil *retrieval* yang ditampilkan ada dua jenis, yaitu hasil *retrieval* dari *query* asli dan hasil *retrieval* dari *expanded query*. Keduanya ditampilkan dalam bentuk *tab*.
 - a. Opsi untuk memilih hasil *retrieval* yang ingin ditampilkan: *query* asli atau *expanded query* (lihat Gambar C.3 bagian (d))
 - b. Skor MAP, khusus untuk *batch query* (lihat Gambar C.3 bagian (e))
 - c. Navigasi untuk melihat hasil *retrieval* untuk setiap *query* untuk *batch query* (lihat Gambar C.3 bagian (f))
 - d. Tombol untuk menampilkan *inverted file* (lihat Gambar C.3 bagian (g)). *Inverted file* seluruh dokumen ditampilkan pada *pop-up* seperti pada Gambar C.7. Untuk

menampilkan bobot kata pada dokumen tertentu, pilih dokumen yang diinginkan pada *dropdown*, seperti pada Gambar C.8.

- e. Daftar dokumen pada hasil *retrieval* (lihat Gambar C.3 bagian (h)), terdiri dari banyaknya dokumen yang diperoleh beserta skor AP (khusus *batch query*). Setiap dokumen terdiri dari ID, *rank*, *similarity*, judul, penulis, konten, serta bibliografi dokumen.

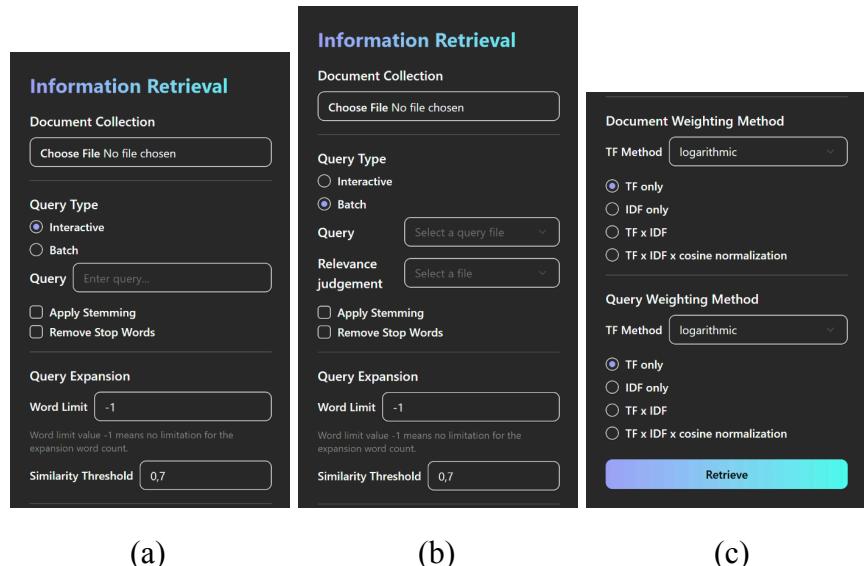
Lampiran C: Antarmuka Aplikasi

The screenshot shows a web-based application for Information Retrieval. On the left, there is a sidebar with various configuration options:

- Document Collection:** A file input field labeled "Choose File No file chosen".
- Query Type:** Radio buttons for "Interactive" (selected) and "Batch".
- Query:** An input field labeled "Enter query...".
- Checkboxes:** "Apply Stemming" and "Remove Stop Words".
- Query Expansion:**
 - Word Limit:** A dropdown menu set to "-1".Word limit value -1 means no limitation for the expansion word count.
 - Similarity Threshold:** A dropdown menu set to "0,7".
- Document Weighting Method:**
 - TF Method:** A dropdown menu set to "logarithmic".
 - Radio Buttons:** "TF only" (selected), "IDF only", "TF x IDF", and "TF x IDF x cosine normalization".

The main area features a large blue magnifying glass icon followed by the text "Information Retrieval".

Gambar C.1 Antarmuka web dasar



Gambar C.2 Toolbar: (a) tampilan *toolbar* untuk *query* interaktif / teks; (b) tampilan *toolbar* untuk *batch query*; (c) tampilan *toolbar* untuk metode pembobotan kata

Gambar C.3 Komponen antarmuka hasil *retrieval* (*batch query*)

Information Retrieval

Document Collection

Query Type

 Interactive
 Batch

Query information system

Apply Stemming
 Remove Stop Words

Query Expansion

Word Limit

Word limit value -1 means no limitation for the expansion word count.

Similarity Threshold

Document Weighting Method

TF Method

TF only
 IDF only
 TF x IDF
 TF x IDF x cosine normalization

Results for information system

Expanded query terms: inform system transfer dissemin interact storag tool interfac creat flow econom modern interact featur storag interfac mechan design computer-bas implement compon experiment

[View query details](#)

Original Query Result
Expanded Query Result
Inverted File

805 documents retrieved

Document ID: 689 • Rank #1 • Similarity: 0.12525
The GREMAS System, an Integral Part of the IDC System for Chemical Documentation
 by Rossler, SigridKolb, Arthur

The Genealogical Retrieval by Magnetic Tapes Storage (GREMAS) system and the potential it offers for searches are described. The input and retrieval procedures of the system are explained as well as the integration of the GREMAS system into the IDC system - i.e., machine generation of the GREMAS coding from topological input and of the superimposed bit code from the GREMAS coding.

Document ID: 872 • Rank #2 • Similarity: 0.12459
The Shared Cataloging System of the Ohio College Library Center
 by Kilgour, Frederick G.Long, Philip LLandgraf, Alan LWyckoff, John A.

Development and implementation of an off-line catalog card production system and an on-line shared cataloging system are described. In off-line production, average cost per card for 529,893 catalog cards in finished form and alphabetized for filing was 6.57¢. An account is given of system design and equipment selection for the on-line system. File organization and programs are described, and the on-line cataloging system is discussed. The system is easy to use, efficient, reliable, and cost beneficial.

Document ID: 1139 • Rank #3 • Similarity: 0.11647
The Language of an Polytechnical Automated Information Retrieval System
 by Korolev, E. I.

The principal design features are described of an information system using the natural language and a descriptor language: thesaurus organization, relevance criterion, indexing procedure, experimental estimates of the information language, and parametric information processing techniques.

Document ID: 1289 • Rank #4 • Similarity: 0.11507

Gambar C.4 Antarmuka hasil *retrieval* untuk *interactive query* (*query asli*)

Information Retrieval

Document Collection

Query Type

 Interactive
 Batch

Query information system

Apply Stemming
 Remove Stop Words

Query Expansion

Word Limit

Word limit value -1 means no limitation for the expansion word count.

Similarity Threshold

Document Weighting Method

TF Method

TF only
 IDF only
 TF x IDF
 TF x IDF x cosine normalization

Results for information system

Expanded query terms: inform system transfer dissemin interact storag tool interfac creat flow econom modern interact featur storag interfac mechan design computer-bas implement compon experiment

[View query details](#)

Original Query Result
Expanded Query Result
Inverted File

545 documents retrieved

Document ID: 1295 • Rank #1 • Similarity: 0.06531
Communication or Chaos?
 by Baker, D.B.

Effective transfer of scientific and technical information continues to be a pressing national problem.

Document ID: 630 • Rank #2 • Similarity: 0.05998
A Novel Philosophy for the Design of Information Storage and Retrieval Systems Appropriate for the '70's
 by Scheffler, Frederic L.

The philosophy of a systems approach to the design of information storage and retrieval systems is suggested in which the computer is recognized in its proper perspective as a powerful and effective alternative tool. This philosophy is in contrast to a prevalent philosophy of the '60's in which many information systems designer stoutly the computer as the answer to all information storage and retrieval situations. Important principles of information system design incorporated within the framework of the novel philosophy for the '70's...

Document ID: 1122 • Rank #3 • Similarity: 0.05581
Analysis of Some Regularities of the Flow of Engineering Information
 by Mitsevich, A. T.Solov'ev, N. K.

Some theoretical propositions are considered with respect to the flow of engineering information with the purpose of drawing practical conclusions for the editing of information announcement publications.

Document ID: 1301 • Rank #4 • Similarity: 0.04760
Citation Analysis as a Tool in Journal Evaluation
 by Garfield, E.

Gambar C.5 Antarmuka hasil *retrieval* untuk *interactive query* (*expanded query*)

Information Retrieval

Document Collection
Choose File No file chosen

Query Type
 Interactive
 Batch

Query query.text
Relevance judgement qrels.text

Apply Stemming
 Remove Stop Words

Query Expansion
Word Limit 4

Word limit value -1 means no limitation for the expansion word count.

Similarity Threshold 0,7

Document Weighting Method
TF Method logarithmic

TF only
 IDF only
 BM25

Results for What problems and concerns are there in making up descriptive titles?What difficulties are involved in automatically retrieving articles from approximate titles?What is the usual relevance of the content of articles to their titles?

Expanded query terms: problem concern make descrip titl ? difficulti involv automat retriev articl fromapproxim titl ? usual relev content articl titl ? shortcom rigor stabil coher

[View query details](#)

Original Query Result Expanded Query Result Inverted File

1460 documents retrieved • AP score: 0.20884

Document ID: 364 • Rank #1 • Similarity: 557.69501
Economic Analysis of the Public Libraries
by Newhouse, J.P.
This study addresses itself to several questions important to all public libraries. How should the library allocate its book budget? What kinds of books should it tend to buy? What types of households use the library? Why do some households not use the library? What is the cost of the various services provided by the library? What specific steps can the library take to improve its services? What are the library's options in choosing among the different circulation systems? For how long should the library allow books to be checked out? Ho...

Document ID: 589 • Rank #2 • Similarity: 492.96272
Are Titles of Chemical Papers Becoming More Informative?
by Tocatlian, Jacques J.
The efficiency of key-word-in-context (KWIC) permuted-title indexes and their numerous variations is highly dependent upon authors' choices of titles for their papers. Titles are important not only in commercial services, such as Chemical Titles, BASIC, Current Contents, and CA Condensates, but also in scanning primary journals, and in traditional library services, such as bibliographies. It is generally believed and often stated that titles of chemical papers are becoming more informative as authors become increasingly aware of the...

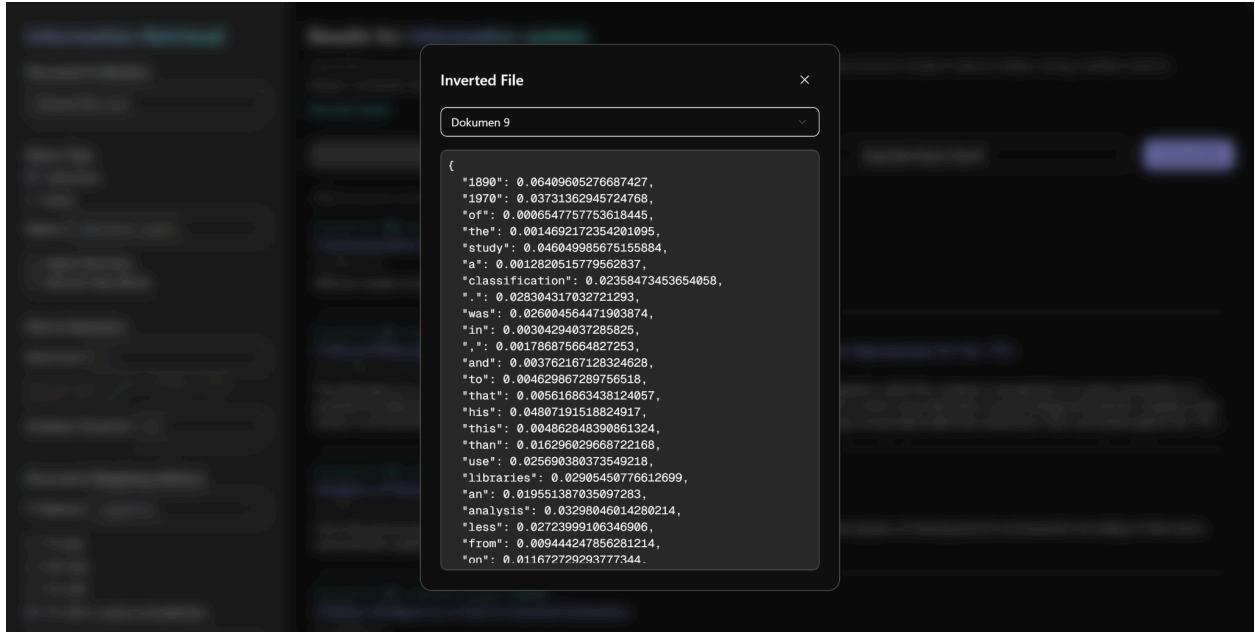
Document ID: 24 • Rank #3 • Similarity: 390.47027
Libraries and Technological Forces Affecting Them
by Cuadra, C.A.

Gambar C.6 Antarmuka hasil *retrieval* untuk *batch query*

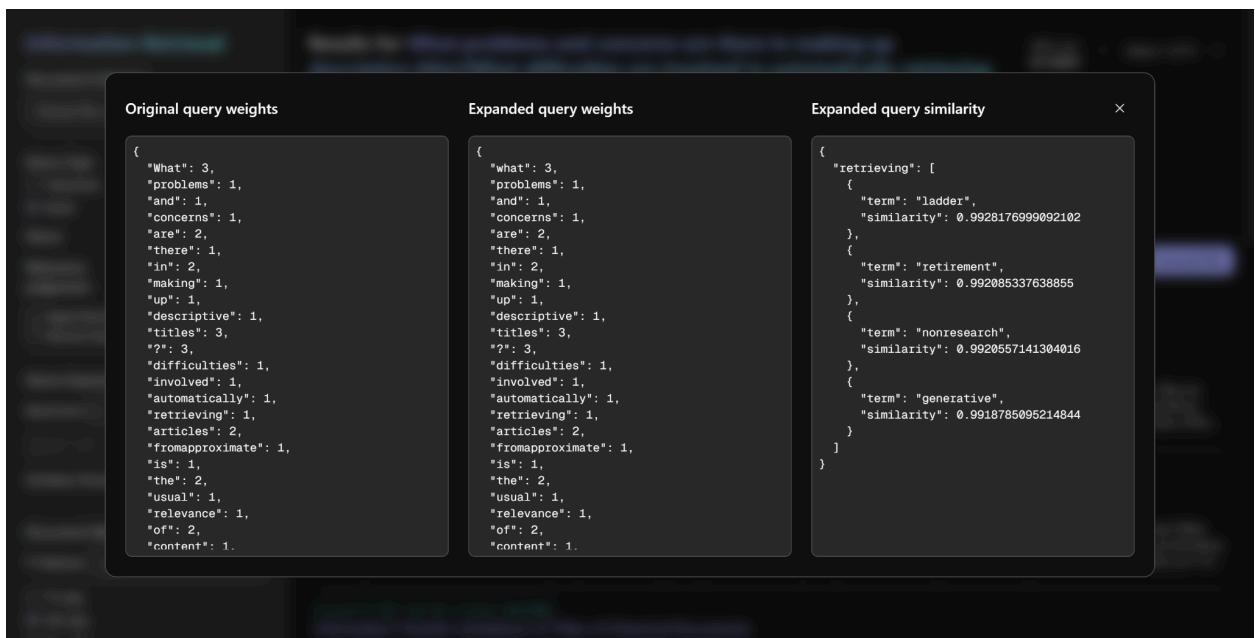
The screenshot shows a modal window titled "Inverted File" with a close button "X". Below the title is a dropdown menu set to "All Documents". The main area displays a JSON-like structure representing the inverted index:

```
{
  1: {
    "24": 0.0,
    "39": 0.0,
    "41": 0.0,
    "43": 0.0,
    "50": 0.0,
    "51": 0.0,
    "61": 0.0,
    "89": 0.0,
    "97": 0.0,
    "99": 0.0,
    "109": 0.0,
    "111": 0.0,
    "119": 0.0,
    "135": 0.0,
    "176": 0.0,
    "190": 0.0,
    "198": 0.0,
    "220": 0.0,
    "225": 0.0,
    "260": 0.0,
    "269": 0.0,
    "274": 0.0,
    "310": 0.0
  }
}
```

Gambar C.7 Antarmuka tampilan *inverted files* seluruh dokumen



Gambar C.8 Antarmuka tampilan pembobotan kata pada dokumen tertentu



Gambar C.9 Antarmuka tampilan detail bobot *query* asli dan *expanded query*

Lampiran D: Pembagian Tugas

No	Nama	NIM	Foto	Peran
1	Kevin John Wesley Hutabarat	13521042		<p><i>Back-end:</i></p> <ul style="list-style-type: none"> • Fungsi <i>parsing</i>: <code>parser_docs</code> • Fungsi <i>retrieval service</i>: <code>calculate_tf_idf</code>, <code>calculate_query_weight</code>, <code>retrieve_document_single_query</code>, <code>retrieve_document_by_id</code>, <code>retrieve_document_by_ids</code> • Fungsi <i>preprocessing</i>: <code>stem_word</code>, <code>stem_tokens</code>
2	Arleen Chrysantha Gunardi	13521059		<p><i>Front-end:</i></p> <ul style="list-style-type: none"> • Desain antarmuka dan implementasinya • Integrasi dengan API dari <i>back-end</i> • Implementasi <i>flow</i> aplikasi untuk semua kasus dan <i>end-to-end</i> <p><i>Back-end:</i></p> <ul style="list-style-type: none"> • Fungsi <code>expand_query</code>
3	Moh. Aghna Maysan Abyan	13521076		<p><i>Back-end:</i></p> <ul style="list-style-type: none"> • Fungsi <i>parsing</i>: <code>parser_qrels</code>, <code>parser_query</code>, <code>parser_docs</code> • File JSON: <code>parsing_qrels.json</code>, <code>parsing_query.json</code>, <code>parsing_docs.json</code>
4	Austin Gabriel Pardosi	13521084		<p><i>Back-end:</i></p> <ul style="list-style-type: none"> • Fungsi <i>query expansion & query expansion batch</i> • API Wrapping untuk semua service dan fungsi • Fungsi train & retraining word2vec • Fungsi <code>get_document_details</code> • Initialize & Dockerize

				Backend
5	Ryan Samuel Chandra	13521140	 A photograph of a young man with short dark hair and glasses, wearing a maroon vest over a white t-shirt. He is sitting at a table with a glass in front of him, smiling at the camera.	<p><i>Back-end:</i></p> <ul style="list-style-type: none"> • Fungsi <i>retrieval service</i>: <code>create_inverted_file</code>, <code>calculate_similarity</code>, <code>retrieve_document (single, batch)</code>, <code>get_weight_by_document_id</code>; • Fungsi <i>preprocessing</i>: tokenisasi, <i>remove stop word</i>, integrasi; • Fungsi evaluasi: <i>precision@K</i>, AP, MAP

Daftar Referensi

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., & Dean, J. (2013, Oktober 16). Distributed Representations of Words and Phrases and their Compositionality. *Advances in Neural*

Information Processing Systems, 1(1), 26. <https://arxiv.org/abs/1310.4546>

Vechtomova, O., & Wang, Y. (2006, July). A Study of the Effect of Term Proximity on Query Expansion. *Journal of Information Science, 32(4)*, 324-333.

<http://dx.doi.org/10.1177/0165551506065787>