REKOMENDASI FILM BERBASIS SINOPSIS MENGGUNAKAN *EMBEDDING* SBERT DAN *COSINE SIMILARITY*

SKRIPSI

Oleh :
ELA ILMATUL HIDAYAH
NIM. 220605110040

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2026

REKOMENDASI FILM BERBASIS SINOPSIS MENGGUNAKAN *EMBEDDING* SBERT DAN *COSINE SIMILARITY*

SKRIPSI

Diajukan kepada:
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk memenuhi Salah Satu Persyaratan dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)

Oleh:
ELA ILMATUL HIDAYAH
NIM. 220605110040

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2026

HALAMAN PERSETUJUAN

REKOMENDASI FILM BERBASIS SINOPSIS MENGGUNAKAN *EMBEDDING* SBERT DAN *COSINE SIMILARITY*

SKRIPSI

Oleh:
ELA ILMATUL HIDAYAH
NIM. 220605110040

Telah Diperiksa dan Disetujui untuk Diuji:
Tanggal: 19 Mei 2026

Pembimbing I,	Pembimbing II,
Khadijah Fahmi Hayati Holle, M.Kom	Ajib Hanani, M.T
NIP. 19900626 202203 2 002	NIP. 19840731 202321 1 013

Mengetahui,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang

Supriyono, M.Kom
NIP. 19841010 201903 1 012

HALAMAN PENGESAHAN

REKOMENDASI FILM BERBASIS SINOPSIS MENGGUNAKAN *EMBEDDING* SBERT DAN *COSINE SIMILARITY*

SKRIPSI

ELA ILMATUL HIDAYAH
NIM. 220605110040

Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
Tanggal: 15 Juni 2026

Susunan Dewan Penguji

Ketua Penguji	: Dr. Cahyo Crysdian, M.Cs NIP. 19740424 200901 1 008	( )
Anggota Penguji I	: Dr. M. Amin Hariyadi, M.T NIP. 19670118 200501 1 001	( )
Anggota Penguji II	: Khadijah Fahmi Hayati Holle, M.Kom NIP. 19900626 202203 2 002	( )
Anggota Penguji III	: Ajib Hanani, M.T NIP. 19840731 202321 1 013	( )

Mengetahui dan Mengesahkan,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang

Supriyono, M.Kom
NIP. 19841010 201903 1 012

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama	: Ela Ilmatul Hidayah
NIM	: 220605110040
Fakultas / Program Studi	: Sains dan Teknologi / Teknik Informatika
Judul Skripsi	: Rekomendasi Film Berbasis Sinopsis Menggunakan *Embedding* SBERT dan *Cosine Similarity*

Menyatakan dengan sebenarnya bahwa Skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambil alihan data, tulisan, atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan skripsi ini merupakan hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 15 Juni 2026
Yang membuat pernyataan,

Ela Ilmatul Hidayah
NIM.220605110040

MOTTO

... "Tuhan tidak akan memberi ujian diluar batas kemampuan manusia" ...

HALAMAN PERSEMBAHAN

Segala puji bagi Allah SWT atas rahmat, karunia, dan hidayah-Nya, sehingga penulis dapat menyelesaikan skripsi dengan baik. Dengan penuh rasa syukur, penulis persembahkan kepada:

1. Kedua orang tua tercinta, Bapak Kasdi Sujiyanto dan Ibu Siti Nur Hidayah, yang senantiasa memberikan dukungan, baik moral maupun material, serta mencurahkan kasih sayang, doa, dan pengorbanan yang tiada henti sejak penulis lahir hingga saat ini. Terima kasih atas segala cinta, ketulusan, dan perjuangan yang telah diberikan sehingga penulis dapat menempuh dan menyelesaikan pendidikan ini.
2. Kakak penulis, Khafid Khoirul Anam dan Devi Noriana, yang selalu mendampingi proses perjalanan penulis, membantu meringankan beban dan tanggung jawab orang tua, serta memberikan semangat, dukungan, dan kasih sayang kepada penulis. Tidak lupa kepada keponakan tercinta, Arumi Noriana Shezan, yang senantiasa menghadirkan senyum dan kebahagiaan di tengah perjalanan perkuliahan penulis.
3. Keluarga Bani Miseri khususnya Mas Wafa, Mbak Catrine, dan Dek Ria, yang selalu hadir memberikan dukungan, perhatian, dan pendampingan kepada penulis selama menjalani kehidupan di tanah perantauan. Terima kasih telah menjadi keluarga sekaligus sahabat yang menemani penulis dalam berbagai keadaan.
4. Dosen pembimbing penulis, Ibu Khadijah Fahmi Hayati Holle, M.Kom., yang dengan penuh kesabaran telah membimbing, mengarahkan,

memberikan motivasi, serta senantiasa mengingatkan penulis selama proses penyusunan skripsi hingga selesai.

5. Teman-teman penulis, Infinity 2022, yang selalu memberikan bantuan, dukungan, dan kebersamaan selama masa perkuliahan. Secara khusus kepada teman PKL, Bela dan Firna; teman satu bimbingan, Wasik dan Ridiey; serta teman satu kamar asrama yang telah menjadi bagian penting dalam perjalanan akademik penulis.
6. Rekan-rekan *Public Relation* HIMATIF "ENCODER" dan Paguyuban Duta Fakultas Sains dan Teknologi Tahun 2023, yang telah menjadi keluarga kedua bagi penulis di tanah perantauan serta memberikan banyak pengalaman, pembelajaran, dan kenangan berharga.
7. Diri penulis sendiri, Ela Ilmatul Hidayah, yang telah berjuang dengan penuh ketekunan, kesabaran, dan semangat dalam menempuh pendidikan, menghadapi berbagai tantangan, serta terus bertahan hingga mampu menyelesaikan skripsi ini dengan baik.
8. Seluruh pihak yang telah membantu dan memberikan dukungan, baik secara langsung maupun tidak langsung, sejak awal masa perkuliahan hingga penyelesaian skripsi ini, yang tidak dapat disebutkan satu per satu.

KATA PENGANTAR

Assalamu'alaikum Wr. Wb.

Alhamdulillahirabbil'alamin, segala puji dan rasa syukur penulis panjatkan ke hadirat Allah Subhanahu Wa Ta'ala atas rahmat, karunia, dan hidayah-Nya, sehingga penulis dapat menyelesaikan skripsi yang berjudul "Rekomendasi Film Berbasis Sinopsis Menggunakan *Embedding* SBERT dan *Cosine Similarity*". Sholawat serta salam tetap tercurahkan kepada junjungan kita Nabi Muhammad Shallallahu Alaihi Wasallam, yang menjadi suri tauladan dan semoga kita mendapat syafaat di akhir kelak, Aamiin.

Dalam penyusunan skripsi ini, penulis menyadari bahwa banyak pihak telah memberikan doa dan dukungan. Oleh karena itu, penulis mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Prof. Dr. Hj. Ifi Nur Diana, M.Si., CAHRM., CRMP., selaku Rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Dr. Agus Mulyono, M.Kes., selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Supriyono, M.Kom., selaku Ketua Program Studi Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang.
4. Khadijah Fahmi Hayati Holle, M.Kom selaku dosen pembimbing I yang selalu memberikan *support* selama penyusunan skripsi.
5. Ajib Hanani, M.T selaku dosen pembimbing II yang selalu memberikan bimbingan kepenulisan selama penyusunan skripsi.

6. Dr. Cahyo Crysdian, M.Cs selaku Ketua Dosen Penguji yang telah memberi saran dan masukan sehingga penulis dapat menyelesaikan skripsi dengan baik.
7. Dr. M. Amin Hariyadi, M.T selaku Penguji I yang telah memberi saran dan masukan kepenulisan sehingga penulis dapat menyelesaikan skripsi dengan baik.
8. Nia Faricha, S.Si., selaku Admin Program Studi Teknik Informatika, yang dengan penuh kesabaran telah membantu penulis dalam berbagai urusan administrasi.

Demikian penulisan skripsi ini disusun dengan harapan penulisan ini dapat berkontribusi dalam pengembangan ilmu pengetahuan di masa depan.

Wassalamu'alaikum Wr, Wb.

Malang, 15 Juni 2026

Penulis

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

ABSTRAK

Hidayah, Ilmatul, Ela. 2026. **Rekomendasi Film Berbasis Sinopsis Menggunakan *Embedding* SBERT dan *Cosine Similarity*.** Skripsi. Program Studi Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Khadijah Fahmi Hayati Holle, M.Kom. (II) Ajib Hanani, M.T.

.

Kata kunci: *Cosine Similarity,* SBERT*, Semantic Similarity,* Sistem Rekomendasi Film

Rekomendasi film merupakan salah satu solusi untuk membantu pengguna menemukan film yang sesuai dengan preferensinya di tengah banyaknya pilihan film yang tersedia. Namun, pendekatan rekomendasi berbasis kecocokan kata sering kali belum mampu memahami makna semantik dari kebutuhan pengguna sehingga menghasilkan rekomendasi yang kurang relevan. Penelitian ini bertujuan membangun sistem rekomendasi film berbasis sinopsis menggunakan *Sentence*-BERT (SBERT) dan *Cosine Similarity* untuk menghasilkan rekomendasi berdasarkan kesamaan semantik antara *query* pengguna dan sinopsis film. Sistem dikembangkan dengan model all-MiniLM-L6-v2 untuk menghasilkan *embedding* kalimat dan metode *Cosine Similarity* untuk menghitung tingkat kemiripan antara *query* pengguna dan sinopsis film. Pengujian dilakukan menggunakan 40 *query* yang terdiri dari 20 *query* pendek dan 20 *query* panjang pada delapan skenario yang mengkombinasikan penggunaan *preprocessing*, panjang *query*, dan jumlah rekomendasi Top-5 serta Top-10. Validasi relevansi dilakukan oleh satu validator dan dievaluasi menggunakan metrik *Precision*, *Mean Average Precision* (MAP), *Mean Reciprocal Rank* (MRR), dan *Normalized Discounted Cumulative Gain* (nDCG). Hasil pengujian menunjukkan bahwa sistem mampu menghasilkan rekomendasi yang relevan pada seluruh skenario pengujian yang digunakan. Skenario terbaik diperoleh pada pengujian tanpa *preprocessing*, menggunakan *query* pendek dan Top-5, dengan nilai *Precision* sebesar 0.990, MAP sebesar 1, MRR sebesar 1, dan nDCG sebesar 1. Secara keseluruhan, perpaduan SBERT dan *Cosine Similarity* efektif digunakan untuk memahami kemiripan makna antara *query* pengguna dan sinopsis film sehingga mampu menghasilkan rekomendasi yang relevan dan memiliki kualitas ranking yang baik.

ABSTRACT

Hidayah, Ilmatul Ela. 2026. **Movie Recommendation Based on Synopsis Using SBERT *Embedding* and *Cosine Similarity*.** Thesis. Informatics Engineering Study Program, Faculty of Science and Technology, Maulana Malik Ibrahim State Islamic University of Malang. Advisor: (I) Khadijah Fahmi Hayati Holle, M.Kom. (II) Ajib Hanani, M.T.

.

Keywords: Cosine Similarity, Movie Recommendation System, SBERT, Semantic Similarity

Movie recommendation systems provide a solution to help users discover films that match their preferences amid the vast number of available movie choices. However, recommendation approaches based solely on keyword matching often fail to capture the semantic meaning of user needs, resulting in less relevant recommendations. This study aims to develop a synopsis-based movie recommendation system using Sentence-BERT (SBERT) and Cosine Similarity to generate recommendations based on the semantic similarity between user queries and movie synopses. The system employs the all-MiniLM-L6-v2 model to generate sentence embeddings and utilizes Cosine Similarity to measure the similarity between user queries and movie synopses. The evaluation was conducted using 40 queries consisting of 20 short queries and 20 long queries across eight experimental scenarios combining preprocessing techniques, query lengths, and the number of recommendations Top-5 and Top-10. Relevance validation was performed by validator and evaluated using Precision, Mean Average Precision (MAP), Mean Reciprocal Rank (MRR), and Normalized Discounted Cumulative Gain (nDCG). The results indicate that the proposed system is capable of generating relevant recommendations across all testing scenarios used in this study. The best performance was achieved in the scenario without preprocessing, using short queries and Top-5 recommendations, obtaining a Precision score of 0.990, MAP of 1., MRR of 1., and nDCG of 1. Overall, the combination of SBERT and Cosine Similarity effectively captures semantic similarity between user queries and movie synopses, resulting in relevant, accurate, and well-ranked movie recommendations.

مستخلص البحث

هداية، علمة إيلا. ٢٠٢٦م. نظام توصية الأفلام المعتمد على الملخصات باستخدام إس-بيرت (SBERT) وتشابه جيب التمام. بحث جامعي. قسم هندسة المعلوماتية، كلية العلوم والتكنولوجيا، جامعة مولانا مالك إبراهيم الإسلامية الحكومية بمالانج. المشرفان: (الأول) خديجة فهمي حياة الله هولي الماجستير. (الثاني) عجيب حناني، الماجستير.

الكلمات المفتاحية: نظام توصية الأفلام، إس-بيرت، تشابه جيب التمام، التشابه الدلالي

يُعَدُّ نظامُ توصيةِ الأفلام أحدَ الحلول التي تساعد المستخدمين على العثور على الأفلام التي تتوافق مع تفضيلا تهم في ظل العدد الكبير من الأفلام المتاحة. ومع ذلك، فإن أساليب التوصية المعتمدة على مطابقة الكلمات المفتاحية غالبًا ما تعجز عن فهم المعنى الدلالي لاحتياجات المستخدم، مما يؤدي إلى تقديم توصيات أقل ملاءمة. يهدف هذا البحث إلى بناء نظام لتوصية الأفلام اعتمادًا على الملخصات باستخدام نموذج إس-بيرت وتشابه جيب التمام لإنتاج (SBERT) توصيات قائمة على التشابه الدلالي بين استفسارات المستخدم وملخصات الأفلام. تم تطوير النظام باستخدام نموذج all-MiniLM-L6-v2 لتوليد التمثيلات الدلالية للنصوص، كما استُخدم أسلوب تشابه جيب التمام لقياس درجة التشابه بين استفسارات المستخدم وملخصات الأفلام. وأُجريت عملية الاختبار باستخدام أربعين استفسارًا، تتكون من عشرين استفسارًا قصيرًا وعشرين استفسارًا طويلًا ، ضمن ثمانية سيناريوهات تجمع بين استخدام المعالجة المسبقة، وطول الاستفسار وعدد التوصيات (أفضل خمسة وأفضل عشرة). وتم التحقق من مدى ملاءمة النتائج من خلال المحكِّمين، كما تم تقييم أداء النظام باستخدام مقاييس الدقة (Precision)، ومتوسط الدقة العام (MAP)، ومتوسط مقلوب الرتبة (MRR)، ومقياس الكسب التراكمي المخصوم المطبَّع (nDCG). وأظهرت نتائج الاختبار أن النظام قادر على تقديم توصيات ذات صلة عالية في جميع سيناريوهات الاختبار. وقد تحقق أفضل أداء في سيناريو عدم استخدام المعالجة المسبقة مع الاستفسارات القصيرة وأفضل خمسة توصيات، حيث بلغت قيمة الدقة 0.990، ومتوسط الدقة العام 1، ومتوسط مقلوب الرتبة 1 ومقياس الكسب التراكمي المخصوم المطبَّع 1. وبصورة عامة، أثبت الجمع بين نموذج إس-بيرت وتشابه جيب التمام فعاليته في فهم التشابه الدلالي بين استفسارات المستخدم وملخصات الأفلام، مما أسهم في إنتاج توصيات ذات صلة ودقة عالية وجودة ترتيب جيدة.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Era digital yang serba cepat ini, kemajuan teknologi informasi memberikan dampak yang signifikan terhadap berbagai sektor. Salah satu sektor yang menonjol yakni industri hiburan. Industri *streaming* video global yang mengalami pertumbuhan pesat mencerminkan perubahan besar dalam kebiasaan menonton masyarakat. Menurut laporan *Fortune Business Insights*, pasar video *streaming* global diperkirakan akan tumbuh dari USD 674,25 miliar pada tahun 2024 menjadi USD 2.660.88 miliar pada tahun 2032, dengan tingkat pertumbuhan tahunan gabungan (CAGR) sebesar 18,5% selama periode proyeksi (*Video Streaming Market Size, Share & Trends | Growth [2032]*, n.d.). Perubahan ini mencerminkan pergeseran signifikan dari media tradisional seperti televisi kabel dan siaran ke *platform streaming digital*. Kini konsumen semakin memilih untuk menikmati hiburan film melalui layanan *streaming* seperti Netflix, Prime Video dan *platform* lainnya yang menawarkan fleksibilitas dalam waktu dan lokasi menonton.

Teknologi yang semakin berkembang dan preferensi konsumen yang mengarah ke digitalisasi, peralihan ini semakin jelas terasa di masyarakat. Pada tahun 2025, layanan *streaming* seperti Netflix dan Prime Video untuk pertama kalinya melampaui saluran televisi tradisional dalam hal jumlah penonton. Menurut laporan Nielsen, perusahaan riset pasar dan media global terkemuka menunjukkan bahwa *streaming* menyumbang 44,8% dari total waktu menonton televisi, mengungguli televisi kabel dan siaran masing-masing yang menyumbang 24,1%

dan 20,1% (“Streaming Reaches Historic TV Milestone, Eclipses Combined Broadcast and Cable Viewing For First Time,” n.d.). Fenomena ini menandai peralihan penting dalam cara masyarakat mengakses hiburan dan konsumsi media yang sebelumnya didominasi oleh saluran televisi tradisional. Meskipun industri video *streaming* berkembang pesat, sistem rekomendasi yang digunakan oleh *platform streaming* masih menghadapi berbagai kendala dalam menghasilkan rekomendasi yang akurat dan sesuai dengan preferensi pengguna. Beberapa permasalahan yang umum ditemukan antara lain *cold start problem* dan data *sparsity* yang dapat menurunkan kualitas rekomendasi yang diberikan.

Cold start problem merupakan salah satu dari sekian permasalahan dalam sistem rekomendasi, yaitu kondisi ketika sistem kesulitan memberikan rekomendasi yang akurat kepada pengguna baru atau item baru karena belum tersedia data interaksi yang memadai. Masalah ini terjadi ketika pengguna baru tidak disarankan dengan tepat karena kurangnya informasi preferensi yang lebih rinci yang menghambat sistem rekomendasi secara efektif (Yuan & Hernandez, 2023). Masalah lainnya adalah *data sparsity*, dimana kurangnya data interaksi pengguna menyebabkan kesulitan dalam mengidentifikasi film yang cocok untuk ditonton. Penelitian oleh Gunathilaka pada tahun 2025 menyoroti tantangan ini, dimana kurangnya data peringkat yang tersedia menghambat kemampuan sistem rekomendasi untuk memahami minat pengguna dan hubungan antar item secara akurat (Gunathilaka *et al*., 2025). Kedua masalah ini berdampak pada kualitas rekomendasi yang telah diberikan, seringkali menghasilkan rekomendasi yang tidak relevan atau kurang sesuai dengan preferensi pengguna yang pada gilirannya

mempengaruhi pengalaman menonton pengguna. Oleh karena itu, pendekatan *content-based filtering* menjadi alternatif yang lebih stabil karena hanya bergantung pada karakteristik item, seperti sinopsis film, sehingga tidak terlalu terpengaruh oleh keterbatasan data interaksi pengguna.

Sistem rekomendasi film yang ada saat ini banyak didasarkan pada teknik *Content-Based Filtering* dan *collaborative filtering* yang menganalisis interaksi dan preferensi pengguna (Bhowmick *et al*., 2021). Namun, sistem rekomendasi berbasis konten seringkali terbatas pada kesamaan fitur eksplisit seperti *genre* dan *rating* yang tidak selalu menangkap semantik kontekstual dari sinopsis film yang menjadi salah satu faktor penting dalam pemilihan film (Grezes *et al*., 2022). Oleh karena itu, diperlukan pendekatan yang lebih canggih untuk menangkap dan memahami konteks cerita dalam sinopsis film. Pada penelitian yang dilakukan Javaji dkk mengusulkan model rekomendasi hibrida yang mengatasi keterbatasan dengan menggabungkan kelebihan *collaborative filtering* dan *Content-Based Filtering* (Javaji & Sarode, 2023). Pemfilteran yang digunakan adalah *collaborative filtering* untuk preferensi pengguna dan *content based filtering* untuk atribut film untuk mendapatkan rekomendasi yang akurat dan beragam. Namun, pengembangan model hibrida memerlukan strategi integrasi dan pembobotan yang tepat agar setiap metode dapat bekerja secara optimal (Ram *et al*., 2023). Kesalahan dalam menentukan kombinasi metode dapat mempengaruhi relevansi dan akurasi rekomendasi yang dihasilkan.

Pemahaman semantik terhadap sinopsis film menjadi sangat penting dalam meningkatkan relevansi dan personalisasi sistem rekomendasi. Setiap film

memiliki cerita, tema, dan pesan tertentu yang dapat mencerminkan preferensi dan minat penonton (Rahmatabadi *et al*., n.d.), untuk itu, *Natural Language Processing* (NLP) hadir sebagai solusi yang memungkinkan sistem untuk memahami teks sinopsis yang lebih baik. Dengan menggunakan model *pre-trained* seperti *Sentence*-BERT (SBERT), sistem rekomendasi dapat memahami konteks cerita dalam sinopsis secara lebih mendalam dibandingkan pendekatan berbasis kata (Patil et al., 2024). Kemampuan tersebut memungkinkan sistem menghasilkan rekomendasi yang lebih relevan dengan makna *query* yang diberikan pengguna.

Salah satu permasalahan dalam pengembangan sistem rekomendasi berbasis sinopsis adalah menentukan tingkat kesamaan makna antara *query* yang dimasukkan pengguna dan sinopsis film secara tepat (Mustafa et al., 2024). Untuk mengatasi hal tersebut, digunakan metode *Cosine Similarity* sebagai teknik pengukuran kemiripan antar *embedding* yang dihasilkan oleh SBERT. Metode ini bekerja dengan menghitung kedekatan vektor representasi semantik, sehingga film yang memiliki makna paling relevan dengan *query* pengguna dapat ditempatkan pada urutan rekomendasi teratas.

Dalam Al-Qur'an, konsep pengetahuan dan pemahaman seringkali dikaitkan dengan pencarian yang mendalam terhadap makna dan hikmah di balik segala sesuatu. Dalam Surah *Al-'Alaq* surat ke 96 ayat 1-5, Allah SWT berfirman yang berbunyi:

اِقْرَأْ بِاسْمِ رَبِّكَ الَّذِيْ خَلَقَۚ ﴿١﴾ خَلَقَ الْاِنْسَانَ مِنْ عَلَقٍۚ ﴿٢﴾ اِقْرَأْ وَرَبُّكَ الْاَكْرَمُۙ ﴿٣﴾ الَّذِيْ عَلَّمَ بِالْقَلَمِۙ ﴿٤﴾
عَلَّمَ الْاِنْسَانَ مَا لَمْ يَعْلَمْۗ ﴿٥﴾

"1) Bacalah dengan (menyebut) nama Tuhanmu yang menciptakan! 2) Dia menciptakan manusia dari segumpal darah, 3) Bacalah! dan Tuhanmu yang Maha

Mulia, 4) yang mengajar dengan pena, 5) Dia mengajarkan manusia apa yang tidak diketahuinya" (QS. Al-'Alaq/96: 1-5).

Dalam penelitian ini, pemilihan Surah *Al-'Alaq* ayat 1-5 sebagai landasan teori cukup berkaitan dengan pentingnya pengetahuan dan proses pembelajaran. Berdasarkan Tafsir Ibnu Katsir, kelima ayat tersebut menyampaikan pesan yang sangat relevan dengan tujuan penelitian yang berfokus pada pemahaman dan analisis sinopsis film menggunakan teknologi canggih. Ayat pertama dan ketiga menekankan pentingnya membaca sebagai sarana memperoleh pengetahuan. Nilai tersebut sejalan dengan penelitian ini yang memanfaatkan teknologi pemrosesan bahasa alami untuk memahami informasi yang terkandung dalam sinopsis film. Ayat keempat menegaskan pentingnya sarana dalam penyebaran ilmu pengetahuan. Dalam konteks penelitian ini, teknologi komputasi berperan sebagai sarana untuk mengolah informasi sehingga dapat menghasilkan rekomendasi yang bermanfaat bagi pengguna. Surah ini mengajarkan bahwa ilmu tidak hanya diperoleh melalui proses membaca, tetapi juga melalui pemahaman dan analisis. Nilai tersebut sejalan dengan penelitian ini yang memanfaatkan SBERT dan NLP untuk memahami makna sinopsis film secara lebih kontekstual dan semantik. (Al-Sheikh, 1994)

Berdasarkan permasalahan dan kajian yang telah dipaparkan, diajukan penelitian berjudul "Rekomendasi Film Berbasis Sinopsis Menggunakan *Embedding* SBERT dan *Cosine Similarity*". Penelitian ini bertujuan untuk menganalisis performa sistem rekomendasi film berbasis sinopsis menggunakan SBERT dan *Cosine Similarity* dalam menghasilkan rekomendasi yang relevan berdasarkan kesamaan semantik antara *query* pengguna dan sinopsis film.

Pemilihan SBERT didasarkan pada kemampuannya menghasilkan representasi kalimat yang mempertimbangkan konteks dan makna semantik secara menyeluruh. Kemampuan tersebut memungkinkan sistem memahami hubungan antar kata dan kalimat dalam sinopsis film sehingga rekomendasi yang dihasilkan tidak hanya berdasarkan kesamaan kata, tetapi juga kesamaan makna. Sementara itu, *Cosine Similarity* digunakan untuk mengukur tingkat kemiripan antar representasi teks yang dihasilkan oleh SBERT sehingga sistem dapat menemukan film yang memiliki makna paling dekat dengan *query* pengguna secara lebih efektif.

Penelitian ini menggunakan 4 metrik evaluasi, yaitu *Precision*, *Mean Average Precision* (MAP), *Mean Reciprocal Rank* (MRR), dan *Normalized Discounted Cumulative Gain* (nDCG). *Precision* digunakan untuk mengukur tingkat ketepatan sistem dalam memberikan rekomendasi pada posisi K teratas dengan melihat proporsi item yang relevan terhadap seluruh item yang direkomendasikan. MAP digunakan untuk mengevaluasi kualitas sistem secara keseluruhan dengan mempertimbangkan urutan kemunculan item relevan dalam daftar rekomendasi. MRR digunakan untuk mengukur seberapa cepat sistem mampu menampilkan item relevan pertama pada hasil rekomendasi, sehingga mencerminkan efisiensi sistem dalam memenuhi kebutuhan pengguna. Sementara itu, nDCG digunakan untuk menilai kualitas peringkat dengan memberikan bobot lebih tinggi pada item relevan yang muncul di posisi teratas.

Pemilihan metrik evaluasi tersebut didasarkan pada kesesuaiannya dengan karakteristik sistem rekomendasi berbasis *ranking*, di mana urutan hasil rekomendasi menjadi aspek yang sangat penting. Dengan menggunakan kombinasi

metrik ini, evaluasi sistem diharapkan mampu memberikan gambaran yang komprehensif mengenai kualitas relevansi serta kemampuan sistem dalam menyajikan rekomendasi film yang sesuai dengan preferensi pengguna secara lebih akurat dan terurut.

1.2 Pernyataan Masalah

1. Bagaimana performa rekomendasi film berbasis sinopsis menggunakan *Embedding* SBERT dan *Cosine Similarity*?
2. Bagaimana pengaruh penggunaan preprocessing terhadap performa sistem rekomendasi film berbasis sinopsis?
3. Bagaimana pengaruh panjang *query* terhadap kualitas rekomendasi yang dihasilkan?
4. Bagaimana pengaruh jumlah rekomendasi Top-5 dan Top-10 terhadap kualitas hasil rekomendasi

1.3 Batasan Masalah

a. Dataset yang digunakan tmdb_5000_movies.csv dari Ibtesam Ahmed berbahasa inggris yang dapat diakses secara umum di *platform* Kaggle.
b. Metrik evaluasi yang digunakan pada penelitian ini terbatas pada *Precision*, *Mean Average Precision* (MAP), *Mean Reciprocal Rank* (MRR), dan *Normalized Discounted Cumulative Gain* (nDCG).
c. Penelitian ini dilakukan hingga tahap evaluasi performa terbaik yang dihasilkan oleh sistem rekomendasi.

1.4 Tujuan Penelitian

1. Menganalisis performa sistem rekomendasi film berbasis sinopsis menggunakan *Embedding* SBERT dan *Cosine Similarity* berdasarkan metrik *Precision, Mean Average Precision* (MAP), *Mean Reciprocal Rank* (MRR), dan *Normalized Discounted Cumulative Gain* (nDCG).
2. Menganalisis pengaruh penggunaan *preprocessing* terhadap performa sistem rekomendasi film berbasis sinopsis.
3. Menganalisis pengaruh panjang *query* terhadap kualitas rekomendasi yang dihasilkan.
4. Menganalisis pengaruh jumlah rekomendasi Top-5 dan Top-10 terhadap kualitas hasil rekomendasi.

1.5 Manfaat Penelitian

Diharapkan hasil dari penelitian ini mampu memberikan manfaat baik dari segi teoritis maupun praktis, sebagai berikut:

a. Penelitian ini diharapkan dapat memperkaya literatur mengenai sistem rekomendasi film berbasis sinopsis menggunakan SBERT dan *Cosine Similarity.*
b. Penelitian ini diharapkan dapat digunakan oleh pengembang dalam menyajikan sistem rekomendasi yang lebih memudahkan pengguna dalam mencari rekomendasi film.

BAB II

STUDI PUSTAKA

2.1 Rekomendasi Film Berbasis Sinopsis

Rekomendasi film berbasis sinopsis merupakan salah satu pendekatan dalam sistem rekomendasi yang memanfaatkan informasi deskriptif berupa ringkasan cerita film untuk menemukan film lain yang memiliki kemiripan konten (Harahap & Rachman, 2025). Pendekatan ini termasuk ke dalam metode *Content-Based Filtering*, yaitu teknik rekomendasi yang menghasilkan saran berdasarkan karakteristik atau konten dari suatu item (Dwi Ayu Nouvalina & Kusuma Hati, 2024). Dalam konteks film, sinopsis menjadi atribut yang penting karena mampu menggambarkan alur cerita, tema, konflik, latar, dan karakter yang terdapat dalam film secara lebih rinci dibandingkan atribut lain seperti genre atau rating.

Pemanfaatan sinopsis dalam sistem rekomendasi dilakukan dengan mengubah teks deskriptif menjadi representasi yang dapat diproses oleh komputer. Penelitian yang dilakukan oleh Harahap dan Rachman (2025) membangun sistem rekomendasi film menggunakan metode *Content-Based Filtering* dengan memanfaatkan kemiripan deskripsi cerita film. Pada penelitian tersebut, sinopsis film terlebih dahulu melalui tahapan *preprocessing*, kemudian direpresentasikan menggunakan TF-IDF dan dihitung tingkat kemiripannya menggunakan *Cosine Similarity*. Hasil penelitian menunjukkan bahwa kemiripan sinopsis dapat digunakan untuk menemukan film dengan karakteristik cerita yang serupa tanpa memerlukan data preferensi pengguna secara langsung.

Pendekatan serupa juga diterapkan oleh Fuady, Mawardi, dan Sutrisno (2023) yang mengembangkan sistem rekomendasi film berdasarkan genre dan sinopsis. Dalam penelitian tersebut, sinopsis digunakan sebagai sumber informasi utama untuk membangun representasi konten film. Setelah dilakukan pembobotan kata menggunakan TF-IDF, kemiripan antar film dihitung menggunakan *Cosine Similarity* sehingga sistem dapat memberikan rekomendasi film yang memiliki kesamaan konteks cerita dengan film yang dipilih pengguna. Hasil penelitian menunjukkan bahwa kombinasi informasi sinopsis dan teknik pengukuran kemiripan teks mampu meningkatkan relevansi rekomendasi yang dihasilkan.

Pentingnya informasi tekstual dalam sistem rekomendasi film juga ditunjukkan oleh penelitian Madhavi et al. (2021) yang mengembangkan *Movie Recommendation System using Free Text*. Penelitian tersebut memanfaatkan deskripsi film sebagai sumber informasi utama dalam proses rekomendasi. Hasil penelitian menunjukkan bahwa penggunaan teks bebas mampu memberikan rekomendasi yang lebih relevan dibandingkan pendekatan yang tidak memanfaatkan informasi tekstual secara mendalam (Yeole Madhavi B. et al., 2021). Temuan ini menunjukkan bahwa isi cerita yang terkandung dalam sinopsis memiliki kontribusi penting dalam merepresentasikan karakteristik film sehingga dapat digunakan untuk menemukan film-film yang memiliki kesamaan konten.

Selain itu, Bhavsar et al. (2024) mengembangkan *Content-Based Movie Recommendation System* dengan memanfaatkan informasi tekstual dari sinopsis film pada dataset TMDB. Penelitian tersebut menunjukkan bahwa pendekatan berbasis konten yang memanfaatkan analisis teks mampu menghasilkan

rekomendasi yang lebih akurat dibandingkan metode dasar yang hanya menggunakan atribut sederhana. Hasil penelitian tersebut memperkuat bahwa sinopsis merupakan sumber informasi yang kaya dan mampu menggambarkan hubungan antar film secara lebih baik dibandingkan metadata yang bersifat umum (Apura M. Bhavsar et al., 2024).

Perkembangan teknologi pemrosesan bahasa alami NLP juga mendorong peningkatan kualitas sistem rekomendasi berbasis sinopsis. Patil et al. (2024) membandingkan beberapa metode representasi teks, yaitu Bag-of-Words, Word2Vec, TF-IDF, dan BERT pada sistem rekomendasi film. Hasil penelitian menunjukkan bahwa model berbasis transformer seperti BERT mampu menghasilkan performa yang lebih baik dibandingkan metode representasi teks konvensional karena dapat memahami hubungan kontekstual dan makna semantik antar kata dalam deskripsi film. Temuan tersebut menunjukkan bahwa kualitas representasi teks memiliki pengaruh yang signifikan terhadap kualitas rekomendasi yang dihasilkan, terutama pada sistem yang mengandalkan sinopsis sebagai sumber informasi utama.

Secara umum, proses rekomendasi film berbasis sinopsis dilakukan melalui beberapa tahapan, yaitu pengumpulan data sinopsis film, *preprocessing* teks, pembentukan representasi teks, perhitungan tingkat kemiripan antar film, dan pengurutan hasil berdasarkan nilai kemiripan tertinggi. Film yang memiliki tingkat kemiripan paling tinggi dianggap memiliki kesamaan cerita dengan film acuan sehingga direkomendasikan kepada pengguna. Pendekatan ini memungkinkan

sistem memberikan rekomendasi berdasarkan kesamaan konten film tanpa memerlukan data histori pengguna.

Tabel 2. 1 Penelitian mengenai Sistem Rekomendasi Film Berbasis Sinopsis

Peneliti	**Metode dan Dataset**	**Jenis Evaluasi**	**Keterbatasan**	**Perbedaan dengan Penelitian ini**
Sistem Rekomendasi Film Berdasarkan Kemiripan Deskripsi Cerita Menggunakan Metode *Content-Based Filtering* (Harahap & Rachman 2025)	- Content-Based Filtering, TF-IDF, *Cosine Similarity* - Dataset sinopsis film	Berbasis nilai similarity	Belum memanfaatkan representasi semantik berbasis transformer	Penelitian ini menggunakan SBERT untuk menghasilkan *embedding* semantik kalimat
Implementasi Metode *Content-Based Filtering* untuk Rekomendasi Film Berdasarkan Genre dan Sinopsis (Fuady, Mawardi, & Sutrisno 2023)	- Content-Based Filtering, TF-IDF, *Cosine Similarity* - Dataset film informasi genre dan sinopsis	Evaluasi berbasis nilai *Cosine Similarity* dan hasil rekomendasi	Representasi teks masih berbasis frekuensi kata	Menggunakan *embedding* SBERT yang mampu memahami konteks kalimat
Movie Recommendation System using Free Text (Madhavi *et al*., 2021)	- TF-IDF, Word2Vec - Dataset tmdb_5000_movies.csv dari *platform* Kaggle	*Accuracy*	Belum memanfaatkan model transformer	Menggunakan SBERT dengan representasi semantik yang lebih kontekstual
Content-Based Movie Recommendation System (Bhavsar *et al*., 2024)	- TF-IDF, *Cosine Similarity* - Dataset tmdb_5000_movies.csv dari *platform* Kaggle	*Accuracy*	Ketergantungan pada kata kunci eksplisit	Menggunakan *embedding* kalimat berbasis transformer
Enhancing Movie Recommendation Systems with BERT: A Deep Learning Approach (Patil *et al*., 2024)	- BERT. TF-IDF, Word2Vec, BoW - Dataset tmdb_5000_movies.csv dan tmdb_5000_credits.csv dari *platform* Kaggle	*Precision, Recall, F1-Score* dan MAP	Fokus pada BERT dengan kebutuhan komputasi lebih besar	Menggunakan all-MiniLM-L6-v2 yang lebih ringan dan efisien

Berdasarkan berbagai penelitian terdahulu pada Tabel 2.1 dapat disimpulkan bahwa sinopsis merupakan sumber informasi yang efektif dalam sistem rekomendasi film. Sebagian besar penelitian menggunakan metode representasi teks seperti TF-IDF, Word2Vec, dan BERT untuk menangkap informasi dari sinopsis film. Meskipun metode tersebut mampu menghasilkan rekomendasi yang relevan, TF-IDF dan Word2Vec masih memiliki keterbatasan dalam memahami hubungan semantik secara kontekstual, sedangkan BERT memiliki kebutuhan komputasi yang relatif lebih besar. Selain itu, sebagian besar penelitian terdahulu masih menggunakan representasi teks berbasis TF-IDF dan Word2Vec yang memiliki keterbatasan dalam memahami hubungan semantik secara kontekstual. Penelitian yang menggunakan model transformer umumnya berfokus pada BERT yang memiliki kebutuhan komputasi relatif tinggi. Oleh karena itu, penelitian ini memanfaatkan SBERT dengan *backbone* all-MiniLM-L6-v2 yang lebih ringan dan dioptimalkan untuk tugas *semantic similarity*.

Berdasarkan celah penelitian tersebut, penelitian ini mengembangkan sistem rekomendasi film berbasis sinopsis menggunakan model *Sentence*-BERT (SBERT) dengan backbone all-MiniLM-L6-v2 dan *Cosine Similarity*. Sinopsis film digunakan sebagai sumber informasi utama untuk menghasilkan representasi semantik dalam bentuk *embedding* berdimensi 384 yang kemudian dibandingkan menggunakan *Cosine Similarity* guna menghasilkan rekomendasi film yang relevan.

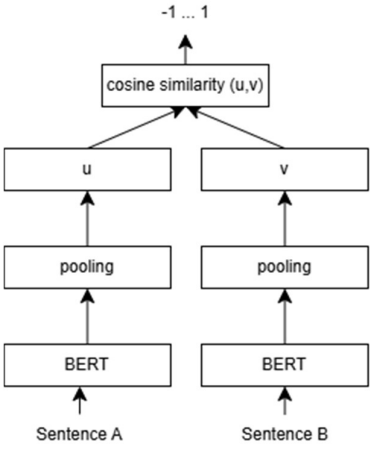
2.2 *Embedding* SBERT

Sentence-BERT (SBERT) merupakan modifikasi dari *Bidirectional Encoder Representations from Transformers* (BERT) yang dirancang untuk menghasilkan representasi kalimat (*sentence embedding*) yang bermakna secara semantik dan dapat dibandingkan secara langsung menggunakan metode pengukuran kemiripan seperti *Cosine Similarity* (Reimers & Gurevych, 2019). Berbeda dengan BERT standar yang memerlukan pemrosesan dua kalimat secara bersamaan untuk mengukur kemiripan, SBERT menghasilkan *embedding* kalimat yang dapat dihitung kemiripannya secara lebih efisien sehingga cocok digunakan pada tugas *semantic similarity* dan sistem rekomendasi.

Pemanfaatan SBERT pada sistem rekomendasi menunjukkan hasil yang baik dalam memahami hubungan semantik antar teks. Penelitian yang dilakukan oleh Lim dan Lee (2026) menyatakan bahwa penggunaan SBERT pada sistem rekomendasi mampu meningkatkan relevansi rekomendasi yang diberikan kepada pengguna karena model dapat memahami konteks kebutuhan pengguna secara lebih baik (Lim & Lee, 2026). Temuan-temuan tersebut menunjukkan bahwa SBERT merupakan pendekatan yang sesuai untuk digunakan dalam sistem rekomendasi yang memanfaatkan informasi tekstual sebagai sumber utama data.

Kemampuan SBERT dalam memahami makna kalimat didukung oleh penggunaan *embedding* kontekstual. *embedding* kontekstual merupakan representasi numerik yang mempertimbangkan konteks kemunculan suatu kata dalam kalimat. Berbeda dengan metode *embedding* tradisional yang menghasilkan satu representasi tetap untuk setiap kata, *embedding* kontekstual mampu

menghasilkan representasi yang berbeda untuk kata yang sama apabila digunakan pada konteks yang berbeda (Liu et al., 2020). Kemampuan tersebut memungkinkan model memahami hubungan semantik antar kata dan kalimat secara lebih baik sehingga dapat meningkatkan performa beberapa tugas *Natural Language Processing* (NLP).



Gambar 2. 1 Arsitektur SBERT

Secara arsitektur pada Gambar 2.1, SBERT dibangun di atas model Transformer yang terdiri atas beberapa komponen utama, yaitu *tokenizer*, *Transformer Encoder*, *pooling layer*, dan *Sentence embedding*. Proses dimulai ketika teks berupa *query* pengguna atau sinopsis film dimasukkan ke dalam *tokenizer* untuk dipecah menjadi token-token yang dapat dipahami oleh model. Token hasil tokenisasi kemudian diproses oleh *Transformer Encoder* yang memanfaatkan mekanisme *self-attention* untuk mempelajari hubungan antar token dalam suatu kalimat. Penggunaan *self-attention* mengolah setiap token untuk memperoleh informasi dari token lain sehingga model mampu memahami konteks kalimat secara dua arah (Vaswani et al., 2017). *Output* dari *Transformer Encoder*

berupa representasi vektor untuk setiap token yang selanjutnya diproses menggunakan pooling layer. Pada model *Sentence* Transformer, *pooling* digunakan untuk mengubah kumpulan token *embedding* menjadi satu vektor tunggal yang merepresentasikan keseluruhan makna kalimat. Hasil akhir dari proses tersebut adalah *embedding* kalimat yang dapat digunakan untuk mengukur kemiripan semantik antar teks

Dalam penelitian ini digunakan model *Sentence* Transformer all-MiniLM-L6-v2 sebagai backbone encoder untuk menghasilkan *embedding* kalimat. Model all-MiniLM-L6-v2 dibangun menggunakan arsitektur MiniLM yang dikembangkan melalui pendekatan deep *self-attention distillation*, yaitu proses transfer pengetahuan dari model transformer berukuran besar ke model yang lebih kecil tanpa mengurangi kemampuan pemahaman bahasa secara signifikan (Wang et al., 2020). Model ini terdiri atas enam layer *Transformer Encoder* dengan *hidden size* sebesar 384 dimensi dan setiap layer memanfaatkan mekanisme *multi-head self-attention* untuk mempelajari hubungan antar token secara kontekstual dan *feed-forward network* untuk memperkaya representasi fitur, sehingga mampu menghasilkan representasi semantik yang efisien namun tetap akurat. Dibandingkan model transformer yang lebih besar, all-MiniLM-L6-v2 memiliki jumlah parameter yang lebih sedikit dan waktu inferensi yang lebih cepat sehingga cocok digunakan untuk memproses ribuan sinopsis film dalam sistem rekomendasi. Ortakci (2024) menjelaskan bahwa model ini memiliki performa yang kompetitif pada berbagai tugas *semantic similarity* dengan kebutuhan komputasi yang relatif rendah (Ortakci, 2024).

Berdasarkan karakteristik tersebut, SBERT dengan model all-MiniLM-L6-v2 dipilih dalam penelitian ini untuk menghasilkan representasi semantik dari *query* pengguna dan sinopsis film. Setiap teks akan diubah ke dalam bentuk *embedding* kalimat berdimensi 384 yang merepresentasikan makna keseluruhan kalimat. Representasi tersebut kemudian digunakan dalam proses perhitungan *Cosine Similarity* untuk menentukan tingkat kemiripan antara *query* pengguna dan sinopsis film sehingga dapat menghasilkan rekomendasi film yang relevan sesuai dengan konteks kebutuhan pengguna.

2.3 *Cosine Similarity*

Cosine Similarity merupakan salah satu metode yang banyak digunakan untuk mengukur tingkat kemiripan antar dua vektor dalam ruang multidimensi. Pada bidang *Natural Language Processing* (NLP), metode ini sering digunakan untuk membandingkan representasi teks yang telah diubah ke dalam bentuk vektor sehingga dapat diketahui tingkat kesamaan makna antar dokumen, kalimat, maupun kata. Berbeda dengan metode berbasis jarak seperti *Euclidean Distance* yang memperhatikan besarnya nilai setiap dimensi, *Cosine Similarity* mengukur sudut yang terbentuk antara dua vektor. Semakin kecil sudut yang terbentuk, semakin tinggi tingkat kemiripan antara kedua vektor tersebut (Zhou et al., 2022).

Penelitian yang dilakukan oleh Harahap dan Rachman (2025) memanfaatkan *Cosine Similarity* untuk menghitung kemiripan antar sinopsis film yang telah direpresentasikan menggunakan TF-IDF. Hasil penelitian menunjukkan bahwa metode tersebut mampu menghasilkan rekomendasi film berdasarkan kesamaan isi cerita. Fuady, Mawardi, dan Sutrisno (2023) dalam penelitiannya juga

menggunakan *Cosine Similarity* untuk membandingkan representasi teks sinopsis dan memperoleh rekomendasi film yang relevan dengan preferensi pengguna. Temuan tersebut menunjukkan bahwa *Cosine Similarity* merupakan metode yang efektif untuk digunakan dalam sistem rekomendasi berbasis teks karena mampu mengukur hubungan antar dokumen secara sederhana namun akurat.

Secara matematis, *Cosine Similarity* menghitung nilai *cosinus* dari sudut yang terbentuk antara dua vektor. Pada penelitian ini, vektor pertama merepresentasikan *embedding query* pengguna, sedangkan vektor kedua merepresentasikan *embedding* sinopsis film. Perhitungan dilakukan dengan membandingkan arah kedua vektor dalam ruang *embedding* berdimensi 384. Nilai *similarity* diperoleh dari hasil perkalian titik (*dot product*) kedua vektor yang kemudian dinormalisasi menggunakan panjang (*magnitude*) masing-masing vektor Semakin kecil sudut yang terbentuk antara kedua vektor, semakin tinggi nilai *Cosine Similarity* yang dihasilkan. Berikut persamaan untuk perhitungan *Cosine Similarity* (Nurma Romihim Fadlilah, 2025):

$$Cosine\ Similarity = \frac{A \cdot B}{|A||B|} = \frac{\sum_{i=1}^{n} A_i \ x \ B_i}{\sqrt{\sum_{i=1}^{n}(A_I)^2} \ x \ \sqrt{\sum_{i=1}^{n}(B_I)^2}} \qquad (2.1)$$

Keterangan:
A: vektor *embedding query* pengguna
B: vektor *embedding* sinopsis film
A · B: *dot product* antara vektor A dan vektor B
|A|: Panjang vektor A
|B|: Panjang vektor B
Ai: Nilai pada dimensi ke-i dari vektor *embedding query*
Bi: Nilai pada dimensi ke-i dari vektor *embedding* sinopsis
N: Jumlah dimensi *embedding* yang digunakan

Nilai *Cosine Similarity* berada pada rentang -1 hingga 1. Nilai 1 menunjukkan bahwa kedua vektor memiliki arah yang sama sehingga memiliki

kemiripan yang sangat tinggi. Nilai 0 menunjukkan bahwa kedua vektor tidak memiliki hubungan atau kemiripan, sedangkan nilai -1 menunjukkan bahwa kedua vektor memiliki arah yang berlawanan. Namun, pada representasi *embedding* yang dihasilkan oleh model SBERT, nilai *similarity* umumnya berada pada rentang 0 hingga 1 karena representasi vektor cenderung berada pada ruang semantik yang sama. Semakin mendekati nilai 1, semakin tinggi tingkat kemiripan makna antara *query* dan sinopsis film yang dibandingkan.

Dalam sistem rekomendasi, proses perhitungan *Cosine Similarity* dilakukan terhadap seluruh sinopsis film yang terdapat dalam dataset. Setiap *query* pengguna terlebih dahulu diubah menjadi *embedding* menggunakan model SBERT. Selanjutnya, *embedding query* dibandingkan dengan *embedding* seluruh sinopsis film sehingga diperoleh nilai *similarity* untuk masing-masing film. Nilai *similarity* yang dihasilkan kemudian diurutkan secara menurun (*descending sorting*) untuk membentuk peringkat rekomendasi. Film yang memiliki nilai *similarity* tertinggi dianggap memiliki tingkat kesamaan semantik paling tinggi terhadap *query* pengguna sehingga ditempatkan pada posisi teratas hasil rekomendasi.

Penelitian Singh et al. (2024) dan Munkholm et al. (2024) menunjukkan bahwa *Cosine Similarity* efektif digunakan dalam sistem rekomendasi berbasis konten karena mampu membandingkan hubungan semantik antar representasi teks dengan baik (Munkholm et al., 2024; Singh et al., 2024). Oleh karena itu, metode ini digunakan dalam penelitian untuk menghitung tingkat kemiripan antara *embedding query* pengguna dan *embedding* sinopsis film sebagai dasar dalam proses perangkingan rekomendasi.

BAB III

DESAIN DAN IMPLEMENTASI

3.1 Dataset

Data yang digunakan merupakan data sekunder yang dikembangkan oleh Ibtesam Ahmed, yaitu tmdb_5000_movies.csv yang terdiri dari 4.803 data film berbahasa inggris. Dataset ini dapat diakses secara umum melalui *website* resmi Kaggle. Dataset TMDB dipilih karena memiliki jumlah data yang cukup besar dan menyediakan informasi sinopsis yang beragam yang memungkinkan model untuk mempelajari hubungan semantik antar film secara lebih baik. Dataset tersebut memuat berbagai informasi terkait film, seluruh fitur yang tersedia pada dataset ditunjukkan pada Tabel 3.1.

Tabel 3. 1 Fitur Dataset

Fitur	Keterangan
budget	Anggaran produksi film
genres	Daftar genre film
homepage	URL situs resmi film (jika tersedia)
id	ID unik untuk setiap film
Keywords	Kata kunci yang berhubungan dengan tema film
original_language	Bahasa asli film
original_title	Judul asli film
overview	Sinopsis film, yang menjelaskan cerita utama film
popularity	Skor popularitas berdasarkan data TMDB
production_companies	Daftar perusahaan yang memproduksi film
production_countries	Negara tempat produksi film dilakukan.
release_date	Tanggal perilisan film
revenue	Pendapatan film
runtime	Durasi film dalam menit
spoken_languages	Bahasa yang digunakan dalam film
status	Status film (misalnya: Released, Post Production)
tagline	Slogan atau frasa promosi film
title	Judul resmi film
vote_average	Rata-rata rating yang diberikan oleh pengguna TMDB
vote_count	Jumlah suara yang diberikan oleh pengguna

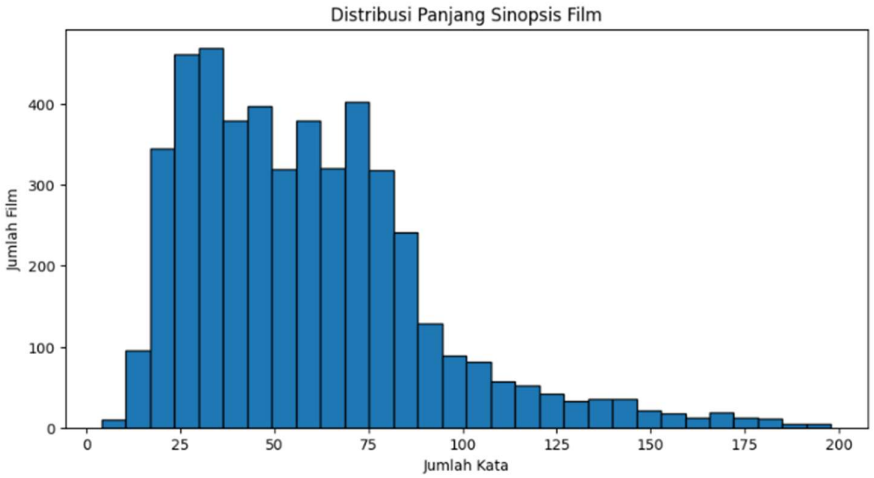
Penelitian ini menggunakan fitur *title*, *overview*, dan *genres*. Fitur *overview* digunakan sebagai sumber utama pembentukan *embedding*, *title* digunakan untuk menampilkan hasil rekomendasi, sedangkan *genres* digunakan sebagai informasi pendukung dalam proses validasi.. Pemilihan fitur pada Tabel 3.1 didasarkan pada tujuan penelitian yang berfokus pada sistem rekomendasi film berbasis sinopsis. Fitur *overview* berisi ringkasan cerita film yang mampu menggambarkan tema, alur cerita, karakter, konflik, serta konteks yang terdapat dalam film secara lebih lengkap dibandingkan fitur lainnya.

Pada Tabel 3.2. Tabel tersebut menampilkan beberapa sinopsis film yang berasal dari fitur *overview*. Setiap sinopsis memiliki panjang dan tingkat detail informasi yang berbeda-beda, sehingga memberikan variasi data yang dapat digunakan untuk menguji kemampuan model dalam memahami makna semantik dari teks yang diproses.

Tabel 3.2 Contoh Fitur *Overview*.

Overview
In the 22nd century, a paraplegic Marine is dispatched to the moon Pandora on a unique mission, but becomes torn between following orders and protecting an alien civilization.
As Harry begins his sixth year at Hogwarts, he discovers an old book marked as 'Property of the Half-Blood Prince', and begins to learn more about Lord Voldemort's dark past.
One year after their incredible adventures in the Lion, the Witch and the Wardrobe, Peter, Edmund, Lucy and Susan Pevensie return to Narnia to aid a young prince whose life has been threatened by the evil King Miraz. Now, with the help of a colorful cast of new characters, including Trufflehunter the badger and Nikabrik the dwarf, the Pevensie clan embarks on an incredible quest to ensure that Narnia is returned to its rightful heir.
When Tony Stark's world is torn apart by a formidable terrorist called the Mandarin, he starts an odyssey of rebuilding and retribution.

Untuk mendukung proses pemahaman terhadap karakteristik data yang digunakan, dilakukan tahap *Exploratory Data Analysis* (EDA) dengan memvisualisasikan distribusi panjang sinopsis pada fitur *overview*.



Gambar 3. 1 EDA Panjang Sinopsis

Berdasarkan hasil EDA, dataset yang digunakan terdiri atas 4.800 film setelah melalui proses penghapusan *missing value* dan data duplikat. Panjang sinopsis minimum adalah4 kata, panjang maksimum 198 kata, dengan rata-rata 52 kata dan median 48 kata. Nilai tersebut menunjukkan bahwa sebagian besar sinopsis memiliki panjang yang cukup untuk merepresentasikan isi cerita film secara deskriptif.

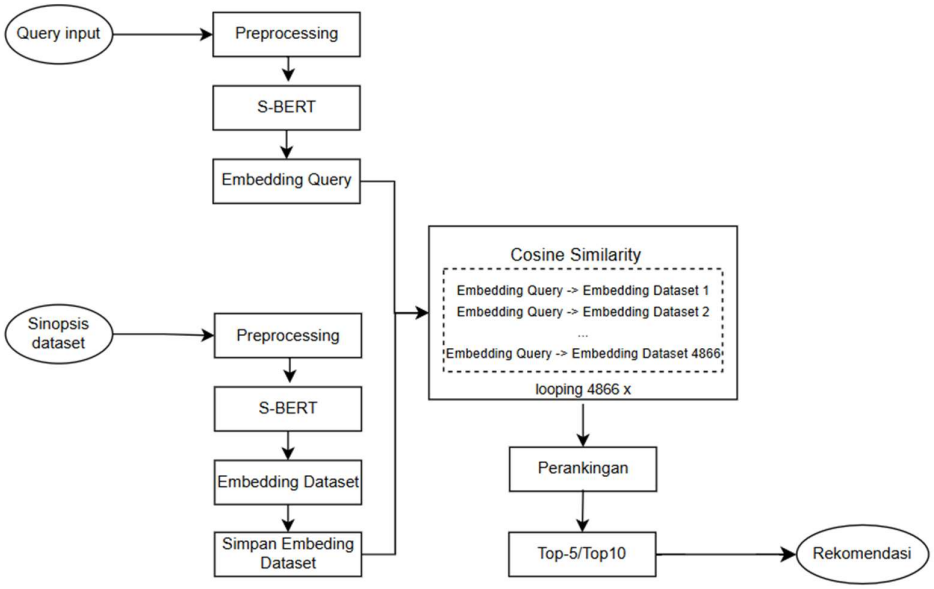
Visualisasi distribusi panjang sinopsis pada Gambar 3.1 menunjukkan bahwa panjang sinopsis bervariasi antar film, namun sebagian besar data terkonsentrasi pada rentang tertentu yang mendekati nilai rata-ratanya. Variasi panjang sinopsis ini memberikan informasi yang beragam bagi model dalam mempelajari hubungan semantik antar film.

Selain itu, dilakukan pemeriksaan terhadap panjang teks untuk memastikan kesesuaiannya dengan kemampuan pemrosesan model *all-MiniLM-L6-v2*. Hasil analisis menunjukkan bahwa tidak terdapat sinopsis yang memiliki panjang lebih

dari 200 kata (0 data). Dengan demikian, seluruh sinopsis dapat diproses oleh model tanpa memerlukan pemotongan teks (*truncation*) yang berpotensi menghilangkan informasi penting dari sinopsis film.

3.2 Desain Sistem

Pada penelitian ini, Gambar 3.2 menyajikan tahapan proses sistem rekomendasi film berbasis sinopsis yang meliputi:



Gambar 3. 2 Desain Sistem

Pembersihan data (*data cleaning*) merupakan tahapan awal yang dilakukan sebelum proses pembentukan embedding dan perhitungan kemiripan dilakukan. Tahap ini bertujuan untuk meningkatkan kualitas data yang digunakan dalam penelitian sehingga proses pengolahan data dapat berjalan dengan lebih efektif dan menghasilkan rekomendasi yang lebih relevan. Pada penelitian ini, pembersihan data dilakukan melalui penghapusan nilai kosong (*missing value*), penghapusan

data duplikat (*duplicate removal*), dan pemilihan fitur yang digunakan dalam sistem rekomendasi.

a. Penghapusan Nilai Kosong (*Missing Value Removal*)

   Nilai kosong (*missing value*) merupakan kondisi ketika suatu atribut tidak memiliki informasi yang lengkap. Keberadaan nilai kosong dapat mempengaruhi proses pembentukan representasi teks dan menyebabkan informasi film menjadi tidak lengkap. Oleh karena itu, data yang memiliki nilai kosong pada atribut yang digunakan dalam penelitian dihapus dari dataset sehingga hanya data yang memiliki informasi lengkap yang digunakan pada tahap selanjutnya.

b. Penghapusan Data Duplikat (*Duplicate Removal*)

   Setelah proses penghapusan nilai kosong, dilakukan pemeriksaan terhadap data duplikat. Data duplikat merupakan data film yang muncul lebih dari satu kali dalam dataset. Keberadaan data duplikat dapat menyebabkan bias pada proses pembentukan *embedding* dan perhitungan kemiripan karena suatu film dapat direpresentasikan lebih dari satu kali. Dataset awal terdiri dari 4.803 film. Setelah dilakukan penghapusan *missing value* dan *duplicate removal*, jumlah data yang digunakan dalam penelitian menjadi 4.799 film.

c. Pemilihan Fitur Dataset

   Dataset TMDB 5000 Movies memiliki berbagai atribut yang mendeskripsikan informasi film, seperti judul film, sinopsis, genre, tanggal rilis, popularitas, dan atribut lainnya. Pemilihan fitur dilakukan untuk menyesuaikan kebutuhan sistem rekomendasi film berbasis sinopsis yang dibangun. Fitur yang digunakan

dalam penelitian ini terdiri atas *title*, *overview*, dan *genres*. Fitur *title* digunakan untuk menampilkan judul film pada hasil rekomendasi, sedangkan *overview* digunakan sebagai sumber informasi utama dalam pembentukan *embedding* menggunakan model SBERT. Selain itu, fitur *genres* digunakan sebagai informasi pendukung dalam proses validasi hasil rekomendasi.

3.3 Skenario *Preprocessing*

Pada penelitian ini, pengujian dilakukan menggunakan dua skenario *preprocessing*, yaitu tanpa *preprocessing* dan dengan *preprocessing*. Tujuan dari pengujian ini adalah untuk mengetahui pengaruh tahapan *preprocessing* terhadap kualitas rekomendasi yang dihasilkan oleh sistem rekomendasi film berbasis SBERT. Perbedaan kedua skenario terletak pada ada atau tidaknya proses pembersihan teks yang diterapkan sebelum pembentukan *embedding*.

3.3.1 Tanpa *Preprocessing*

Pada skenario tanpa *preprocessing*, *query* pengguna digunakan secara langsung sebagai masukan model SBERT tanpa melalui proses pembersihan teks. *Query* yang dimasukkan pengguna tetap mempertahankan bentuk aslinya, termasuk penggunaan huruf kapital, tanda baca, maupun *stopword* yang terdapat di dalam kalimat. Selanjutnya *query* tersebut diproses menggunakan model SBERT untuk menghasilkan representasi semantik dalam bentuk *embedding*.

Skenario ini digunakan untuk mengetahui kemampuan model SBERT dalam memahami makna semantik teks tanpa bantuan tahapan *preprocessing*. Hasil dari skenario ini kemudian dibandingkan dengan skenario yang menggunakan

preprocessing untuk mengetahui pengaruh *preprocessing* terhadap kualitas rekomendasi yang dihasilkan.

3.3.2 Menggunakan *Preprocessing*

Pada skenario dengan *preprocessing*, *query* pengguna terlebih dahulu melalui beberapa tahapan pembersihan teks sebelum diproses menggunakan model SBERT. Tahapan *preprocessing* bertujuan untuk mengurangi *noise* pada data teks dan menghasilkan representasi yang lebih terstruktur. Adapun tahapan *preprocessing* yang digunakan pada penelitian ini meliputi *case folding*, *tokenization*, *lemmatization*, *symbol and punctuation removal*, serta *stopword removal*.

a. Tokenisasi dan *Case folding*

Tahap ini dilakukan dengan mengubah seluruh huruf pada data menjadi huruf kecil untuk memastikan keseragaman penulisan. Tujuan dari tahap ini adalah untuk menghindari perbedaan penulisan kata yang seharusnya memiliki makna yang sama, seperti perbedaan antara huruf kapital dan huruf kecil, sehingga dapat meningkatkan konsistensi dan keakuratan dalam proses analisis. Pada Tabel 3.3 ditampilkan contoh penggunaan *case folding*.

Tabel 3. 3 Contoh proses tokenisasi dan *case folding*

Sebelum *case folding*	**Tokenisasi sederhana**	**Sesudah *case folding***
" OMG!! In 1995, a rookie FBI Agent joins an Elite team to stop a bank robbery in Las Vegas—only to realize the mastermind is his own bro. U won't believe the twist!!! :)) #Heist "	*['OMG, '!', '!' 'in', '1995', ',', 'a', 'rookie', 'FBI', 'Agent', 'joins', 'an', 'Elite', 'team', 'to', 'stop', 'a', 'bank', 'robbery', 'in', 'Las', 'Vegas', '—', 'only', 'to', 'realize', 'the', 'mastermind', 'is', 'his', 'own', 'bro', '.', 'U', 'won't', 'believe', 'the', 'twist', '!', '!', '!', ':', ')', ')', '#', 'Heist']*	*" omg!! in 1995, a rookie fbi agent joins an elite team to stop a bank robbery in las vegas—only to realize the mastermind is his own bro. u won't believe the twist!!! :)) #heist "*

b. *Lemmatization*

Lemmatization dilakukan untuk mengubah setiap kata menjadi bentuk dasarnya (*lemma*). Dalam penelitian ini, digunakan *WordNetLemmatizer* dari pustaka NLTK untuk melakukan *lemmatization* pada sinopsis film. Fokusnya pada perapian ejaan tidak baku, pelurusan *slang* atau singkatan, dan penyetaraan penulisan nama umum ("*Agent/agent*") sesuai kaidah kalimat. Pada Tabel 3.4 terdapat contoh penggunaan *lemmatization* sebagai berikut*:*

Tabel 3. 4 Contoh *lemmatization*

Sebelum *lemmatization*	**Sesudah *lemmatization***
" omg!! in 1995, a rookie fbi agent joins an elite team to stop a bank robbery in las vegas—only to realize the mastermind is his own bro. u won't believe the twist!!! :)) #heist "	*['omg', '!', '!' 'in', '1995', ',', 'a', 'rookie', 'fbi', 'agent', 'joins', 'an', 'elite', 'team', 'to', 'stop', 'a', 'bank', 'robbery', 'in', 'las', 'vegas', '—', 'only', 'to', 'realize', 'the', 'mastermind', 'is', 'his', 'own', 'bro', '.', 'u', 'will', 'not', 'believe', 'the', 'twist', '!', '!', '!', ':', ')', ')', '#', 'heist']*

c. *Symbol and Punctuation Removal*

Symbol and Punctuation Removal dilakukan untuk menghapus simbol dan tanda baca yang tidak memiliki banyak kontribusi terhadap makna teks sehingga teks menjadi lebih bersih dan seragam sebelum diproses lebih lanjut. Penanganan simbol dan tanda baca dilakukan secara minimalis dan bermakna: karakter yang jelas tergolong *noise*—seperti *hashtag*, *mention*, emoji, serta deretan tanda baca berulang dihapus atau dinormalkan (misalnya "!!!" → "!"), sementara tanda yang mempengaruhi struktur dan makna dipertahankan: koma, titik, titik dua pada judul, tanda hubung/em–dash sebagai pemisah klausa, serta apostrof. Angka dan tahun rilis tetap dijaga karena kerap menjadi penanda identitas film.

Tabel 3. 5 Contoh proses *Symbol* dan *Punctuation Removal*

Sebelum *Symbol and Punctuation Removal*	**Sesudah *Symbol and Punctuation Removal***
['omg', '!', '!' 'in', '1995', ',', 'a', 'rookie', 'fbi', 'agent', 'joins', 'an', 'elite', 'team', 'to', 'stop', 'a', 'bank', 'robbery', 'in', 'las', 'vegas', '—', 'only', 'to', 'realize', 'the', 'mastermind', 'is', 'his', 'own', 'bro', '.', 'u', 'will', 'not', 'believe', 'the', 'twist', '!', '!', '!', ':', ')', ')', '#', 'heist']	*['omg', '!' 'in', '1995', ',', 'a', 'rookie', 'fbi', 'agent', 'joins', 'an', 'elite', 'team', 'to', 'stop', 'a', 'bank', 'robbery', 'in', 'las', 'vegas', '—', 'only', 'to', 'realize', 'the', 'mastermind', 'is', 'his', 'own', 'bro', '.', 'u', 'will', 'not', 'believe', 'the', 'twist', '!']*

d. Stopword Removal

Stopwords adalah kata-kata umum dalam bahasa inggris (seperti *a, an, the, in, to*) yang tidak membawa makna penting dalam konteks analisis teks. Menghapus *stopwords* membantu sistem untuk fokus pada kata-kata yang memiliki bobot semantik lebih tinggi. Pada penelitian ini, kata *seperti "not", "no",* dan *"without"* dipertahankan karena memiliki peran penting dalam menjaga makna kalimat.

Tabel 3. 6 Contoh proses *Stopword removal*

Sebelum *Stopword removal*	**Sesudah *Stopword removal***
['omg', '!' 'in', '1995', ',', 'a', 'rookie', 'fbi', 'agent', 'joins', 'an', 'elite', 'team', 'to', 'stop', 'a', 'bank', 'robbery', 'in', 'las', 'vegas', '—', 'only', 'to', 'realize', 'the', 'mastermind', 'is', 'his', 'own', 'bro', '.', 'u', 'will', 'not', 'believe', 'the', 'twist', '!']	*['omg', '1995', 'rookie', 'fbi', 'agent', 'joins', 'elite', 'team', 'stop', 'bank', 'robbery', 'las', 'vegas', 'realize', 'mastermind', 'bro', 'you', 'will', 'not', 'believe', 'twist']*

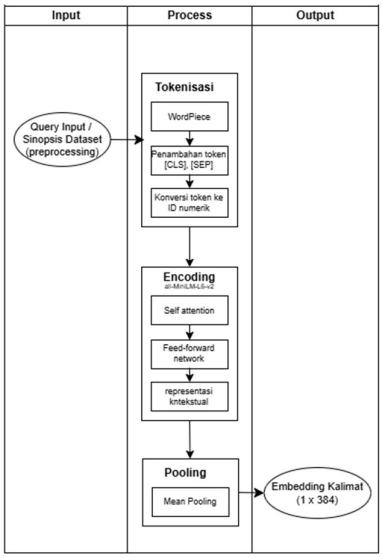
Preprocessing digunakan sebagai salah satu skenario pembanding untuk mengetahui pengaruh pembersihan teks terhadap performa rekomendasi. Pada model berbasis Transformer seperti SBERT, *preprocessing* berat tidak selalu meningkatkan kualitas representasi karena model memanfaatkan konteks kalimat secara utuh. Oleh karena itu, hasil skenario *preprocessing* dibandingkan dengan skenario tanpa *preprocessing* untuk melihat perlakuan mana yang menghasilkan rekomendasi lebih baik

3.4 Pembentukan SBERT *Embedding*

Setelah melalui tahap pembersihan data dan *preprocessing*, seluruh sinopsis film pada dataset diubah menjadi representasi numerik menggunakan model SBERT dengan *backbone all-MiniLM-L6-v2*. Proses ini bertujuan untuk menghasilkan representasi semantik dari setiap sinopsis film sehingga makna yang terkandung dalam teks dapat direpresentasikan dalam bentuk vektor numerik. Hasil dari proses ini berupa *sentence embedding* berdimensi 384 yang digunakan sebagai basis pencarian rekomendasi film.

Berbeda dengan *query* pengguna yang diproses pada saat pencarian dilakukan, pembentukan *embedding* sinopsis dilakukan satu kali pada seluruh data film yang terdapat dalam dataset. *Embedding* yang dihasilkan kemudian disimpan sehingga dapat digunakan kembali pada proses pencarian tanpa perlu dilakukan pembentukan embedding ulang

Pada penelitian ini, baik *query* pengguna maupun sinopsis film diproses menggunakan model yang sama agar menghasilkan representasi pada ruang vektor yang seragam. Alur pembentukan *embedding* pada penelitian ini ditunjukkan pada Gambar 3.3.



Gambar 3. 3 Proses SBERT *Embedding*

3.4.1 Tokenisasi

Pada tahap tokenisasi, *query* pengguna dan sinopsis film hasil *preprocessing* diubah menjadi token-token yang dapat dipahami oleh model SBERT. Proses ini menggunakan WordPiece Tokenizer yang merupakan tokenizer bawaan model all-MiniLM-L6-v2. Tujuan tokenisasi adalah mengubah teks yang masih berupa kalimat menjadi unit-unit token sehingga dapat diproses oleh model transformer pada tahap berikutnya. Pada Tabel 3.7 merupakan contoh hasil tokenisasi *wordpiece*:

Tabel 3.7 Contoh Hasil *SBERT Tokenization*

Sinopsis	***Preprocessing***	**Tokens**	**Token ID**
"As Harry begins his sixth year at Hogwarts, he discovers an old book marked as Property of the Half-Blood Prince and begins to learn more about Lord Voldemort's dark past."	*['harry', 'begins', 'sixth', 'year', 'hogwarts', 'discovers', 'old', 'book', 'marked', 'property', 'half', 'blood', 'prince', 'begins', 'learn', 'lord', 'voldemort', 'dark', 'past']*	*['[CLS]', 'harry', 'begins', 'sixth', 'year', 'hogwarts', 'discovers', 'old','book', 'marked', 'property', 'half', 'blood', 'prince', 'begins', 'learn', 'lord', 'vol', '##de', '##mort', 'dark', 'past', '[SEP]']*	[CLS] → 101 harry → 4302 begins → 4269 sixth → 4369 year → 2095 ... vol → 5285 ##de → 3207 ##mort → 15256 ... [SEP] → 102

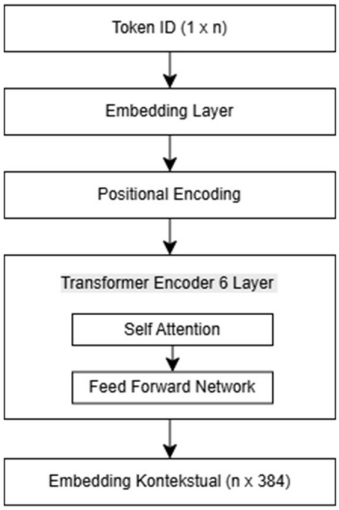
Pada hasil tokenisasi terlihat bahwa kata *voldemort* dipecah menjadi beberapa subkata yaitu: ‘vol’, ‘##de’, ‘##mort’, karena tokenizer tidak menemukan kata *voldemort* secara utuh pada vocabulary model. Dengan menggunakan pendekatan WordPiece, kata yang tidak tersedia dapat direpresentasikan melalui kombinasi subkata yang lebih umum tanpa kehilangan informasi semantik yang terkandung di dalamnya. Setelah tokenisasi selesai, setiap token dikonversi menjadi token ID sesuai vocabulary yang dimiliki model all-MiniLM-L6-v2. Hasil konversi tersebut menghasilkan urutan token ID yang akan digunakan sebagai masukan pada tahap encoding. Secara umum, apabila sebuah sinopsis menghasilkan n token setelah proses tokenisasi, maka *output* tahap ini berupa vektor token ID dengan panjang n. Token-token tersebut selanjutnya diproses menggunakan *backbone* all-MiniLM-L6-v2 untuk menghasilkan representasi kontekstual pada tahap encoding.

3.4.2 Encoding dengan Arsitektur *Backbone* all-MiniLM-L6-v2

Pada tahap tokenisasi menghasilkan urutan token ID, data tersebut diproses melalui tahap encoding menggunakan *backbone* all-MiniLM-L6-v2 yang merupakan bagian utama dari model SBERT. *Backbone* ini terdiri atas enam lapisan *Transformer Encoder* yang bertugas mengubah token ID menjadi representasi kontekstual. Setiap encoder memiliki mekanisme *Multi-head self-attention* dan *Feed forward network* (FFN) yang memungkinkan model memahami hubungan antar token dalam kalimat.

Melalui proses encoding, token yang semula hanya berupa identitas numerik secara bertahap diubah menjadi representasi vektor yang mampu merepresentasikan makna berdasarkan konteks kalimat. Apabila hasil tokenisasi

menghasilkan n token, maka output akhir proses encoding berupa representasi kontekstual dengan ukuran n x 384, dimana setiap token direpresentasikan oleh vektor berdimensi 384 sesuai *hidden size* model all-MiniLM-L6-v2. Alur pemrosesan data pada *backbone* all-MiniLM-L6-v2 ditunjukkan pada Gambar 3.4.



Gambar 3. 4 Proses *Backbone* all-MiniLM-L6-v2

Apabila hasil tokenisasi menghasilkan n token ID, data tersebut masih berupa bilangan bulat yang hanya berfungsi sebagai identitas token dan belum memiliki informasi semantik. Oleh karena itu token ID terlebih dahulu diproses pada *embedding layer*. Tahapan-tahapan sebelum memasuki encoding sebagai berikut:

1. *Embedding layer*

   Embedding layer bertugas mengubah setiap token ID menjadi vektor numerik berdimensi 384. Proses *embedding* dapat dinyatakan menggunakan Persamaan 3.1:

$$E_i = X_i W_e \tag{3.1}$$

Keterangan:
Ei: token *embedding* ke-i
Xi: representasi token ID ke-i
We: matriks *embedding*

Setelah token diubah menjadi representasi numerik, informasi posisi token dipertahankan melalui mekanisme positional *embedding* yang telah menjadi bagian dari arsitektur transformer pada model all-MiniLM-L6-v2.

2. *Positional encoding*

Tahap *positional encoding* melakukan penambahan informasi mengenai posisi token agar model dapat membedakan urutan kata dalam teks Proses tersebut dapat dinyatakan menggunakan Persamaan 3.2:

$$PE_i = E_i + P_i \tag{3.2}$$

Keterangan:
PEi: posisi token *embedding* ke-i
Ei: token *embedding* ke-i
Pe: vector posisi token ke-i

Output dari tahap ini selanjutnya digunakan sebagai input pada *Transformer Encoder* untuk diproses menggunakan mekanisme *Multi-head self-attention* dan *Feed forward network*. Persamaan 3.1 dan 3.2 diperoleh dari penelitian yang dilakukan oleh (Devlin et al., 2018).

3. *Multi-head self-attention*

Multi-head self-attention bertujuan untuk mempelajari hubungan antar token dalam kalimat sehingga setiap token dapat memperoleh informasi kontekstual dari token lainnya. Proses *self-attention* diawali dengan membentuk tiga representasi untuk setiap token *embedding*, yaitu *Query* (Q), *Key* (K), dan *Value* (V). sebagaimana yang tertera pada penelitian yang dilakukan oleh Vaswani et

al, 2017, ketiga representasi tersebut diperoleh melalui transformasi linear yang dinyatakan pada Persamaan 3.3, Persamaan 3.4, dan Persamaan 3.5:

$$Q = XW_Q \tag{3.3}$$

$$K = XW_K \tag{3.4}$$

$$V = XW_V \tag{3.5}$$

Selanjutnya tingkat perhatian antar token dihitung menggunakan mekanisme *scaled dot-product attention* yang dinyatakan pada Persamaan 3.6:

$$Attention(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{3.6}$$

Melalui proses tersebut, model menghitung tingkat keterkaitan setiap token terhadap token lainnya dan memberikan bobot perhatian yang berbeda sesuai tingkat relevansinya. Token yang memiliki hubungan lebih kuat akan memperoleh bobot perhatian yang lebih besar dibandingkan token yang kurang relevan.

Pada model all-MiniLM-L6-v2, proses *attention* dilakukan menggunakan beberapa *head* secara paralel yang disebut *Multi-head self-attention*. Output dari setiap *head* kemudian digabungkan dan ditransformasikan kembali menggunakan Persamaan 3.7:

$$\text{Multi}Head(Q, K, V) = \text{Concat}(head_1, \dots, head_h)W_O \tag{3.7}$$

Keterangan:
Q: *query* matriks
K: *key* matriks
V: *value* matriks
X: output *positional embedding*
W_Q, W_K, W_V: matriks bobot yang dipelajari model
d_k: dimensi *key* vektor
$softmax$: $fungsi\ normalisasi\ bobot$
$head_i$: output *attention* pada *head* ke-
$head_1, \dots, head_h$: output seluruh *attention head*
Concat: proses penggabungan seluruh *head*

W_O: matriks bobot output
h: jumlah *attention head*

Pada tahap ini ukuran data tidak mengalami perubahan. Perubahan yang terjadi terletak pada nilai setiap dimensi yang telah diperbarui berdasarkan hubungan antar token sehingga menghasilkan representasi yang lebih kontekstual. Output dari tahap ini selanjutnya diproses oleh *Feed forward network* untuk memperkaya representasi fitur yang telah diperoleh.

4. *Feed forward network* (FFN)

Pada tahap ini berfungsi untuk memperkaya representasi fitur yang telah diperoleh dari proses *attention* sehingga model dapat mempelajari hubungan yang lebih kompleks antar fitur semantik. Tahap pertama, input x ditransformasikan menggunakan bobot W_1 dan bias b_1. Hasil transformasi tersebut kemudian diproses menggunakan fungsi aktivasi GELU (*Gaussian Error Linear Unit*). Fungsi aktivasi ini bertugas menentukan informasi mana yang perlu dipertahankan dan informasi mana yang perlu dikurangi pengaruhnya sebelum diteruskan ke lapisan berikutnya. fungsi GELU dinyatakan menggunakan Persamaan 3.8:

$$\mathrm{FFN}(x) = W_2\big(\mathrm{GELU}(W_1 x + b_1)\big) + b_2 \tag{3.8}$$

Keterangan:
FFN_x: output FFN
x: $input\ dari\ Multi - Head\ Self - Attention$
W_1, W_2: matriks bobot lapisan ½ yang dipelajari model
b_1, b_2: 1/2
$GELU$: fungsi aktivasi *Gaussian Error Linear Unit*

Perubahan yang terjadi terletak pada nilai setiap dimensi *embedding* yang telah diperkaya melalui transformasi linear dan fungsi aktivasi nonlinier. Output dari

tahap ini selanjutnya diproses menggunakan *residual connection* dan *layer normalization* sebelum diteruskan ke *Transformer Encoder* berikutnya.

5. *Residual connection* dan *Layer normalization*

   Tahap *Residual connection* dan *Layer normalization* bertujuan untuk menjaga informasi penting yang telah diperoleh pada proses sebelumnya serta menghasilkan representasi yang lebih stabil sebelum diteruskan ke *Transformer Encoder* berikutnya. *Residual connection* bekerja dengan menambahkan kembali masukan sebelumnya ke *output* suatu lapisan. Proses ini dilakukan untuk mengurangi hilangnya informasi ketika data melewati banyak lapisan encoder. Sebagaimana penelitian yang dilakukan oleh (He et al., 2016), perhitungan *Residual connection* dinyatakan menggunakan Persamaan 3.9:

$$R = X + F(X) \tag{3.9}$$

   Setelah proses *Residual connection*, data diproses menggunakan *Layer normalization*. Tahap ini bertujuan untuk menstabilkan distribusi nilai pada setiap dimensi *embedding* sehingga proses pembelajaran dan representasi fitur menjadi lebih konsisten. Berdasarkan referensi (Ba et al., 2016), perhitungan *Layer normalization* dinyatakan menggunakan Persamaan 3.10:

$$\mathrm{LN}(x) = \gamma \frac{x-\mu}{\sqrt{\sigma^2+\epsilon}} + \beta \tag{3.10}$$

Keterangan:
R: *output residual connection*
X: input suatu lapisan
F(X): output lapisan yang diproses
LN_x,: data input
μ: *mean* dari seluruh elemen pada vektor masukan
σ^2: varians dari seluruh elemen pada vektor masukan
ϵ: konstanta bernilai sangat kecil untuk mencegah pembagian dengan nol
γ: parameter skala yang dipelajari model
β: parameter pergeseran yang dipelajari model
$GELU$: fungsi aktivasi *Gaussian Error Linear Unit*

Setelah melewati seluruh tahapan pada *Transformer Encoder* model yang dilakukan sebanyak enam kali, setiap token memperoleh informasi kontekstual yang semakin kaya karena terus diperbarui berdasarkan hubungan dengan token-token lain dalam kalimat. Hasil akhir dari tahap encoding berupa representasi kontekstual untuk setiap token dengan ukuran matriks n x 384. Ukuran data tidak mengalami perubahan selama proses encoding. Namun nilai pada setiap dimensi *embedding* telah diperbarui melalui enam *Transformer Encoder* sehingga mampu merepresentasikan makna token berdasarkan konteks kalimat secara keseluruhan. Representasi kontekstual tersebut selanjutnya digunakan sebagai masukan pada tahap pooling untuk menghasilkan satu vektor *embedding* yang mewakili keseluruhan kalimat.

3.4.3 *Pooling*

Tahap *pooling* merupakan tahap akhir dalam proses pembentukan *Embedding* pada SBERT yang bertujuan untuk merangkum seluruh representasi token menjadi satu vektor kalimat dari input n x 384. Pada penelitian ini digunakan metode *mean pooling*, yaitu dengan menghitung rata-rata seluruh representasi token yang dihasilkan oleh encoder. Metode ini dipilih karena mampu menghasilkan representasi kalimat yang stabil serta mempertimbangkan kontribusi setiap token secara merata dalam pembentukan *embedding* kalimat. Menurut Reimers dan Gurevych (2019), *mean pooling* merupakan salah satu metode *pooling* yang efektif untuk menghasilkan *Sentence embedding* pada model SBERT.

Secara matematis, proses *mean pooling* dinyatakan menggunakan Persamaan 3.11:

$$e = \frac{1}{n}\sum_{i=1}^{n} h_i \quad (3.11)$$

Keterangan:
e: vektor representasi kalimat hasil *Pooling*.
h_i: representasi kontekstual token ke-i
n: jumlah token

Melalui proses tersebut, seluruh representasi token dirata-ratakan sehingga menghasilkan satu vektor yang merepresentasikan makna keseluruhan kalimat. Pada tahap ini terjadi perubahan ukuran data dari $H \in \mathbb{R}^{n\times 384}$ menjadi $e \in \mathbb{R}^{1\times 384}$. Perubahan tersebut menunjukkan bahwa sekumpulan token *embedding* telah diringkas menjadi satu *Sentence embedding* berdimensi 384. Vektor hasil *pooling* inilah yang selanjutnya digunakan pada tahap perhitungan *Cosine Similarity* untuk mengukur tingkat kesamaan antara *query* pengguna dan sinopsis film pada dataset.

Pada penelitian ini, proses encoding sinopsis dilakukan terhadap seluruh data sinopsis film yang terdapat pada dataset. Karena sinopsis umumnya terdiri atas kalimat yang lebih panjang dengan jumlah token yang lebih banyak dibandingkan *query* pengguna, proses encoding menghasilkan representasi semantik yang lebih kaya terhadap informasi tema, alur cerita, karakter, dan konteks film. Hasil encoding sinopsis kemudian digunakan sebagai dasar pembentukan *embedding* sinopsis yang disimpan dan digunakan pada proses pencarian rekomendasi. Berbeda dengan encoding sinopsis, proses encoding *query* dilakukan secara dinamis setiap kali pengguna memasukkan kata kunci pencarian ke dalam sistem. *Query* pengguna umumnya memiliki jumlah token yang lebih sedikit sehingga menghasilkan representasi kontekstual yang lebih ringkas. Meskipun demikian, penggunaan model all-MiniLM-L6-v2 yang sama memastikan bahwa *embedding query* dan *embedding* sinopsis berada pada ruang vektor yang identik. Dengan

demikian, tingkat kemiripan semantik antara *query* pengguna *dan* sinopsis film dapat dihitung secara akurat menggunakan metode *Cosine* Similarity pada tahap selanjutnya.

3.5 Cosine Similarity

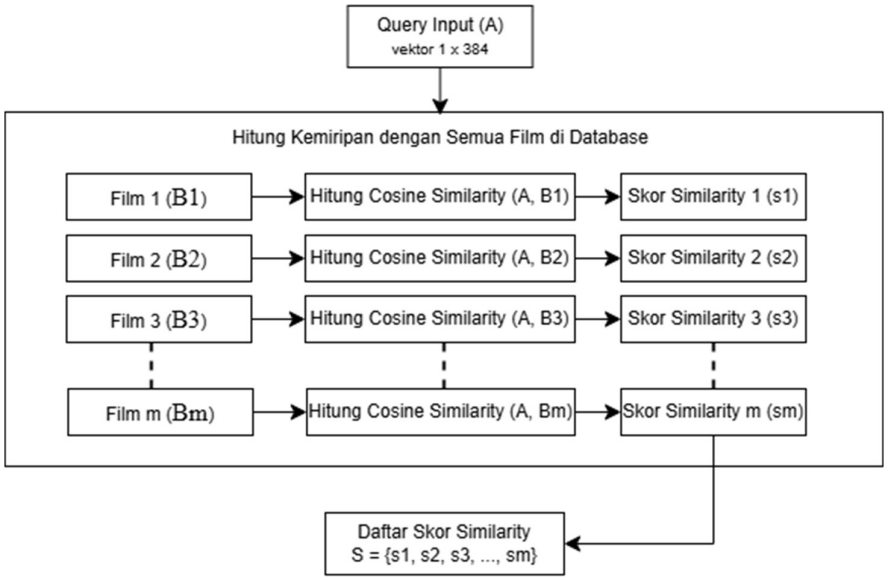
Cosine Similarity merupakan metode yang digunakan untuk mengukur tingkat kemiripan antara dua vektor dalam ruang multidimensi. Pada penelitian ini, *Cosine Similarity* digunakan untuk mengukur tingkat kemiripan semantik antara *embedding query* pengguna dengan *embedding* setiap sinopsis film yang terdapat pada dataset. *Cosine Similarity* mengukur kemiripan dua vektor berdasarkan sudut yang terbentuk di antara keduanya. Semakin kecil sudut antar vektor, maka semakin tinggi tingkat kemiripan kedua teks yang dibandingkan. Perhitungan *Cosine Similarity* dinyatakan menggunakan Persamaan 3.12 (Nurma Romihim Fadlilah, 2025):

$$\textit{Cosine Similarity}\ (\mathrm{A}, \mathrm{B}) = \frac{\mathrm{A} \cdot \mathrm{B}}{|\mathrm{A}||\mathrm{B}|} \tag{3.12}$$

Keterangan:
A: *embedding query* pengguna
B: *embedding* sinopsis film
A · B: *dot product* antara vektor A dan vektor B
|A|: Panjang vektor A.
|B|: Panjang vektor B.

Pada tahap ini, *embedding query* pengguna digunakan sebagai vektor acuan yang dibandingkan dengan seluruh *embedding* sinopsis film yang telah tersimpan dalam dataset. Proses perbandingan dilakukan secara iteratif terhadap setiap film sehingga sistem dapat memperoleh nilai kemiripan untuk seluruh data film. Apabila jumlah film dalam dataset dinyatakan sebagai m, maka proses *Cosine Similarity*

dilakukan sebanyak m kali. Setiap iterasi menghasilkan satu nilai *similarity* yang menunjukkan tingkat kemiripan antara *query* pengguna dan satu sinopsis film. Proses iterasi tersebut dapat dilihat pada Gambar 3.5 sebagai berikut:



Gambar 3. 5 Implementasi SBERT *Embedding*

Hasil dari proses tersebut berupa sekumpulan nilai *similarity* yang dapat direpresentasikan sebagai $S = \{s_1, s_2, s_3, \dots, s_m\}$, dengan S merupakan himpunan nilai *similarity* dan s_i nilai *similarity* ke-i. Penjelasan tentang rumus yang digunakan bersumber dari penelitian (Christopher D. Manning *et al*., 2008)

Pada skor *similarity* jika semakin mendekati nilai 1 menunjukkan bahwa *query* pengguna dan sinopsis film memiliki kemiripan semantik yang semakin tinggi, sedangkan nilai yang mendekati 0 menunjukkan tingkat kemiripan yang rendah. Oleh karena itu, film dengan nilai *similarity* terbesar dianggap paling relevan terhadap *query* yang diberikan pengguna. Hasil perhitungan *Cosine Similarity* selanjutnya digunakan pada tahap perangkingan rekomendasi untuk

mengurutkan film berdasarkan nilai *similarity* tertinggi sehingga dapat diperoleh rekomendasi film yang paling sesuai dengan kebutuhan pengguna

3.6 Perangkingan Top-K

Tahap pertama dalam proses perangkingan adalah melakukan pengurutan nilai *similarity* secara *descending*. Pengurutan dilakukan mulai dari nilai *similarity* tertinggi hingga nilai *similarity* terendah. Tujuan dari proses ini adalah menempatkan film dengan tingkat kemiripan tertinggi pada posisi teratas sehingga lebih mudah dipilih sebagai rekomendasi. Sebelum dilakukan pengurutan, hasil *Cosine Similarity* dapat direpresentasikan sebagai pasangan data $(title_i, similarity_i)$, Proses *sorting* dilakukan dengan menjadikan nilai *similarity* sebagai kunci pengurutan. Setelah seluruh data diurutkan, film dengan nilai *similarity* terbesar akan berada pada peringkat pertama, diikuti oleh film dengan nilai *similarity* yang lebih rendah secara berurutan.

Hasil dari tahap ini berupa daftar film rekomendasi yang telah diurutkan berdasarkan nilai *similarity* tertinggi, seperti contoh pada Gambar 3.6 berikut:

Film	Cosine Similarity
Film A	0.85
Film B	0.92
Film C	0.78

menjadi

Ranking	Film	Similarity
1	Film B	0.92
2	Film A	0.85
3	Film C	0.78

Gambar 3. 6 Contoh Hasil Perankingan

Dalam penelitian ini, hasil pengurutan tersebut digunakan untuk mengambil sejumlah film teratas yaitu Top-5 atau Top-10, yang kemudian ditampilkan kepada pengguna sebagai hasil rekomendasi. Proses *sorting* ini tidak mengubah nilai *similarity*, melainkan hanya menyusun ulang urutan data berdasarkan tingkat

relevansi. Dengan demikian, kualitas rekomendasi sangat bergantung pada hasil perhitungan *Cosine Similarity* sebelumnya, sedangkan tahap *sorting* berperan dalam menentukan prioritas penyajian hasil kepada pengguna.

3.7 Skenario Uji Coba

Pengujian dilakukan untuk mengevaluasi performa sistem rekomendasi film berbasis sinopsis yang dibangun menggunakan SBERT dan *Cosine Similarity*. Evaluasi dilakukan dengan membandingkan penggunaan *preprocessing*, panjang *query*, dan jumlah rekomendasi yang ditampilkan oleh sistem. Hasil rekomendasi kemudian dievaluasi menggunakan metrik *Precision*, *Mean Average Precision* (MAP), *Mean Reciprocal Rank* (MRR), dan *Normalized Discounted Cumulative Gain* (nDCG).

Pada penelitian ini digunakan delapan skenario pengujian yang merupakan kombinasi dari penggunaan *preprocessing*, panjang *query*, dan jumlah rekomendasi yang ditampilkan. Penggunaan *preprocessing* bertujuan untuk mengetahui pengaruh proses pembersihan teks terhadap kualitas rekomendasi yang dihasilkan. Selain itu, pengujian juga dilakukan menggunakan *query* pendek dan *query* panjang untuk mengetahui pengaruh jumlah kata terhadap kemampuan model dalam memahami kebutuhan pengguna. Variasi jumlah rekomendasi yang ditampilkan terdiri atas Top-5 dan Top-10 untuk mengevaluasi kualitas rekomendasi pada jumlah hasil yang berbeda. Delapan skenario pengujian yang digunakan pada penelitian ini ditunjukkan pada Tabel 3.8.

Tabel 3. 8 Skenario Uji Coba

No.	Skenario Pengujian			Tujuan
1.		Menggunakan *Query* pendek (< 5)	Top-5	
2.			Top-10	

No.	Skenario Pengujian			Tujuan
3.	Rekomendasi film dengan *Preprocessing*	Menggunakan *Query* panjang (> 5)	Top-5	Mengukur performa sistem ketika teks sudah dibersihkan dan diproses
4.			Top-10	
5.	Rekomendasi film tanpa *Preprocessing*	Menggunakan *Query* pendek (< 5)	Top-5	Mengukur performa sistem ketika teks mentah digunakan
6.			Top-10	
7.		Menggunakan *Query* panjang (> 5)	Top-5	
8.			Top-10	

Data pengujian menggunakan total 40 *query* yang terdiri atas 20 *query* pendek dan 20 *query* panjang. *Query* disusun berdasarkan berbagai genre film yang terdapat pada dataset TMDB, seperti *Action, Adventure, Comedy, Drama, Fantasy, Horror, Mystery, Romance, Science Fiction, Thriller, Documentary, War, Western, Musical, Crime, Family, Animation, History, Foreign, dan Superhero*. *Query* pendek digunakan untuk merepresentasikan kebutuhan pencarian yang lebih ringkas, sedangkan *query* panjang digunakan untuk merepresentasikan kebutuhan pencarian yang lebih spesifik dan deskriptif. Contoh *query* yang digunakan dalam penelitian ini ditunjukkan pada Tabel 3.9.

Tabel 3. 9 *Query* Panjang dan Pendek

No.	***Query* Pendek**	***Query* Panjang**
1.	*king and queens*	*a story about kings and queens dealing with royal politics, power struggles, and life in a kingdom*
2.	*sea adventure*	*an adventure story about a journey across the ocean facing dangers, storms, and unexpected challenges at sea*
3.	*spy movie*	*a story about a secret agent carrying out dangerous missions involving espionage, intelligence, and undercover operations*
4.	*space movie*	*a story about space exploration involving astronauts traveling through space, facing challenges, and discovering new worlds*
5.	*creative animated film*	*animated movie with creative visuals and engaging storytelling*
6.	*light comedy movie*	*comedy movie that is entertaining and full of humor*
7.	*dark crime story*	*movie about crime, mafia, and criminal investigations*
8.	*real documentary film*	*documentary film presenting real-life stories and information*
9.	*emotional drama movie*	*movie with emotional story and deep life conflicts*
10.	*warm family movie*	*movie suitable for family viewing with heartwarming story*
11.	*stories about past events*	*movie based on historical events from the past*
12.	*scary horror movie*	*scary movie with tense and frightening atmosphere*
13.	*musical themed movie*	*movie focused on music, singers, or musical journeys*
14.	*puzzling mystery movie*	*movie with puzzles, secrets, and unpredictable storyline*
15.	*romantic love story*	*movie about love story that touches the heart*

No.	*Query* Pendek	*Query* Panjang
16.	*suspense thriller movie*	*movie with high tension and suspenseful storyline*
17.	*intense war movie*	*movie about war, battles, and military strategy*
18.	*classic western movie*	*movie about cowboys and life in the old west*
19.	*international foreign film*	*movie from other countries with different culture and language*
20.	*superhero film and comedy*	*a lighthearted superhero story where a hero with special powers fights villains while experiencing humorous and comedic situations*

Setelah sistem menghasilkan daftar rekomendasi film berdasarkan nilai *Cosine Similarity*, dilakukan proses validasi untuk menentukan tingkat relevansi hasil rekomendasi terhadap *query* yang diberikan. Validasi dilakukan oleh satu validator yang memiliki kemampuan memahami teks berbahasa inggris. Pemilihan validator dilakukan karena seluruh sinopsis film pada dataset TMDB menggunakan bahasa inggris sehingga diperlukan penilaian yang mampu memahami makna dan konteks cerita yang terkandung dalam sinopsis.

Validator melakukan penilaian dengan membandingkan *query* yang digunakan pada pengujian dengan sinopsis film yang direkomendasikan oleh sistem. Penilaian dilakukan berdasarkan kesesuaian tema cerita, konteks naratif, konflik utama, karakter, dan genre yang terkandung dalam sinopsis terhadap maksud *query* pengguna. Hasil penilaian menggunakan relevansi biner yang ditunjukkan pada Tabel 3.10.

Tabel 3. 10 Skema Validasi Relevansi

Nilai	Keterangan
1	Film relevan dengan *query*
0	Film tidak relevan dengan *query*

Untuk mengukur kualitas rekomendasi yang dihasilkan sistem, penelitian ini menggunakan empat metrik evaluasi, yaitu *Precision*, *Mean Average Precision*

(MAP), *Mean Reciprocal Rank* (MRR), dan *Normalized Discounted Cumulative Gain* (nDCG). Keempat metrik tersebut dipilih karena mampu mengevaluasi sistem rekomendasi dari berbagai aspek, mulai dari tingkat relevansi rekomendasi yang diberikan, posisi kemunculan film relevan pada daftar rekomendasi, hingga kualitas urutan (ranking) hasil rekomendasi yang dihasilkan sistem.

a. *Precision*

Precision digunakan untuk mengukur proporsi film yang relevan dari seluruh film yang direkomendasikan sistem. Metrik ini menunjukkan seberapa akurat sistem dalam menghasilkan rekomendasi yang sesuai dengan *query* pengguna. Semakin tinggi nilai *precision*, maka semakin banyak film relevan yang berhasil direkomendasikan oleh sistem. Perhitungan *precision* dilakukan menggunakan Persamaan (3.13).

$$Precision = \frac{Total\ Relevan\ Item}{Total\ Rekomendasi\ Item} \quad 3.13$$

b. *Mean Average Precision* (MAP)

Mean Average Precision (MAP) digunakan untuk mengevaluasi kemampuan sistem dalam menempatkan film relevan pada posisi yang lebih tinggi dalam daftar rekomendasi. MAP diperoleh dari rata-rata nilai Average *Precision* (AP) pada seluruh *query* pengujian. Semakin tinggi nilai MAP, maka semakin baik sistem dalam mengurutkan film relevan pada posisi atas. Perhitungan *Average Precision* (AP) dan MAP dilakukan menggunakan Persamaan (3.14) dan Persamaan (3.15).

$$AP = \frac{1}{RN_r}\sum_{k=1}^{K} P(k)\ \cdot rel(k) \quad 3.14$$

$$MAP = \frac{1}{Q}\sum_{q=1}^{Q} AP_q \quad 3.15$$

Keterangan:
N_r : jumlah item relevan
K : jumlah rekomendasi yang dipertimbangkan
$P(k)$: *Precision* posisi ke-K
$rel(k)$: nilai relevansi pada posisi ke-k
Q :jumlah *Query*
AP_q :*average Precision* untuk *Query* ke-q

c. *Mean Reciprocal Rank* (MRR)

Mean Reciprocal Rank (MRR) digunakan untuk mengukur seberapa cepat sistem menampilkan film relevan pertama pada daftar rekomendasi. Nilai MRR akan semakin tinggi apabila film relevan pertama muncul pada posisi yang semakin atas. Perhitungan MRR dilakukan menggunakan Persamaan (3.16).

$$MRR = \frac{1}{Q}\sum_{q=1}^{Q} \frac{1}{rank_q} \quad 3.16$$

Keterangan:
Q : jumlah *Query*
rank-q: posisi rekomendasi relevan pertama untuk *Query* ke-q

d. *Normalized Discounted Cumulative Gain* (nDCG)

Normalized Discounted Cumulative Gain (nDCG) digunakan untuk mengevaluasi kualitas urutan rekomendasi berdasarkan posisi film relevan pada daftar rekomendasi. Metrik ini memberikan bobot yang lebih besar pada film relevan yang muncul pada posisi teratas sehingga sangat sesuai digunakan pada sistem rekomendasi berbasis ranking. Perhitungan nDCG dilakukan melalui perhitungan *Discounted Cumulative Gain* (DCG) dan *Ideal Discounted*

Cumulative Gain (IDCG) yang dinyatakan pada Persamaan (3.17), Persamaan (3.18), dan Persamaan (3.19).

$$DCGK = \sum_{i=1}^{K} \frac{2^{rel_i} - 1}{log_2(i+1)} \quad 3.17$$

$$IDCGK = \sum_{i=1}^{K} \frac{2^{rel_i} - 1}{log_2(i+1)} \quad 3.18$$

$$nDCGK = \frac{DCGK}{IDCGK} \quad 3.19$$

Keterangan:
Q : jumlah *Query*
rel-I : relevansi item di posisi
K : jumlah rekomendasi Top-K
DCGK : DCG pada Top-K
IDCGK : DCG maksimal (ideal) pada Top-K,

Melalui keempat metrik tersebut, kualitas sistem rekomendasi dapat dievaluasi secara menyeluruh, baik dari aspek ketepatan rekomendasi, posisi kemunculan film relevan, maupun kualitas urutan hasil rekomendasi yang diberikan kepada pengguna

BAB IV

UJI COBA DAN PEMBAHASAN

4.1 Hasil Uji Coba

Pada subbab ini dilakukan analisis terhadap hasil pengujian yang telah diperoleh pada setiap skenario uji. Pembahasan difokuskan pada interpretasi nilai metrik evaluasi, yaitu *Precision*, *Mean Average Precision* (MAP), *Mean Reciprocal Rank* (MRR), dan *Normalized Discounted Cumulative Gain* (nDCG), untuk mengetahui sejauh mana sistem mampu menghasilkan rekomendasi yang relevan dan terurut dengan baik. Selain nilai rata-rata, ditampilkan pula nilai standar deviasi untuk menggambarkan tingkat variasi hasil antar *query* pada setiap skenario pengujian. Standar deviasi dihitung menggunakan sample *standard deviation* (n−1) karena data yang digunakan merupakan sampel *query* pengujian, bukan keseluruhan populasi *query* yang mungkin digunakan pada sistem rekomendasi.Analisis dilakukan dengan membandingkan hasil antar skenario yang memiliki perbedaan perlakuan, seperti penggunaan *preprocessing* dan non-*preprocessing*, variasi panjang *Query*, serta perbedaan jumlah rekomendasi. Melalui perbandingan ini, dapat diketahui faktor-faktor yang mempengaruhi kinerja sistem serta kondisi yang menghasilkan performa terbaik.

4.1.1 Skenario Uji Coba 1

Skenario 1 merupakan pengujian yang menggunakan *preprocessing* pada *query* pendek dengan panjang kurang dari lima kata dan menghasilkan lima rekomendasi teratas Top-5. Pengujian ini dilakukan untuk mengetahui kemampuan

sistem dalam menghasilkan rekomendasi film yang relevan ketika menggunakan *query* sederhana dengan jumlah rekomendasi yang terbatas. Hasil evaluasi rata-rata pada Skenario 1 ditunjukkan pada Tabel 4.1

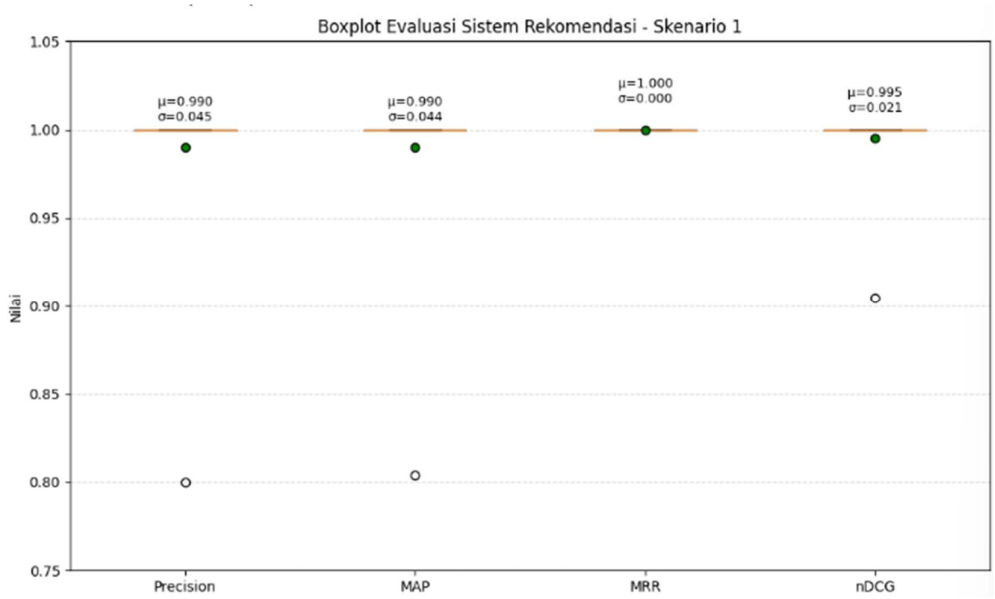
Tabel 4. 1 Hasil Skenario Uji Coba 1

Query	***Precision***	**MAP**	**MRR**	**nDCG**
Query 1	1	1	1	1
Query 2	1	1	1	1
Query 3	1	1	1	1
Query 4	1	1	1	1
Query 5	1	1	1	1
Query 6	1	1	1	1
Query 7	1	1	1	1
Query 8	1	1	1	1
Query 9	1	1	1	1
Query 10	1	1	1	1
Query 11	1	1	1	1
Query 12	1	1	1	1
Query 13	1	1	1	1
Query 14	1	1	1	1
Query 15	1	1	1	1
Query 16	1	1	1	1
Query 17	1	1	1	1
Query 18	0.8	0.804	1	0.905
Query 19	1	1	1	1
Query 20	1	1	1	1
Rata-rata	0.990	0.990	1	0.995
Standar Deviasi	0.045	0.044	0	0.021

Berdasarkan Tabel 4.1, Skenario 1 memperoleh nilai *Precision* sebesar 0.99, MAP sebesar 0.990, MRR sebesar 1, dan nDCG sebesar 0.995. Nilai tersebut menunjukkan bahwa sistem mampu menghasilkan rekomendasi yang relevan terhadap *query* yang diberikan.

Nilai *Precision* yang mendekati 1 menunjukkan bahwa hampir seluruh film yang direkomendasikan pada Top-5 dinilai relevan oleh validator. Selain itu, nilai MAP yang tinggi menunjukkan bahwa film-film relevan cenderung berada pada posisi atas dalam daftar rekomendasi. Nilai MRR sebesar 1 mengindikasikan bahwa film relevan pertama selalu muncul pada peringkat pertama hasil rekomendasi.

Sementara itu, nilai nDCG sebesar 0.995 menunjukkan bahwa film-film yang relevan cenderung berada pada posisi atas daftar rekomendasi yang dihasilkan sistem. Untuk memperjelas hasil evaluasi pada Skenario 1, visualisasi nilai *Precision*, MAP, MRR, dan nDCG ditunjukkan pada Gambar 4.1.



Gambar 4. 1 *Boxplot* Rata-rata Hasil Skenario Uji Coba 1

Berdasarkan Gambar 4.1, distribusi nilai *Precision*, MAP, MRR, dan nDCG pada Skenario 1 menunjukkan performa sistem yang sangat stabil. Hal ini terlihat dari posisi median seluruh metrik yang berada sangat dekat dengan nilai 1. Selain itu, nilai standar deviasi yang relatif kecil pada *Precision* $\sigma = 0{,}045$, MAP $\sigma = 0{,}044$, dan nDCG $\sigma = 0{,}021$ menunjukkan bahwa hasil evaluasi antar *query* cenderung konsisten. Sementara itu, nilai MRR memiliki standar deviasi sebesar 0 yang menunjukkan bahwa film relevan pertama selalu muncul pada peringkat teratas untuk seluruh *query* yang diuji.

4.1.2 Skenario Uji Coba 2

Skenario 2 merupakan pengujian yang menggunakan *preprocessing* pada *query* pendek dengan panjang kurang dari lima kata dan menghasilkan sepuluh rekomendasi teratas Top-10. Pengujian ini dilakukan untuk mengetahui kemampuan sistem dalam menghasilkan rekomendasi film yang relevan ketika jumlah rekomendasi yang ditampilkan lebih banyak dibandingkan Skenario 1. Hasil evaluasi rata-rata pada Skenario 2 ditunjukkan pada Tabel 4.2.

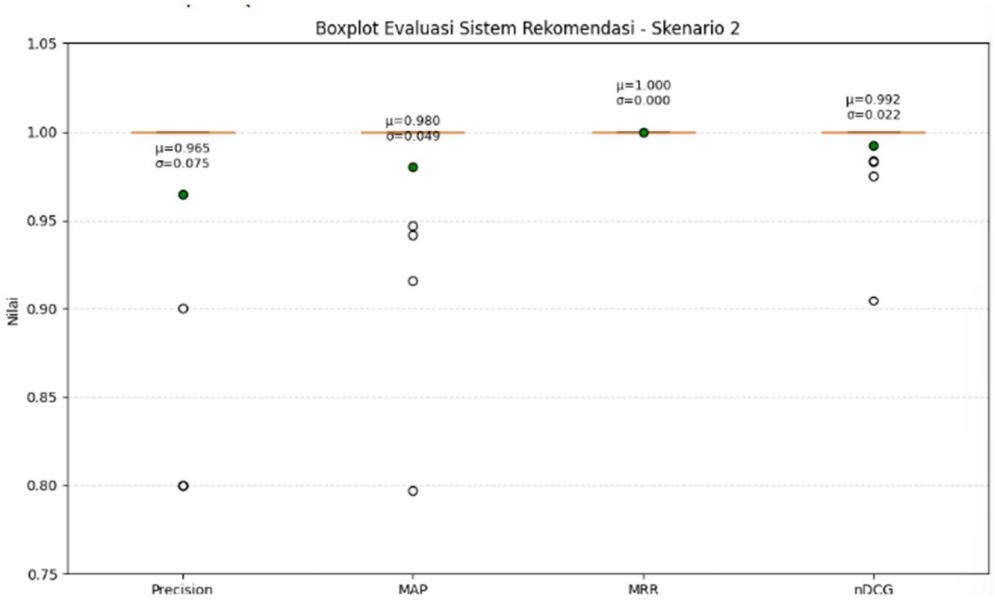
Tabel 4. 2 Hasil Skenario Uji Coba 2

Query	***Precision***	**MAP**	**MRR**	**nDCG**
Query 1	1	1	1	1
Query 2	0.9	0.947	1	0.984
Query 3	1	1	1	1
Query 4	1	1	1	1
Query 5	1	1	1	1
Query 6	1	1	1	1
Query 7	1	1	1	1
Query 8	1	1	1	1
Query 9	1	1	1	1
Query 10	1	1	1	1
Query 11	0.8	0.942	1	0.983
Query 12	1	1	1	1
Query 13	1	1	1	1
Query 14	1	1	1	1
Query 15	1	1	1	1
Query 16	1	1	1	1
Query 17	1	1	1	1
Query 18	0.8	0.797	1	0.905
Query 19	0.8	0.916	1	0.975
Query 20	1	1	1	1
Rata-rata	0.965	0.980	1	0.992
Standar Deviasi	0.075	0.049	0	0.022

Berdasarkan Tabel 4.2, Skenario 2 memperoleh nilai *Precision* sebesar 0,965, MAP sebesar 0.980, MRR sebesar 1, dan nDCG sebesar 0,992. Nilai tersebut menunjukkan bahwa sistem tetap mampu menghasilkan rekomendasi yang relevan

terhadap *query* yang diberikan meskipun jumlah rekomendasi yang ditampilkan diperluas menjadi Top-10.

Nilai *Precision* yang mendekati 1 menunjukkan bahwa sebagian besar film yang direkomendasikan dinilai relevan oleh validator. Selain itu, nilai MAP yang tinggi menunjukkan bahwa film-film relevan cenderung menempati posisi atas dalam daftar rekomendasi. Nilai MRR sebesar 1 mengindikasikan bahwa film relevan pertama selalu muncul pada peringkat pertama hasil rekomendasi untuk seluruh *query* yang diuji. Sementara itu, nilai nDCG sebesar 0,992 menunjukkan bahwa film-film yang relevan cenderung berada pada posisi atas daftar rekomendasi yang dihasilkan sistem. Untuk memperjelas hasil evaluasi pada Skenario 2, visualisasi nilai *Precision*, MAP, MRR, dan nDCG ditunjukkan pada Gambar 4.2.



Gambar 4. 2 *Boxplot* Rata-rata Hasil Skenario Uji Coba 2

Berdasarkan Gambar 4.2, distribusi nilai *Precision*, MAP, MRR, dan nDCG pada Skenario 2 menunjukkan bahwa hasil evaluasi antar query cenderung konsisten. Hal ini terlihat dari posisi median seluruh metrik yang berada sangat dekat dengan nilai 1. Selain itu, nilai standar deviasi yang relatif kecil pada *Precision* $\sigma = 0.075$, MAP $\sigma = 0.049$, dan nDCG $\sigma = 0.022$ menunjukkan bahwa hasil evaluasi antar *query* cenderung konsisten. Sementara itu, nilai MRR memiliki standar deviasi sebesar 0 yang menunjukkan bahwa film relevan pertama selalu muncul pada peringkat teratas untuk seluruh *query* yang diuji.

4.1.3 Skenario Uji Coba 3

Skenario 3 merupakan pengujian yang menggunakan *preprocessing* pada *query* panjang dengan panjang lebih dari lima kata dan menghasilkan lima rekomendasi teratas Top-5. Pengujian ini dilakukan untuk mengetahui kemampuan sistem dalam menghasilkan rekomendasi film yang relevan ketika menggunakan *query* panjang dengan jumlah rekomendasi yang lebih banyak dibandingkan Skenario 1. Hasil evaluasi pada Skenario 3 ditunjukkan pada Tabel 4.3.

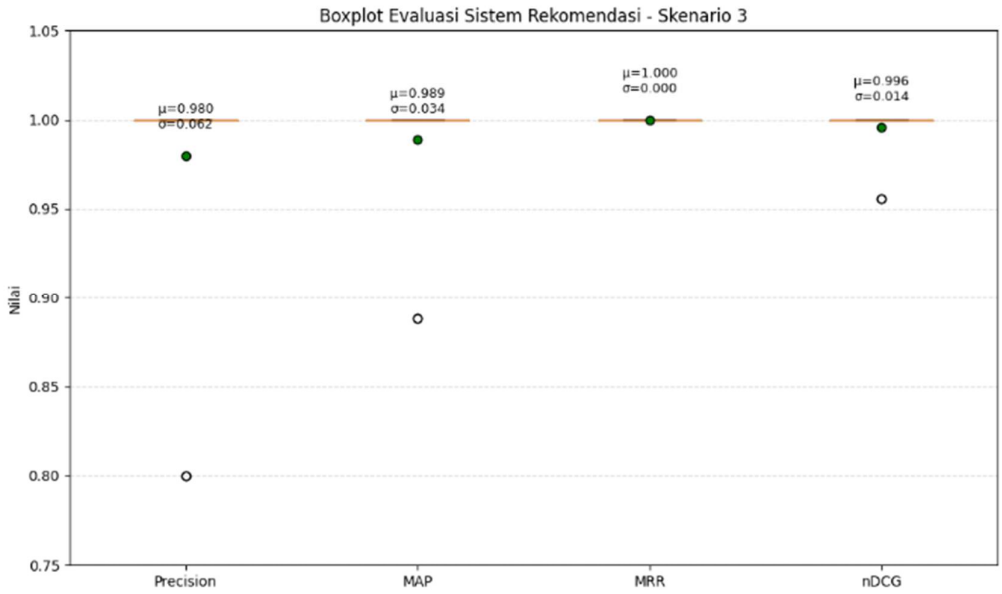
Tabel 4. 3 Hasil Skenario Uji Coba 3

Query	**Precision**	**MAP**	**MRR**	**nDCG**
Query 1	1	1	1	1
Query 2	1	1	1	1
Query 3	1	1	1	1
Query 4	1	1	1	1
Query 5	1	1	1	1
Query 6	1	1	1	1
Query 7	1	1	1	1
Query 8	1	1	1	1
Query 9	1	1	1	1
Query 10	1	1	1	1
Query 11	1	1	1	1
Query 12	1	1	1	1
Query 13	1	1	1	1
Query 14	0.8	0.888	1	0.956
Query 15	1	1	1	1
Query 16	1	1	1	1

Query 17	1	1	1	1
Query 18	0.8	0.888	1	0.956
Query 19	1	1	1	1
Query 20	1	1	1	1
Rata-rata	0.980	0.989	1	0.996
Standar Deviasi	0.062	0.034	0	0.014

Berdasarkan Tabel 4.3, Skenario 3 memperoleh nilai rata-rata *Precision* sebesar 0.980, MAP sebesar 0.989, MRR sebesar 1, dan nDCG sebesar 0.996. Hasil tersebut menunjukkan bahwa sistem mampu menghasilkan rekomendasi film yang relevan terhadap *query* panjang yang telah melalui proses *preprocessing* dengan jumlah rekomendasi sebanyak Top-5.

Nilai *Precision* sebesar 0.980 menunjukkan bahwa sebagian besar film yang direkomendasikan pada lima peringkat teratas dinilai relevan oleh validator. Selain itu, nilai MAP sebesar 0.989 mengindikasikan bahwa film-film yang relevan cenderung menempati posisi atas dalam daftar rekomendasi. Nilai MRR sebesar 1 menunjukkan bahwa film relevan pertama selalu muncul pada peringkat pertama untuk seluruh *query* yang diuji. Sementara itu, nilai nDCG sebesar 0.996 menunjukkan bahwa urutan rekomendasi yang dihasilkan sistem telah mendekati urutan ideal sehingga relevansi film yang direkomendasikan tetap terjaga pada posisi teratas. Untuk memperjelas hasil evaluasi pada Skenario 3, visualisasi distribusi nilai *Precision*, MAP, MRR, dan nDCG ditunjukkan pada Gambar 4.3.



Gambar 4. 3 *Boxplot* Rata-rata Hasil Skenario Uji Coba 3

Berdasarkan Gambar 4.3, distribusi nilai *Precision*, MAP, MRR, dan nDCG pada Skenario 3 menunjukkan performa sistem yang sangat stabil. Hal ini terlihat dari mayoritas nilai evaluasi yang terkonsentrasi pada nilai maksimum, yaitu 1. Hanya terdapat dua *query* yang menghasilkan nilai lebih rendah dibandingkan *query* lainnya, yaitu *Query* 14 dan *Query* 18, sehingga muncul sebagai variasi data pada *boxplot*.

Nilai standar deviasi pada *Precision* sebesar 0.062, MAP sebesar 0.034, dan nDCG sebesar 0.014 menunjukkan bahwa performa sistem antar *query* cenderung konsisten dan perbedaannya relatif kecil. Sementara itu, nilai MRR memiliki standar deviasi sebesar 0, yang menunjukkan bahwa film relevan pertama selalu muncul pada peringkat pertama untuk seluruh *query* yang diuji. Hasil ini mengindikasikan bahwa penggunaan *preprocessing* pada *query* panjang mampu

membantu sistem mempertahankan kualitas rekomendasi yang tinggi dan stabil pada lima rekomendasi teratas.

4.1.4 Skenario Uji Coba 4

Skenario 4 merupakan pengujian yang menggunakan *preprocessing* pada *query* panjang dengan panjang lebih dari lima kata dan menghasilkan sepuluh rekomendasi teratas Top-10. Pengujian ini dilakukan untuk mengetahui kemampuan sistem dalam menghasilkan rekomendasi film yang relevan ketika menggunakan *query* yang lebih deskriptif dengan jumlah rekomendasi yang lebih banyak. Hasil evaluasi pada Skenario 4 ditunjukkan pada Tabel 4.4.

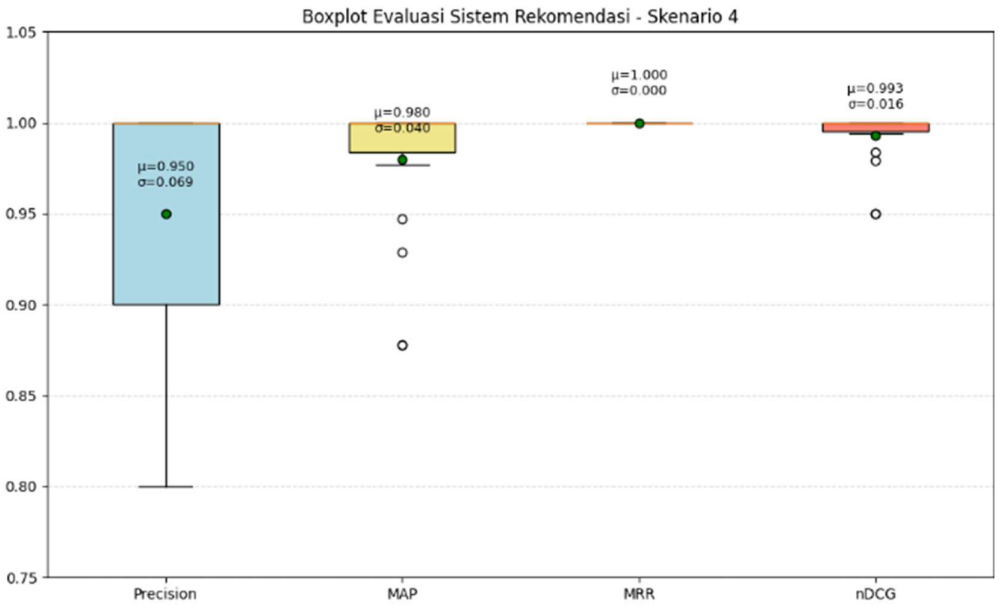
Tabel 4. 4 Hasil Skenario Uji Coba 4

Query	***Precision***	**MAP**	**MRR**	**nDCG**
Query 1	0.9	1	1	1
Query 2	0.8	0.986	1	0.996
Query 3	1	1	1	1
Query 4	1	1	1	1
Query 5	1	1	1	1
Query 6	1	1	1	1
Query 7	1	1	1	1
Query 8	1	1	1	1
Query 9	1	1	1	1
Query 10	0.8	0.929	1	0.979
Query 11	1	1	1	1
Query 12	1	1	1	1
Query 13	1	1	1	1
Query 14	0.9	0.878	1	0.95
Query 15	0.9	0.977	1	0.994
Query 16	0.9	1	1	1
Query 17	1	1	1	1
Query 18	0.9	0.878	1	0.950
Query 19	0.9	0.947	1	0.984
Query 20	1	1	1	1
Rata-rata	0.950	0.980	1	0.993
Standar Deviasi	0.069	0.040	0	0.016

Berdasarkan Tabel 4.4, Skenario 4 memperoleh nilai rata-rata *Precision* sebesar 0.95, MAP sebesar 0.980, MRR sebesar 1, dan nDCG sebesar 0.993. Nilai

tersebut menunjukkan bahwa sistem mampu menghasilkan rekomendasi yang relevan terhadap *query* yang diberikan meskipun jumlah rekomendasi yang ditampilkan diperluas menjadi Top-10.

Nilai *Precision* yang tinggi menunjukkan bahwa sebagian besar film yang direkomendasikan dinilai relevan oleh validator. Selain itu, nilai MAP yang mendekati 1 menunjukkan bahwa film-film relevan cenderung menempati posisi atas dalam daftar rekomendasi. Nilai MRR sebesar 1 mengindikasikan bahwa film relevan pertama selalu muncul pada peringkat pertama hasil rekomendasi untuk seluruh *query* yang diuji. Sementara itu, nilai nDCG sebesar 0.993 menunjukkan bahwa film-film yang relevan cenderung berada pada posisi atas daftar rekomendasi yang dihasilkan sistem. Untuk memperjelas hasil evaluasi pada Skenario 4, visualisasi nilai *Precision*, MAP, MRR, dan nDCG ditunjukkan pada Gambar 4.4.



Gambar 4. 4 *Boxplot* Rata-rata Hasil Skenario Uji Coba 4

Berdasarkan Gambar 4.4, distribusi nilai *Precision*, MAP, MRR, dan nDCG pada Skenario 4 menunjukkan bahwa hasil evaluasi antar query cenderung konsisten. Hal ini terlihat dari posisi median seluruh metrik yang berada sangat dekat dengan nilai maksimum, yaitu 1. Selain itu, nilai standar deviasi yang relatif kecil pada *Precision* sebesar 0.069, MAP sebesar 0.040, dan nDCG sebesar 0.016 menunjukkan bahwa hasil evaluasi antar *query* cenderung konsisten. Sementara itu, nilai MRR memiliki standar deviasi sebesar 0 yang menunjukkan bahwa film relevan pertama selalu muncul pada peringkat teratas untuk seluruh *query* yang diuji.

4.1.5 Skenario Uji Coba 5

Skenario 5 merupakan pengujian yang tidak menggunakan *preprocessing* pada *query* pendek dengan panjang kurang dari lima kata dan menghasilkan lima rekomendasi teratas Top-5. Pengujian ini dilakukan untuk mengetahui kemampuan sistem dalam menghasilkan rekomendasi film yang relevan ketika *query* diproses secara langsung tanpa melalui tahapan *preprocessing*. Hasil evaluasi pada Skenario 5 ditunjukkan pada Tabel 4.5.

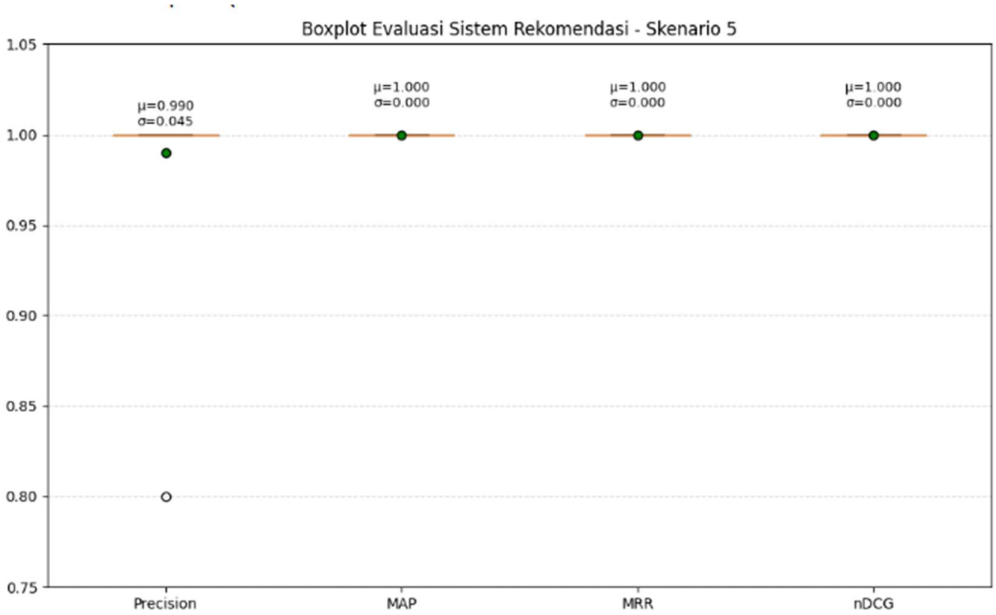
Tabel 4. 5 Hasil Skenario Uji Coba 5

Query	***Precision***	**MAP**	**MRR**	**nDCG**
Query 1	1	1	1	1
Query 2	1	1	1	1
Query 3	1	1	1	1
Query 4	1	1	1	1
Query 5	1	1	1	1
Query 6	0.8	1	1	1
Query 7	1	1	1	1
Query 8	1	1	1	1
Query 9	1	1	1	1
Query 10	1	1	1	1
Query 11	1	1	1	1

Query 12	1	1	1	1
Query 13	1	1	1	1
Query 14	1	1	1	1
Query 15	1	1	1	1
Query 16	1	1	1	1
Query 17	1	1	1	1
Query 18	1	1	1	1
Query 19	1	1	1	1
Query 20	1	1	1	1
Rata-rata	0.990	1	1	1
Standar Deviasi	0.045	0	0	0

Berdasarkan Tabel 4.5, Skenario 5 memperoleh nilai *Precision* sebesar 0,990, MAP sebesar 1, MRR sebesar 1, dan nDCG sebesar 1. Nilai tersebut menunjukkan bahwa sistem mampu menghasilkan rekomendasi yang relevan terhadap *query* yang diberikan meskipun tidak menggunakan *preprocessing*.

Nilai *Precision* yang mendekati 1 menunjukkan bahwa hampir seluruh film yang direkomendasikan pada Top-5 dinilai relevan oleh validator. Selain itu, nilai MAP sebesar 1 menunjukkan bahwa seluruh film relevan berhasil ditempatkan pada posisi yang relevan dalam daftar rekomendasi. Nilai MRR sebesar 1 mengindikasikan bahwa film relevan pertama selalu muncul pada peringkat pertama hasil rekomendasi. Sementara itu, nilai nDCG sebesar 1 menunjukkan bahwa urutan rekomendasi yang dihasilkan sistem telah sesuai dengan urutan ideal. Untuk memperjelas hasil evaluasi pada Skenario 5, visualisasi nilai *Precision*, MAP, MRR, dan nDCG ditunjukkan pada Gambar 4.5.



Gambar 4. 5 *Boxplot* Rata-rata Hasil Skenario Uji Coba 5

Berdasarkan Gambar 4.5, distribusi nilai *Precision*, MAP, MRR, dan nDCG pada Skenario 5 menunjukkan performa sistem yang sangat stabil. Hal ini terlihat dari posisi median seluruh metrik yang berada pada nilai maksimum, yaitu 1. Selain itu, nilai standar deviasi pada MAP, MRR, dan nDCG sebesar 0 menunjukkan bahwa seluruh *query* menghasilkan nilai evaluasi yang identik pada ketiga metrik tersebut. Sementara itu, metrik *Precision* memiliki nilai rata-rata sebesar 0.99 dengan standar deviasi sebesar 0.045 yang menunjukkan adanya variasi hasil pada satu *query*, yaitu *Query* 6, sedangkan *query* lainnya menghasilkan nilai *Precision* sempurna sebesar 1.

4.1.6 Skenario Uji Coba 6

Skenario 6 merupakan pengujian yang tidak menggunakan *preprocessing* pada *query* pendek dengan panjang kurang dari lima kata dan menghasilkan sepuluh

rekomendasi teratas Top-10. Pengujian ini dilakukan untuk mengetahui kemampuan sistem dalam menghasilkan rekomendasi film yang relevan ketika *query* diproses secara langsung tanpa melalui tahapan *preprocessing* dan jumlah rekomendasi yang ditampilkan diperluas menjadi Top-10. Hasil evaluasi pada Skenario 6 ditunjukkan pada Tabel 4.6.

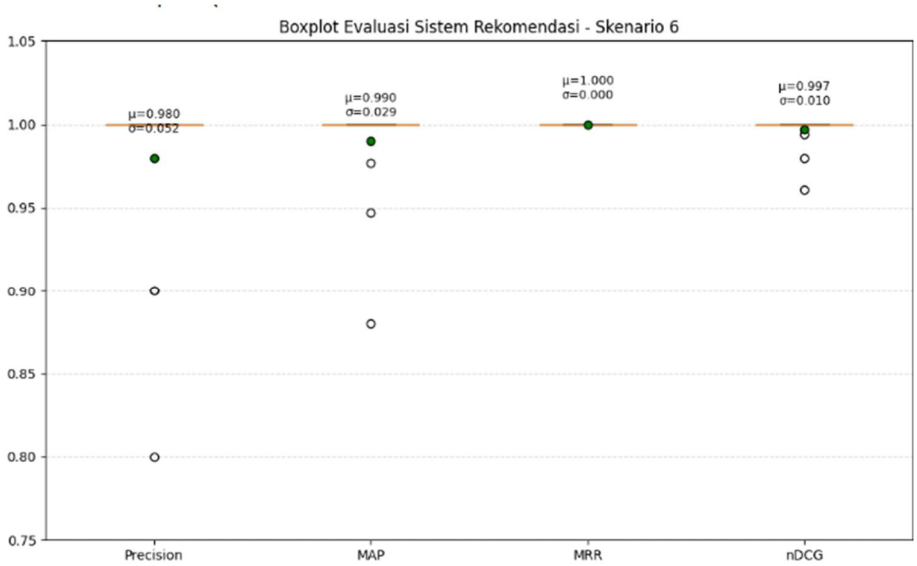
Tabel 4. 6 Hasil Skenario Uji Coba 6

Query	***Precision***	**MAP**	**MRR**	**nDCG**
Query 1	1	1	1	1
Query 2	1	1	1	1
Query 3	1	1	1	1
Query 4	1	1	1	1
Query 5	1	1	1	1
Query 6	0.8	0.880	1	0.961
Query 7	1	1	1	1
Query 8	1	1	1	1
Query 9	1	1	1	1
Query 10	1	1	1	1
Query 11	1	1	1	1
Query 12	1	1	1	1
Query 13	1	1	1	1
Query 14	1	1	1	1
Query 15	0.9	0.947	1	0.98
Query 16	1	1	1	1
Query 17	1	1	1	1
Query 18	0.9	0.977	1	0.994
Query 19	1	1	1	1
Query 20	1	1	1	1
Rata-rata	0.980	0.990	1	0.997
Standar Deviasi	0.052	0.029	0	0.010

Berdasarkan Tabel 4.6, Skenario 6 memperoleh nilai *Precision* sebesar 0,980, MAP sebesar 0,990, MRR sebesar 1, dan nDCG sebesar 0,997. Nilai tersebut menunjukkan bahwa sistem mampu menghasilkan rekomendasi yang relevan terhadap *query* yang diberikan meskipun tanpa menggunakan *preprocessing* dan menampilkan jumlah rekomendasi yang lebih banyak.

Nilai *Precision* yang mendekati 1 menunjukkan bahwa sebagian besar film yang direkomendasikan pada Top-10 dinilai relevan oleh validator. Selain itu, nilai

MAP yang tinggi menunjukkan bahwa film-film relevan cenderung berada pada posisi atas dalam daftar rekomendasi. Nilai MRR sebesar 1 mengindikasikan bahwa film relevan pertama selalu muncul pada peringkat pertama hasil rekomendasi. Sementara itu, nilai nDCG sebesar 0.997 menunjukkan bahwa film-film yang relevan cenderung berada pada posisi atas daftar rekomendasi yang dihasilkan sistem. Untuk memperjelas hasil evaluasi pada Skenario 6, visualisasi nilai *Precision*, MAP, MRR, dan nDCG ditunjukkan pada Gambar 4.6.



Gambar 4. 6 *Boxplot* Rata-rata Hasil Skenario Uji Coba 6

Berdasarkan Gambar 4.6, distribusi nilai *Precision*, MAP, MRR, dan nDCG pada Skenario 6 menunjukkan performa sistem yang sangat stabil. Hal ini terlihat dari posisi median seluruh metrik yang berada sangat dekat dengan nilai maksimum, yaitu 1. Selain itu, nilai standar deviasi yang relatif kecil pada *Precision* sebesar 0.052, MAP sebesar 0.029, dan nDCG sebesar 0.010 menunjukkan bahwa hasil evaluasi antar *query* cenderung konsisten. Sementara itu, nilai MRR memiliki

standar deviasi sebesar 0 yang menunjukkan bahwa film relevan pertama selalu muncul pada peringkat teratas untuk seluruh *query* yang diuji. Meskipun terdapat beberapa *query* yang menghasilkan nilai lebih rendah, yaitu *Query* 6, *Query* 15, dan *Query* 18, mayoritas *query* tetap memperoleh nilai evaluasi sempurna sehingga distribusi data terkonsentrasi pada nilai maksimum.

4.1.7 Skenario Uji Coba 7

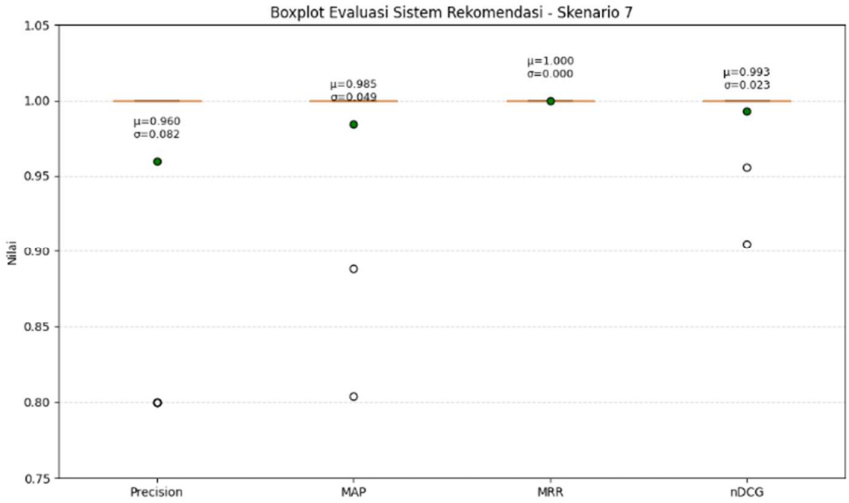
Skenario 7 merupakan pengujian yang tidak menggunakan *preprocessing* pada *query* panjang dengan panjang lebih dari lima kata dan menghasilkan lima rekomendasi teratas Top-5. Pengujian ini dilakukan untuk mengetahui kemampuan sistem dalam menghasilkan rekomendasi film yang relevan ketika *query* diproses secara langsung tanpa melalui tahapan *preprocessing* dan menggunakan *query* yang lebih deskriptif. Hasil evaluasi pada Skenario 7 ditunjukkan pada Tabel 4.7.

Tabel 4. 7 Hasil Skenario Uji Coba 7

Query	***Precision***	**MAP**	**MRR**	**nDCG**
Query 1	0.8	1	1	1
Query 2	1	1	1	1
Query 3	1	1	1	1
Query 4	1	1	1	1
Query 5	1	1	1	1
Query 6	1	1	1	1
Query 7	1	1	1	1
Query 8	1	1	1	1
Query 9	1	1	1	1
Query 10	1	1	1	1
Query 11	1	1	1	1
Query 12	1	1	1	1
Query 13	1	1	1	1
Query 14	1	1	1	1
Query 15	0.8	0.804	1	0.905
Query 16	1	1	1	1
Query 17	1	1	1	1
Query 18	0.8	0.888	1	0.956
Query 19	1	1	1	1
Query 20	0.8	1	1	1
Rata-rata	0.960	0.985	1	0.993
Standar Deviasi	0.082	0.049	0	0.023

Berdasarkan Tabel 4.7, Skenario 7 memperoleh nilai *Precision* sebesar 0.960, MAP sebesar 0.985, MRR sebesar 1, dan nDCG sebesar 0.993. Nilai tersebut menunjukkan bahwa sistem mampu menghasilkan rekomendasi yang relevan terhadap *query* yang diberikan meskipun tanpa menggunakan *preprocessing* pada *query* panjang.

Nilai *Precision* yang mendekati 1 menunjukkan bahwa sebagian besar film yang direkomendasikan pada Top-5 dinilai relevan oleh validator. Selain itu, nilai MAP yang tinggi menunjukkan bahwa film-film relevan cenderung berada pada posisi atas dalam daftar rekomendasi. Nilai MRR sebesar 1 mengindikasikan bahwa film relevan pertama selalu muncul pada peringkat pertama hasil rekomendasi. Sementara itu, nilai nDCG sebesar 0.993 menunjukkan bahwa film-film yang relevan cenderung berada pada posisi atas daftar rekomendasi yang dihasilkan sistem. Untuk memperjelas hasil evaluasi pada Skenario 7, visualisasi nilai *Precision*, MAP, MRR, dan nDCG ditunjukkan pada Gambar 4.7.



Gambar 4. 7 *Boxplot* Rata-rata Hasil Skenario Uji Coba 7

Berdasarkan Gambar 4.7, distribusi nilai *Precision*, MAP, MRR, dan nDCG pada Skenario 7 menunjukkan bahwa hasil evaluasi antar query cenderung konsisten. Hal ini terlihat dari posisi median seluruh metrik yang berada sangat dekat dengan nilai maksimum, yaitu 1. Selain itu, nilai standar deviasi yang relatif kecil pada *Precision* sebesar 0.082, MAP sebesar 0.049, dan nDCG sebesar 0.023 menunjukkan bahwa hasil evaluasi antar query cenderung konsisten. Sementara itu, nilai MRR memiliki standar deviasi sebesar 0 yang menunjukkan bahwa film relevan pertama selalu muncul pada peringkat teratas untuk seluruh *query* yang diuji.

4.1.8 Skenario Uji Coba 8

Skenario 8 merupakan pengujian yang tidak menggunakan *preprocessing* pada *query* panjang dengan panjang lebih dari lima kata dan menghasilkan sepuluh rekomendasi teratas (Top-10). Pengujian ini dilakukan untuk mengetahui kemampuan sistem dalam menghasilkan rekomendasi film yang relevan ketika *query* diproses secara langsung tanpa melalui tahapan *preprocessing* dan jumlah rekomendasi yang ditampilkan diperluas menjadi Top-10. Hasil evaluasi pada Skenario 8 ditunjukkan pada Tabel 4.8.

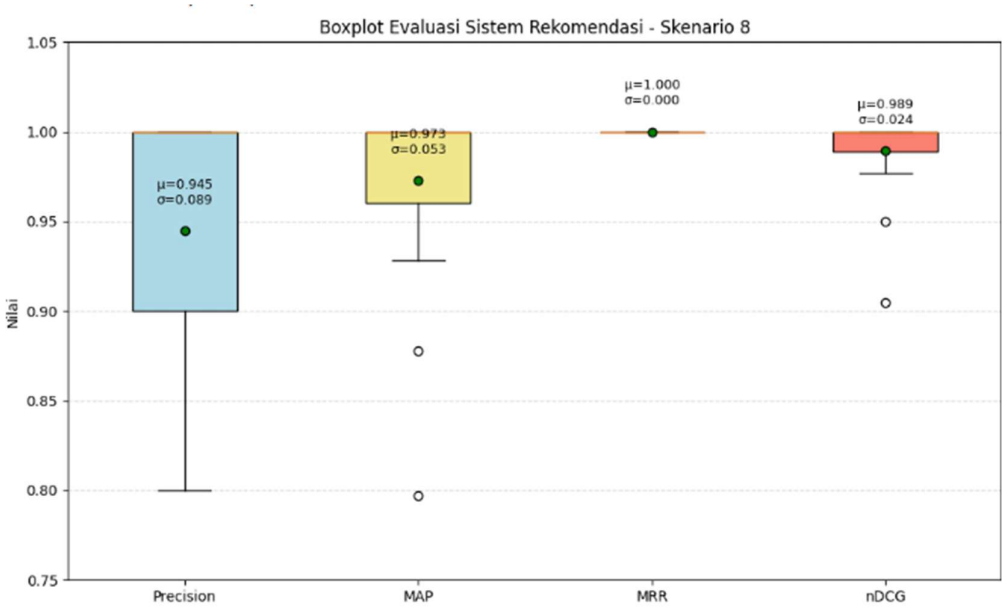
Tabel 4. 8 Hasil Skenario Uji Coba 8

Query	***Precision***	**MAP**	**MRR**	**nDCG**
Query 1	0.7	0.938	1	0.980
Query 2	0.9	0.963	1	0.990
Query 3	1	1	1	1
Query 4	1	1	1	1
Query 5	1	1	1	1
Query 6	0.8	0.953	1	0.986
Query 7	1	1	1	1
Query 8	1	1	1	1
Query 9	1	1	1	1

Query 10	0.9	1	1	1
Query 11	1	1	1	1
Query 12	1	1	1	1
Query 13	1	1	1	1
Query 14	1	1	1	1
Query 15	0.8	0.797	1	0.905
Query 16	1	1	1	1
Query 17	1	1	1	1
Query 18	0.9	0.878	1	0.950
Query 19	1	1	1	1
Query 20	0.9	0.928	1	0.977
Rata-rata	0.945	0.973	1	0.989
Standar Deviasi	0.089	0.053	0	0.0234

Berdasarkan Tabel 4.8, Skenario 8 memperoleh nilai *Precision* sebesar 0.945, MAP sebesar 0.973, MRR sebesar 1, dan nDCG sebesar 0.989. Nilai tersebut menunjukkan bahwa sistem mampu menghasilkan rekomendasi yang relevan terhadap *query* yang diberikan meskipun tanpa menggunakan *preprocessing* pada *query* panjang dan dengan jumlah rekomendasi yang lebih banyak.

Nilai *Precision* yang mendekati 1 menunjukkan bahwa sebagian besar film yang direkomendasikan pada Top-10 dinilai relevan oleh validator. Selain itu, nilai MAP yang tinggi menunjukkan bahwa film-film relevan cenderung berada pada posisi atas dalam daftar rekomendasi. Nilai MRR sebesar 1 mengindikasikan bahwa film relevan pertama selalu muncul pada peringkat pertama hasil rekomendasi. Sementara itu, nilai nDCG sebesar 0.989 menunjukkan bahwa film-film yang relevan cenderung berada pada posisi atas daftar rekomendasi yang dihasilkan sistem. Untuk memperjelas hasil evaluasi pada Skenario 8, visualisasi nilai *Precision*, MAP, MRR, dan nDCG ditunjukkan pada Gambar 4.8.



Gambar 4. 8 *Boxplot* Rata-rata Hasil Skenario Uji Coba 8

Berdasarkan Gambar 4.8, distribusi nilai *Precision*, MAP, MRR, dan nDCG pada Skenario 8 menunjukkan bahwa hasil evaluasi antar *query* cenderung konsisten. Hal ini terlihat dari posisi median seluruh metrik yang berada sangat dekat dengan nilai maksimum, yaitu 1. Selain itu, nilai standar deviasi yang relatif kecil pada *Precision* sebesar 0.0889, MAP sebesar 0.053, dan nDCG sebesar 0.024 menunjukkan bahwa hasil evaluasi antar query cenderung konsisten. Sementara itu, nilai MRR memiliki standar deviasi sebesar 0 yang menunjukkan bahwa film relevan pertama selalu muncul pada peringkat teratas untuk seluruh *query* yang diuji. Meskipun terdapat beberapa query yang menghasilkan nilai lebih rendah dibandingkan query lainnya, mayoritas query tetap memperoleh nilai evaluasi mendekati 1 sehingga distribusi data masih terkonsentrasi pada nilai maksimum.

4.2 Pembahasan

Pada subbab ini dilakukan analisis terhadap hasil pengujian yang telah diperoleh pada seluruh skenario uji. Pembahasan difokuskan pada interpretasi nilai *Precision*, *Mean Average Precision* (MAP), *Mean Reciprocal Rank* (MRR), dan *Normalized Discounted Cumulative Gain* (nDCG) untuk mengetahui sejauh mana sistem rekomendasi film mampu menghasilkan rekomendasi yang relevan dan terurut dengan baik. Analisis dilakukan dengan membandingkan hasil pengujian berdasarkan perbedaan perlakuan yang diterapkan, yaitu penggunaan *preprocessing* dan non-*preprocessing*, variasi panjang *query*, serta jumlah rekomendasi yang ditampilkan pada Top-5 dan Top-10. Melalui perbandingan tersebut, dapat diketahui faktor-faktor yang mempengaruhi performa sistem serta kondisi yang menghasilkan kinerja rekomendasi terbaik.

4.2.1 Analisis Pengaruh *Preprocessing* terhadap Kualitas Rekomendasi

Analisis pengaruh *preprocessing* dilakukan untuk mengetahui sejauh mana tahapan *preprocessing* berkontribusi terhadap kualitas rekomendasi yang dihasilkan oleh sistem. Pada penelitian ini, *preprocessing* terdiri atas proses tokenisasi*, case folding, lemmatization, symbol and punctuation remova*l, serta *stopword removal* yang diterapkan pada *query* pengguna sebelum dilakukan proses *embedding* menggunakan model SBERT. Pengujian dilakukan dengan membandingkan Skenario 1-4 yang menggunakan *preprocessing* dengan Skenario 5-8 yang tidak menggunakan *preprocessing*. Perbandingan dilakukan berdasarkan nilai *Precision*, *Mean Average Precision* (MAP), *Mean Reciprocal Rank* (MRR), dan *Normalized*

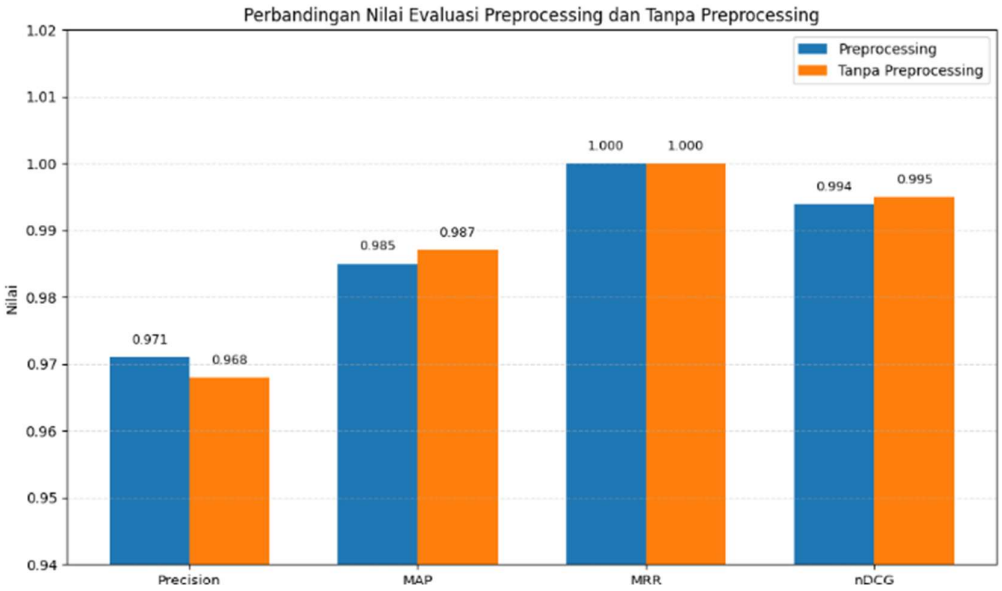
Discounted Cumulative Gain (nDCG) yang diperoleh pada masing-masing skenario.

Untuk mempermudah analisis, nilai rata-rata dari seluruh skenario yang menggunakan *preprocessing* dibandingkan dengan nilai rata-rata dari seluruh skenario tanpa *preprocessing*. Hasil perbandingan tersebut disajikan pada Tabel 4.9.

Tabel 4. 9 Perbandingan Nilai Rata-rata Pengaruh *Preprocessing*

Metrik	***Preprocessing***	**Tanpa *Preprocessing***
Precision	0.971	0.968
MAP	0.985	0.987
MRR	1	1
nDCG	0.994	0.995

Untuk memperjelas perbandingan performa sistem pada kedua kelompok pengujian, hasil evaluasi divisualisasikan dalam bentuk grafik pada Gambar 4.9.



Gambar 4. 9 Grafik Perbandingan Pengaruh *Preprocessing*

Berdasarkan Tabel 4.9 dan Gambar 4.9, kelompok yang menggunakan *preprocessing* memperoleh nilai *Precision* sebesar 0.971, sedangkan kelompok

tanpa *preprocessing* memperoleh nilai sebesar 0.968. Pada metrik MAP, masing-masing memperoleh nilai 0.985 dan 0.987. Sementara itu, nilai MRR pada kedua kelompok sama-sama sebesar 1, sedangkan nilai nDCG masing-masing sebesar 0.994 dan 0.995.

Hasil tersebut menunjukkan bahwa kedua kelompok menghasilkan performa yang relatif berdekatan pada seluruh metrik evaluasi. Kelompok *preprocessing* memperoleh nilai *Precision* sedikit lebih tinggi dibandingkan kelompok tanpa *preprocessing*, sedangkan kelompok tanpa *preprocessing* memperoleh nilai MAP dan nDCG yang sedikit lebih tinggi. Sementara itu, nilai MRR yang sama menunjukkan bahwa item relevan pertama selalu muncul pada posisi teratas hasil rekomendasi pada kedua kelompok pengujian.

Performa yang relatif serupa pada kedua kelompok menunjukkan bahwa model SBERT mampu memahami hubungan semantik antara *query* dan sinopsis film dengan baik, baik pada data yang telah melalui *preprocessing* maupun data tanpa *preprocessing*. Kemampuan ini didukung oleh arsitektur Transformer yang memungkinkan model menangkap konteks dan makna kalimat secara efektif sehingga representasi teks yang dihasilkan tetap berkualitas.

Secara keseluruhan, penggunaan *preprocessing* memberikan pengaruh terhadap kualitas rekomendasi yang dihasilkan. Namun, *perbedaan* performa yang diperoleh antara kelompok *preprocessing* dan tanpa *preprocessing* relatif kecil pada seluruh metrik evaluasi. Hasil ini menunjukkan bahwa sistem tetap mampu menghasilkan rekomendasi yang relevan dan memiliki kualitas peringkat yang baik baik dengan maupun tanpa *preprocessing*.

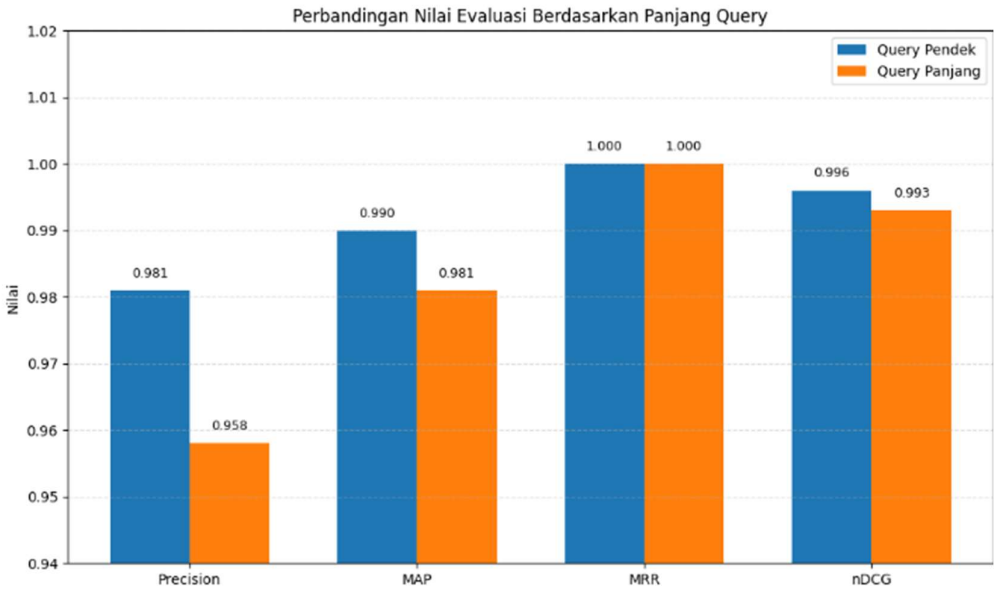
4.2.2 Analisis Pengaruh Panjang *Query* terhadap Kualitas Rekomendasi

Analisis pengaruh panjang *query* dilakukan untuk mengetahui sejauh mana panjang *query* mempengaruhi kualitas rekomendasi yang dihasilkan sistem. Pada penelitian ini, *query* dikelompokkan menjadi dua kategori, yaitu *query* pendek yang terdiri dari kurang dari lima kata dan *query* panjang yang terdiri dari lebih dari lima kata. Perbandingan dilakukan berdasarkan nilai rata-rata *Precision*, *Mean Average Precision* (MAP), *Mean Reciprocal Rank* (MRR), dan *Normalized Discounted Cumulative Gain* (nDCG). Hasil perbandingan tersebut disajikan pada Tabel 4.10.

Tabel 4. 10 Perbandingan Nilai Rata-rata Pengaruh Panjang *Query*

Metrik	***Query* Pendek**	***Query* Panjang**
Precision	0.981	0.958
MAP	0.990	0.981
MRR	1	1
nDCG	0.996	0.993

Untuk memperjelas perbandingan performa sistem pada kedua kelompok pengujian, hasil evaluasi divisualisasikan dalam bentuk grafik pada Gambar 4.10.



Gambar 4. 10 Grafik Perbandingan Pengaruh Panjang *Query*

Berdasarkan Tabel 4.10 dan Gambar 4.10, kelompok *query* pendek memperoleh nilai *Precision* sebesar 0,981, sedangkan kelompok *query* panjang memperoleh nilai sebesar 0,958. Pada metrik MAP, *query* pendek memperoleh nilai sebesar 0,990, lebih tinggi dibandingkan *query* panjang sebesar 0,981. Hasil serupa juga terlihat pada metrik nDCG, di mana query pendek memperoleh nilai sebesar 0,996, sedangkan *query* panjang memperoleh nilai sebesar 0,993. Sementara itu, nilai MRR pada kedua kelompok sama-sama sebesar 1.

Hasil tersebut menunjukkan bahwa *query* pendek cenderung menghasilkan rekomendasi yang lebih relevan dan memiliki kualitas peringkat yang sedikit lebih baik dibandingkan *query* panjang. Kondisi ini dapat terjadi karena *query* pendek umumnya berisi kata kunci yang lebih fokus dan spesifik sehingga lebih mudah direpresentasikan secara semantik oleh model SBERT dan dicocokkan dengan sinopsis film yang relevan. Sebaliknya, *query* panjang mengandung lebih banyak informasi dan variasi konteks yang dapat memperluas ruang pencarian sehingga proses pencocokan semantik menjadi lebih kompleks.

Meskipun demikian, perbedaan nilai evaluasi yang diperoleh antara *query* pendek dan *query* panjang relatif kecil. Selain itu, seluruh metrik evaluasi pada kedua kelompok menunjukkan nilai yang tinggi, dengan MRR mencapai 1 pada kedua kelompok. Hasil ini menunjukkan bahwa model SBERT mampu memahami makna semantik *query* dengan baik, baik pada *query* pendek maupun *query* panjang.

Secara keseluruhan, panjang query memberikan pengaruh terhadap kualitas rekomendasi yang dihasilkan. *Query* pendek cenderung menghasilkan nilai evaluasi yang lebih tinggi dibandingkan *query* panjang, meskipun perbedaan yang diperoleh relatif kecil pada seluruh metrik evaluasi. Nilai rata-rata pada masing-masing kelompok diperoleh dari agregasi seluruh skenario yang menggunakan query pendek maupun query panjang, sehingga hasil yang diperoleh merepresentasikan pengaruh panjang query secara keseluruhan terhadap kualitas rekomendasi sistem.

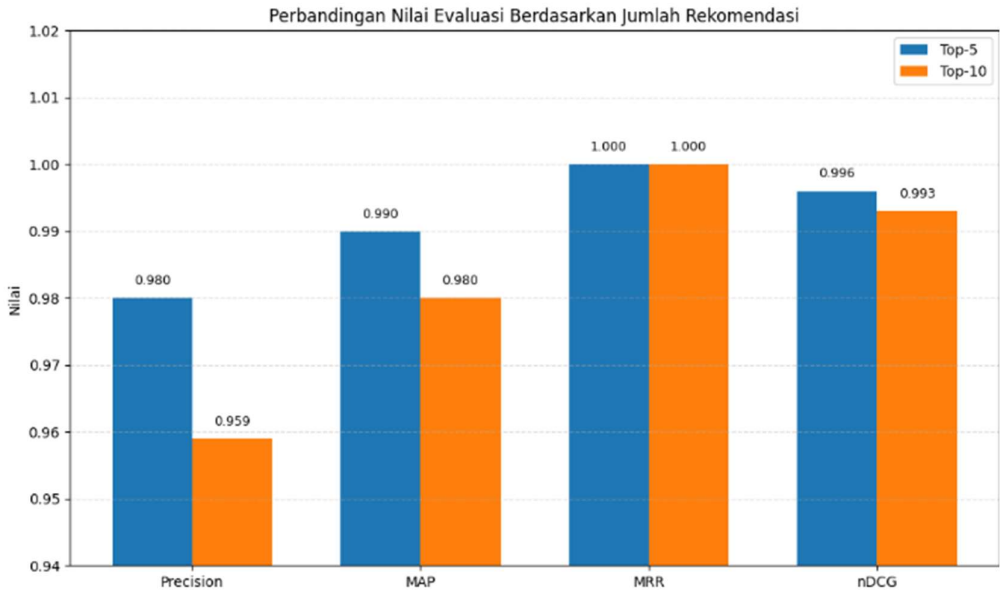
4.2.3 Analisis Pengaruh Jumlah Rekomendasi Top-5 dan Top-10

Analisis pengaruh jumlah rekomendasi dilakukan untuk mengetahui sejauh mana banyaknya film yang direkomendasikan mempengaruhi kualitas hasil rekomendasi yang dihasilkan sistem. Pada penelitian ini, jumlah rekomendasi yang dibandingkan terdiri atas Top-5 dan Top-10. Perbandingan dilakukan berdasarkan nilai rata-rata *Precision*, *Mean Average Precision* (MAP), *Mean Reciprocal Rank* (MRR), dan *Normalized Discounted Cumulative Gain* (nDCG). Hasil perbandingan tersebut disajikan pada Tabel 4.11.

Tabel 4. 11 Perbandingan Hasil Pengujian Berdasarkan Jumlah Rekomendasi

Metrik	Top-5	Top-10
Precision	0.980	0.959
MAP	0.990	0.980
MRR	1	1
nDCG	0.996	0.993

Untuk memperjelas perbandingan performa sistem pada kedua kelompok pengujian, hasil evaluasi divisualisasikan dalam bentuk grafik pada Gambar 4.11.



Gambar 4.11 Perbandingan Hasil Pengujian Berdasarkan Jumlah Rekomendasi

Berdasarkan Tabel 4.11 dan Gambar 4.11, kelompok Top-5 memperoleh nilai *Precision* sebesar 0,980, sedangkan kelompok Top-10 memperoleh nilai sebesar 0,959. Pada metrik MAP, kelompok Top-5 memperoleh nilai sebesar 0,990, lebih tinggi dibandingkan kelompok Top-10 sebesar 0,980. Hasil serupa juga terlihat pada metrik nDCG, di mana kelompok Top-5 memperoleh nilai sebesar 0,996, sedangkan kelompok Top-10 memperoleh nilai sebesar 0,993. Sementara itu, nilai MRR pada kedua kelompok menunjukkan hasil yang sama, yaitu sebesar 1.

Hasil tersebut menunjukkan bahwa jumlah rekomendasi yang lebih sedikit cenderung menghasilkan kualitas rekomendasi yang lebih baik. Ketika sistem hanya menampilkan lima rekomendasi teratas, film-film yang direkomendasikan memiliki tingkat kemiripan semantik yang lebih tinggi terhadap *query* pengguna. Sebaliknya, ketika jumlah rekomendasi diperluas menjadi sepuluh film, sistem

mulai menampilkan film dengan tingkat kemiripan yang lebih rendah sehingga menyebabkan nilai *Precision*, MAP, dan nDCG mengalami penurunan.

Meskipun demikian, perbedaan nilai evaluasi yang diperoleh antara kelompok Top-5 dan Top-10 relatif kecil. Selain itu, nilai MRR yang sama pada kedua kelompok menunjukkan bahwa film relevan pertama tetap muncul pada posisi teratas daftar rekomendasi baik pada Top-5 maupun Top-10. Hasil ini menunjukkan bahwa penambahan jumlah rekomendasi tidak mempengaruhi kemampuan sistem dalam menemukan rekomendasi relevan pertama.

Secara keseluruhan, jumlah rekomendasi mempengaruhi kualitas hasil rekomendasi yang dihasilkan sistem. Kelompok Top-5 menunjukkan performa yang sedikit lebih baik dibandingkan Top-10 pada metrik *Precision*, MAP, dan nDCG. Namun, sistem tetap mampu menghasilkan rekomendasi yang relevan dan memiliki kualitas peringkat yang baik pada kedua jumlah rekomendasi yang diuji. Hasil ini menunjukkan bahwa film-film yang berada pada peringkat teratas memiliki tingkat kemiripan yang lebih tinggi dengan query pengguna dibandingkan film-film pada peringkat berikutnya. Oleh karena itu, ketika jumlah rekomendasi dibatasi pada Top-5, proporsi item relevan menjadi lebih tinggi. Sebaliknya, pada Top-10 sistem mulai memasukkan film dengan tingkat kemiripan yang lebih rendah sehingga menyebabkan penurunan nilai evaluasi. Temuan ini menjelaskan mengapa kelompok Top-5 menghasilkan performa yang lebih baik dibandingkan Top-10 pada penelitian ini.

4.2.4 Analisis Hasil Pengujian Secara Keseluruhan

Berdasarkan seluruh hasil pengujian yang telah dilakukan pada delapan skenario, sistem rekomendasi film berbasis SBERT dan *Cosine Similarity* mampu menghasilkan rekomendasi yang memiliki tingkat relevansi yang tinggi terhadap *query* pengguna. Hal ini ditunjukkan oleh nilai *Precision*, MAP, MRR, dan nDCG yang secara umum berada pada rentang nilai di atas 0.94 pada seluruh skenario pengujian. Hasil tersebut menunjukkan bahwa sistem mampu menghasilkan rekomendasi yang relevan serta menempatkan item relevan pada posisi yang tepat dalam daftar rekomendasi.

Berdasarkan analisis pengaruh *preprocessing*, diperoleh bahwa kelompok *preprocessing* dan non-*preprocessing* menghasilkan performa yang relatif berdekatan pada seluruh metrik evaluasi. Kelompok *preprocessing* memperoleh nilai *Precision* sebesar 0,971, sedangkan kelompok tanpa *preprocessing* memperoleh nilai *Precision* sebesar 0,968. Pada metrik MAP dan nDCG, kelompok tanpa *preprocessing* memperoleh nilai yang sedikit lebih tinggi, yaitu 0,987 dan 0,995, dibandingkan preprocessing yang memperoleh nilai 0,985 dan 0,994. Sementara itu, nilai MRR pada kedua kelompok sama-sama mencapai 1. Selain itu, hasil analisis panjang *query* menunjukkan bahwa *query* pendek memperoleh nilai *Precision* sebesar 0,981, MAP sebesar 0,990, dan nDCG sebesar 0,996, sedangkan *query* panjang memperoleh nilai *Precision* sebesar 0,958, MAP sebesar 0,981, dan nDCG sebesar 0,993. Nilai MRR pada kedua kelompok sama-sama mencapai 1. Sementara itu, pada analisis jumlah rekomendasi, kelompok Top-5 memperoleh nilai evaluasi yang lebih tinggi dibandingkan kelompok Top-10. Hasil pengujian

menunjukkan bahwa sistem menghasilkan rekomendasi dengan tingkat relevansi tinggi berdasarkan penilaian satu validator pada skenario uji yang digunakan. Namun, hasil ini masih terbatas pada dataset, *query*, dan skema validasi yang digunakan dalam penelitian

Berdasarkan tujuan penelitian, sistem ini dirancang untuk memberikan rekomendasi film secara efektif dan relevan melalui pemahaman makna semantik antara *query* pengguna dan sinopsis film. Hasil pengujian menunjukkan bahwa tujuan tersebut telah tercapai, di mana sistem mampu menghasilkan rekomendasi yang relevan pada berbagai kondisi pengujian, baik menggunakan *preprocessing* maupun tanpa *preprocessing*, pada *query* pendek maupun *query* panjang, serta pada jumlah rekomendasi Top-5 maupun Top-10.

Tujuan tersebut sejalan dengan firman Allah Swt. dalam *QS. Az-Zumar* ayat 18:

الَّذِيْنَ يَسْتَمِعُوْنَ الْقَوْلَ فَيَتَّبِعُوْنَ اَحْسَنَهٗۗ اُولٰۤىِٕكَ الَّذِيْنَ هَدٰىهُمُ اللّٰهُ وَاُولٰۤىِٕكَ هُمْ اُولُوا الْاَلْبَابِ

"(Yaitu) orang-orang yang mendengarkan perkataan lalu mengikuti apa yang paling baik di antaranya. Mereka itulah orang-orang yang telah diberi petunjuk oleh Allah dan mereka itulah orang-orang yang mempunyai akal sehat." (QS. Az-Zumar: 18).

Menurut tafsir Ibnu Katsir, ayat ini menjelaskan bahwa orang-orang yang memperoleh petunjuk adalah mereka yang mampu memilah, memahami, dan memilih sesuatu yang terbaik dari berbagai informasi yang diterima. Manusia diperintahkan untuk menggunakan akal dan pertimbangan secara bijak agar dapat menentukan pilihan yang paling baik dan bermanfaat (Al-Sheikh, 1994). Dalam konteks penelitian ini, sistem rekomendasi film membantu pengguna memilih film

yang paling relevan dari banyaknya pilihan film yang tersedia. Melalui proses *semantic similarity*, sistem melakukan analisis terhadap makna *query* dan sinopsis film sehingga rekomendasi yang diberikan lebih sesuai dengan kebutuhan pengguna. Dengan demikian, penelitian ini sejalan dengan nilai Islam yang menekankan pentingnya memilih informasi atau alternatif terbaik berdasarkan pemahaman dan pertimbangan yang tepat.

Berdasarkan hasil pengujian, skenario terbaik diperoleh pada Skenario 5 dengan nilai *Precision* sebesar 0.990, MAP sebesar 1, MRR sebesar 1, dan nDCG sebesar 1. Hasil tersebut menunjukkan bahwa sistem mampu menghasilkan rekomendasi yang relevan dengan kualitas ranking yang baik. Tingginya nilai evaluasi menunjukkan bahwa model mampu memahami hubungan semantik antara *query* dan sinopsis film secara baik sehingga menghasilkan rekomendasi yang relevan dan sesuai dengan kebutuhan pengguna.

Hasil tersebut sejalan dengan firman Allah Swt. dalam QS. *Al-Hujurat* surat ke 49 ayat 6 yang berbunyi:

يٰٓاَيُّهَا الَّذِيْنَ اٰمَنُوْٓا اِنْ جَاۤءَكُمْ فَاسِقٌۢ بِنَبَاٍ فَتَبَيَّنُوْٓا اَنْ تُصِيْبُوْا قَوْمًاۢ بِجَهَالَةٍ فَتُصْبِحُوْا عَلٰى مَا فَعَلْتُمْ نٰدِمِيْنَ

“Wahai orang-orang yang beriman! Jika seseorang yang fasik datang kepadamu membawa suatu berita, maka telitilah kebenarannya, agar kamu tidak mencelakakan suatu kaum karena kebodohan (kecerobohan), yang akhirnya kamu menyesali perbuatanmu itu.” (QS. Al-Hujurat /49: 6).

Menurut tafsir Ibnu Katsir, ayat ini menegaskan pentingnya melakukan tabayyun atau verifikasi terhadap suatu informasi sebelum mengambil keputusan. Sikap tergesa-gesa tanpa melakukan pemeriksaan dapat menyebabkan kesalahan dan penyesalan. Oleh karena itu, Islam mengajarkan agar setiap informasi dianalisis

dan diperiksa secara teliti agar menghasilkan keputusan yang benar dan bertanggung jawab.

Dalam penelitian ini, prinsip *tabayyun* tercermin pada proses evaluasi dan validasi sistem rekomendasi. Hasil rekomendasi yang dihasilkan sistem tidak langsung digunakan sebagai dasar penilaian, tetapi terlebih dahulu divalidasi oleh validator melalui proses penentuan relevansi antara *query* pengguna dan sinopsis film yang direkomendasikan. Selain itu, penggunaan metrik evaluasi *Precision*, MAP, MRR, dan nDCG memungkinkan kualitas rekomendasi diukur secara objektif dan terukur. Hasil evaluasi yang diperoleh menunjukkan bahwa sistem mampu menghasilkan rekomendasi yang relevan berdasarkan skenario pengujian yang digunakan.

Dengan demikian, penelitian ini tidak hanya menunjukkan keberhasilan dari sisi teknologi dalam menghasilkan sistem rekomendasi film yang efektif, tetapi juga mencerminkan penerapan nilai-nilai Islam dalam pengolahan informasi, yaitu memilih alternatif terbaik, melakukan verifikasi secara teliti, serta menghasilkan keputusan yang bermanfaat dan bertanggung jawab bagi pengguna.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil pengujian dan pembahasan yang telah dilakukan, dapat diperoleh beberapa kesimpulan sebagai berikut:

1. Sistem rekomendasi film berbasis sinopsis menggunakan SBERT dan *Cosine Similarity* berhasil dibangun dan mampu menghasilkan rekomendasi film berdasarkan kesamaan semantik antara *query* pengguna dan sinopsis film. Hasil evaluasi menunjukkan bahwa sistem mampu menghasilkan rekomendasi yang relevan dengan performa nilai metrik *Precision*, MAP, MRR, dan nDCG yang secara umum berada di atas 0.94 pada seluruh skenario pengujian.
2. Penggunaan *preprocessing* memberikan peningkatan performa yang relatif kecil terhadap kualitas rekomendasi. Kelompok pengujian yang menggunakan *preprocessing* memperoleh nilai *Precision* sebesar 0.971, MAP sebesar 0.985, MRR sebesar 1, dan nDCG sebesar 0.994. Sementara itu, kelompok tanpa *preprocessing* memperoleh nilai *Precision* sebesar 0.968, MAP sebesar 0.988, MRR sebesar 1, dan nDCG sebesar 0.995. Hasil tersebut menunjukkan bahwa model SBERT tetap mampu memahami makna *query* dengan baik meskipun tanpa tahapan *preprocessing* secara lengkap.
3. Panjang *query* mempengaruhi kualitas rekomendasi yang dihasilkan sistem. *Query* pendek menghasilkan performa yang lebih baik dibandingkan *query* panjang dengan nilai *Precision* sebesar 0.981, MAP sebesar 0.990, dan nDCG sebesar 0.996. Sementara itu, *query* panjang memperoleh nilai *Precision*

sebesar 0.958, MAP sebesar 0.981, dan nDCG sebesar 0.993. Namun demikian, kedua jenis *query* tetap menghasilkan kualitas rekomendasi yang sangat baik.

4. Jumlah rekomendasi yang ditampilkan mempengaruhi kualitas hasil rekomendasi. Kelompok Top-5 menghasilkan nilai *Precision* sebesar 0.980, MAP sebesar 0.990, dan nDCG sebesar 0.996, sedangkan kelompok Top-10 menghasilkan nilai *Precision* sebesar 0.959, MAP sebesar 0.980, dan nDCG sebesar 0.993. Hasil tersebut menunjukkan bahwa rekomendasi dengan jumlah yang lebih sedikit cenderung memiliki tingkat relevansi yang lebih tinggi.
5. Skenario terbaik diperoleh pada Skenario 5 (tanpa *preprocessing*, *query* pendek, dan Top-5) dengan nilai *Precision* sebesar 0.990, MAP sebesar 1, MRR sebesar 1, dan nDCG sebesar 1. Hasil evaluasi menunjukkan bahwa sistem mampu menghasilkan rekomendasi yang tertinggi berdasarkan skenario pengujian dan penilaian satu validator yang digunakan dalam penelitian ini.

5.2 Saran

Berdasarkan hasil penelitian yang telah dilakukan, terdapat beberapa saran yang dapat diberikan untuk pengembangan penelitian selanjutnya, antara lain:

1. Penelitian selanjutnya dapat menggunakan dataset film dengan jumlah yang lebih besar dan lebih beragam agar sistem dapat diuji pada skala data yang lebih luas.
2. Penelitian selanjutnya dapat membandingkan performa SBERT dengan metode representasi teks lainnya seperti TF-IDF, BM25, BERT, RoBERTa, dan MPNet untuk memperoleh gambaran peningkatan performa yang lebih objektif.

3. Pengembangan penelitian selanjutnya dapat dilakukan dengan mengkombinasikan pendekatan *content-based filtering* dan *collaborative filtering* sehingga sistem tidak hanya mempertimbangkan kesamaan sinopsis film, tetapi juga preferensi pengguna berdasarkan riwayat interaksi.
4. Penelitian selanjutnya dapat menerapkan evaluasi yang melibatkan lebih banyak validator atau menggunakan penilaian langsung dari pengguna (*user study*) untuk memperoleh hasil evaluasi yang lebih komprehensif.
5. Sistem dapat dikembangkan menjadi aplikasi web atau aplikasi *mobile* yang terintegrasi dengan basis data film secara *real-time* sehingga dapat digunakan secara langsung oleh pengguna dalam proses pencarian dan pemilihan film.

DAFTAR PUSTAKA

Apura M. Bhavsar, Hitesh S. Girase, Dipak S. Lohar, & Aatif Z. Shaikh. (2024). Content-Based Movie Recommendation System. *INTERNATIONAL JOURNAL OF PRPGRESIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)*, *05*(04).

Ba, J. L., Kiros, J. R., & Hinton, G. E. (2016). *Layer Normalization* (Version 1). arXiv. https://doi.org/10.48550/ARXIV.1607.06450

Bhowmick, H., Chatterjee, A., & Sen, J. (2021). *Comprehensive Movie Recommendation System* (Version 1). arXiv. https://doi.org/10.48550/ARXIV.2112.12463

Christopher D. Manning, Prabhakar Raghavan, & Hinrich Schütze. (2008). *Introduction to Information Retrieval*. Cambridge University Press. https://nlp.stanford.edu/IR-book/

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding* (Version 2). arXiv. https://doi.org/10.48550/ARXIV.1810.04805

Dr. Abdullah bin Muhammad bin Abdurahman bin Ishaq Al-Sheikh. (1994). *Lubaabut Tatsiir Min Ibni Katsiir*. Pustaka Imam asy-Syafi' i.

Dwi Ayu Nouvalina & Kusuma Hati. (2024). Sistem Rekomendasi Produk Skin Care Berdasarkan Permasalahan Kulit Wajah dengan Metode Content Based Filtering. *Jurnal Teknik Informatika STMIK Antar Bangsa*, *10*(2), 60–67. https://doi.org/10.51998/jti.v10i2.586

Grezes, F., Allen, T., Blanco-Cuaresma, S., Accomazzi, A., Kurtz, M. J., Shapurian, G., Henneken, E., Grant, C. S., Thompson, D. M., Hostetler, T. W., Templeton, M. R., Lockhart, K. E., Chen, S., Koch, J., Jacovich, T., & Protopapas, P. (2022). *Improving astroBERT using Semantic Textual Similarity* (arXiv:2212.00744). arXiv. https://doi.org/10.48550/arXiv.2212.00744

Gunathilaka, T. M. A. U., Manage, P. D., Zhang, J., Li, Y., & Kelly, W. (2025). Addressing sparse data challenges in recommendation systems: A systematic review of rating estimation using sparse rating data and profile enrichment techniques. *Intelligent Systems with Applications*, *25*, 200474. https://doi.org/10.1016/j.iswa.2024.200474

Harahap, R. M., & Rachman, A. N. (2025). SISTEM REKOMEDASI FILM BERDASARKAN KEMIRIPAN DESKRIPSI CERITA MENGGUNAKAN METODE CONTENT BASED FILTERING. *Jurnal Informatika Dan Teknik Elektro Terapan*, *13*(3). https://doi.org/10.23960/jitet.v13i3.6577

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770–778. https://doi.org/10.1109/CVPR.2016.90

Javaji, S. R., & Sarode, K. (2023). *Hybrid Recommendation System using Graph Neural Network and BERT Embeddings* (Version 1). arXiv. https://doi.org/10.48550/ARXIV.2310.04878

Lim, D., & Lee, T. (2026). Enhanced Recommender System with Sentiment Analysis of Review Text and SBERT Embeddings of Item Descriptions. *Mathematics*, *14*(1), 184. https://doi.org/10.3390/math14010184

Liu, Q., Kusner, M. J., & Blunsom, P. (2020). *A Survey on Contextual Embeddings* (Version 2). arXiv. https://doi.org/10.48550/ARXIV.2003.07278

Munkholm, N. B., Alphinas, R., & Tambo, T. (2024). *Collaborative filtering, K-nearest neighbor and cosine similarity in home decor recommender systems* (Version 1). arXiv. https://doi.org/10.48550/ARXIV.2402.06233

Mustafa, A., Mheidat, H., & Shatnawi, A. (2024). A semantic similarity search engine for movies. *International Journal of Electrical and Computer Engineering (IJECE)*, *14*(6), 7137. https://doi.org/10.11591/ijece.v14i6.pp7137-7144

Nurma Romihim Fadlilah. (2025). *PERINGKASAN TEKS EKSTRAKTIF PADA ARTIKEL BERITA BAHASA INDONESIA MENGGUNAKAN METODE BERT DAN COSINE SIMILARITY* [UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM MALANG]. http://etheses.uin-malang.ac.id/id/eprint/79135

Ortakci, Y. (2024). Revolutionary text clustering: Investigating transfer learning capacity of SBERT models through pooling techniques. *Engineering Science and Technology, an International Journal*, *55*, 101730. https://doi.org/10.1016/j.jestch.2024.101730

Patil, S., Bijapur, R. R., Bidargaddi, A. P., Kothari, S. D., & Dabade, S. H. (2024). Enhancing Movie Recommendation Systems with BERT: A Deep Learning Approach. *2024 5th International Conference for Emerging Technology (INCET)*, 1–8. https://doi.org/10.1109/INCET61516.2024.10593284

Rahmatabadi, A. F., Bastanfard, A., Amini, A., & Saboohi, H. (n.d.). *Proposing a Semantic Movie Recommendation System Enhanced by ChatGPT's NLP Results*.

Ram, N., Kuzmin, D., Chio, E. K. I., Alzantot, M. F., Ontanon, S., Jash, A., & Li, J. Y. (2023). *Multi-Task End-to-End Training Improves Conversational Recommendation* (Version 1). arXiv. https://doi.org/10.48550/ARXIV.2305.06218

Reimers, N., & Gurevych, I. (2019). *Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks* (arXiv:1908.10084). arXiv. https://doi.org/10.48550/arXiv.1908.10084

Singh, K., Mishra, M., & Singh, Er. S. (2024). Content-based Recommender System Using Cosine Similarity. *International Journal for Research in Applied Science and Engineering Technology*, *12*(5), 2541–2548. https://doi.org/10.22214/ijraset.2024.61835

Streaming Reaches Historic TV Milestone, Eclipses Combined Broadcast and Cable Viewing For First Time. (n.d.). *Nielsen*. Retrieved October 22, 2025, from https://www.nielsen.com/news-center/2025/streaming-reaches-historic-tv-milestone-eclipses-combined-broadcast-and-cable-viewing-for-first-time/

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). *Attention Is All You Need* (Version 7). arXiv. https://doi.org/10.48550/ARXIV.1706.03762

Video Streaming Market Size, Share & Trends | Growth [2032]. (n.d.). Retrieved October 22, 2025, from https://www.fortunebusinessinsights.com/video-streaming-market-103057

Wang, W., Wei, F., Dong, L., Bao, H., Yang, N., & Zhou, M. (2020). *MiniLM: Deep Self-Attention Distillation for Task-Agnostic Compression of Pre-Trained Transformers* (Version 2). arXiv. https://doi.org/10.48550/ARXIV.2002.10957

Yeole Madhavi B., Rokade Monika D., & Khatal Sunil S. (2021). Movie Recommendation System using Free Text. *International Journal of Research Publication an Reviews*, *2*(7), 877–891.

Yuan, H., & Hernandez, A. A. (2023). User Cold Start Problem in Recommendation Systems: A Systematic Review. *IEEE Access*, *11*, 136958–136977. https://doi.org/10.1109/ACCESS.2023.3338705

Zhou, K., Ethayarajh, K., Card, D., & Jurafsky, D. (2022). Problems with Cosine as a Measure of Embedding Similarity for High Frequency Words. *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 401–423. https://doi.org/10.18653/v1/2022.acl-short.45

LAMPIRAN

Lampiran I Biodata Validator

Deby Farhadiba, S.Pd.

E-mail:
farhadibadeby@gmail.com

Institution: MAN 1 Banyuwangi

EDUCATIONAL HISTORY

- **UNIVERSITAS NEGERI MALANG**
  English Language Teaching program
  (GPA 3.88)
- **SMA NEGERI 1 GIRI**
  Science program
- **SMPN 1 BANYUWANGI**
- **SDN KEPATIHAN**

EXPERIENCES

2019 - 2022: Tutor in PRIMAGAMA

2019: ISoLEC 2019 (International Seminar on Language, Education, and Culture)

- Attending the seminar as a presenter to present my research in teacher development.

2018: Bahasa Indonesia tutor for Critical Language Scholarship program

- Bahasa Indonesia tutor for American students sponsored by BIPA (Bahasa Indonesia bagi Penutur Asing) and the U.S. Department of State.

2017: Bahasa Indonesia tutor for In-Country program

- Bahasa Indonesia tutor for Thai students sponsored by BIPA (Bahasa Indonesia bagi Penutur Asing) and Walailak University of Thailand.

2016-2018: A member of HMJ Sastra Inggris Universitas Negeri Malang

Lampiran II Hasil Validasi

Hasil Validasi *Query* Panjang Tanpa *Preprocessing*

query	index	id	title	overview	similarity_score	Label 0/1
nd queens dealing with royal politics, power struggles, a	0	81005	Jack the Giant	The story of an ancient war that is reignited when a young farmhand unwittingly opens a gateway between our world	0.466199994	1
	1	20542	Delgo	In a divided land, it takes a rebellious boy and his clandestine love for a Princess of an opposing race to stop a war	0.449299991	1
	2	14517	Mirrormask	In a fantasy world of opposing kingdoms, a 15-year old girl must find the fabled MirrorMask in order to save the	0.442900002	1
	3	15208	Bathory: Countess	Bathory is based on the legends surrounding the life and deeds of Countess Elizabeth Bathory known as the	0.432700008	1
	4	58595	Snow White and	After the Evil Queen marries the King, she performs a violent coup in which the King is murdered and his daughter,	0.421200007	0
	5	6278	Reign of Fire	In post-apocalyptic England, an American volunteer and a British survivor team up to fight off a brood of fire-	0.415899992	1
	6	62764	Mirror Mirror	After she spends all her money, an evil enchantress queen schemes to marry a handsome, wealthy prince. There's	0.411300004	1
	7	1491	The Illusionist	With his eye on a lovely aristocrat, a gifted illusionist named Eisenheim uses his powers to win her away from her	0.399500012	1
	8	4523	Enchanted	The beautiful princess Giselle is banished by an evil queen from her magical, musical animated land and finds	0.393000007	0
	9	309809	The Little Prince	Based on the best-seller book 'The Little Prince', the movie tells the story of a little girl that lives with resignation in a	0.388300002	0
an adventure story about a journey across the ocean facing dangers, storms, and unexpected challenges at sea	0	152747	All Is Lost	Deep into a solo voyage in the Indian Ocean, an unnamed man (Redford) wakes to find his 39-foot yacht taking on	0.567799985	1
	1	7364	Sahara	Scouring the ocean depths for treasure-laden shipwrecks is business as usual for a thrill-seeking underwater	0.497000009	1
	2	8619	Master and	After an abrupt and violent encounter with a French warship inflicts severe damage upon his ship, a captain of the	0.487699986	1
	3	205775	In the Heart of the	In the winter of 1820, the New England whaling ship Essex was assaulted by something no one could believe: a	0.486099988	1
	4	503	Poseidon	A packed cruise ship traveling the Atlantic is hit and overturned by a massive wave, compelling the passengers to	0.462799996	1
	5	2756	The Abyss	A civilian oil rig crew is recruited to conduct a search and rescue effort when a nuclear submarine mysteriously	0.452199996	1
	6	4476	Legends of the Fall	An epic tale of three brothers and their father living in the remote wilderness of 1900s USA and how their lives are	0.439999998	0
	7	9804	Waterworld	In a futuristic world where the polar ice caps have melted and made Earth a liquid planet, a beautiful barmaid	0.427599996	1
	8	78698	Big Miracle	Based on an inspiring true story, a small-town news reporter (Krasinski) and a Greenpeace volunteer (Barrymore)	0.423599988	1
	9	9457	Deep Rising	A group of heavily armed hijackers board a luxury ocean liner in the South Pacific Ocean to loot it, only to do battle	0.415899992	1

Hasil Validasi *Query* Pendek Tanpa *Preprocessing*

query	index	id	title	overview	similarity_score	Label 0/1
king and queens	0	58595	Snow White and the	After the Evil Queen marries the King, she performs a violent coup in which the King is murdered and his daughter,	0.396200001	1
	1	62764	Mirror Mirror	After she spends all her money, an evil enchantress queen schemes to marry a handsome, wealthy prince. There's	0.391499996	1
	2	6278	Reign of Fire	In post-apocalyptic England, an American volunteer and a British survivor team up to fight off a brood of fire-	0.380800009	1
	3	810	Shrek the Third	The King of Far Far Away has died and Shrek and Fiona are to become King & Queen. However, Shrek wants	0.369500011	1
	4	14517	Mirrormask	In a fantasy world of opposing kingdoms, a 15-year old girl must find the fabled MirrorMask in order to save the	0.364499986	1
	5	1491	The Illusionist	With his eye on a lovely aristocrat, a gifted illusionist named Eisenheim uses his powers to win her away from her	0.363000005	1
	6	9610	Conan the Destroyer	Based on a character created by Robert E. Howard, this fast-paced, occasionally humorous sequel to Conan the	0.361200005	1
	7	241259	Alice Through the	In the sequel to Tim Burton's "Alice in Wonderland", Alice Kingsleigh returns to Underland and faces a new	0.345699996	1
	8	20542	Delgo	In a divided land, it takes a rebellious boy and his clandestine love for a Princess of an opposing race to stop a war	0.3398	1
	9	408	Snow White and the	A beautiful girl, Snow White, takes refuge in the forest in the house of seven dwarfs to hide from her stepmother,	0.337900013	1
sea adventure	0	7364	Sahara	Scouring the ocean depths for treasure-laden shipwrecks is business as usual for a thrill-seeking underwater	0.548600018	1
	1	152747	All Is Lost	Deep into a solo voyage in the Indian Ocean, an unnamed man (Redford) wakes to find his 39-foot yacht taking on	0.519699991	1
	2	57800	Ice Age: Continental	Manny, Diego, and Sid embark upon another adventure after their continent is set adrift. Using an iceberg as a ship,	0.517700016	1
	3	9804	Waterworld	In a futuristic world where the polar ice caps have melted and made Earth a liquid planet, a beautiful barmaid	0.511399984	1
	4	8619	Master and	After an abrupt and violent encounter with a French warship inflicts severe damage upon his ship, a captain of the	0.490799993	1
	5	9457	Deep Rising	A group of heavily armed hijackers board a luxury ocean liner in the South Pacific Ocean to loot it, only to do battle	0.481900007	1
	6	15511	VeggieTales: The	Set Sail For Adventure! A boatload of beloved VeggieTales pals embark on a fun and fresh pirate adventure with	0.477999985	1
	7	11968	Into the Blue	When they take some friends on an extreme sport adventure, the last thing Jared and Sam expect to see below	0.475499988	1
	8	72197	The Pirates! In an	The luxuriantly bearded Pirate Captain is a boundlessly enthusiastic, if somewhat less-than-successful, terror of the	0.47299999	1
	9	2756	The Abyss	A civilian oil rig crew is recruited to conduct a search and rescue effort when a nuclear submarine mysteriously	0.471700013	1

Hasil Validasi *Query* Panjang *Preprocessing*

query	index	id	title	overview	similarity_score	Label 0/1
nd queens dealing with royal politics, power struggles, a	0	58595	Snow White and the	After the Evil Queen marries the King, she performs a violent coup in	0.497500002	1
	1	15208	Bathory: Countess of	Bathory is based on the legends surrounding the life and deeds of	0.482199997	1
	2	14517	Mirrormask	In a fantasy world of opposing kingdoms, a 15-year old girl must find	0.469999999	1
	3	81005	Jack the Giant Slayer	The story of an ancient war that is reignited when a young farmhand	0.458999991	1
	4	62764	Mirror Mirror	After she spends all her money, an evil enchantress queen schemes to	0.452699989	1
	5	810	Shrek the Third	The King of Far Far Away has died and Shrek and Fiona are to become	0.451900005	1
	6	62177	Brave	Brave is set in the mystical Scottish Highlands, where Mérida is the	0.449600011	1
	7	102651	Maleficent	The untold story of Disney's most iconic villain from the 1959 classic	0.429800004	1
	8	18937	Quest for Camelot	During the times of King Arthur, Kayley is a brave girl who dreams of	0.423799992	1
	9	309809	The Little Prince	Based on the best-seller book 'The Little Prince', the movie tells the	0.420599997	0
an adventure story about a journey across the ocean facing dangers, storms, and unexpected challenges at sea	0	152747	All Is Lost	Deep into a solo voyage in the Indian Ocean, an unnamed man	0.613099992	1
	1	503	Poseidon	A packed cruise ship traveling the Atlantic is hit and overturned by a	0.532199979	1
	2	7364	Sahara	Scouring the ocean depths for treasure-laden shipwrecks is business	0.525399983	1
	3	2756	The Abyss	A civilian oil rig crew is recruited to conduct a search and rescue effort	0.516300023	1
	4	57800	Ice Age: Continental	Manny, Diego, and Sid embark upon another adventure after their	0.482499987	1
	5	8619	Master and	After an abrupt and violent encounter with a French warship inflicts	0.473199993	1
	6	9457	Deep Rising	A group of heavily armed hijackers board a luxury ocean liner in the	0.470600009	1
	7	253412	Everest	Inspired by the incredible events surrounding a treacherous attempt to	0.464100003	0
	8	251979	The Perfect Wave	"The Perfect Wave" is the true story of Ian McCormack who grew up	0.460799992	1
	9	9488	Spy Kids 2: The Island	Exploring the further adventures of Carmen and Juni Cortez, who have	0.449900001	0

Hasil Validasi *Query* Pendek *Preprocessing*

query	index	id	title	overview	similarity_score	label 0/1
king and queens	0	58595	Huntsman	which the King is murdered and his daughter, Snow White, is taken	0.557399988	1
	1	11979	Queen of the Damned	existence he has now become this generations new Rock God. While	0.492599994	1
	2	810	Shrek the Third	become King & Queen. However, Shrek wants to return to his	0.487699986	1
	3	62764	Mirror Mirror	to marry a handsome, wealthy prince. There's just one problem - he's	0.450500011	1
	4	164372	The Snow Queen	landscape, with no light, no joy, no happiness, and no free will. A	0.4472	1
	5	408	Seven Dwarfs	seven dwarfs to hide from her stepmother, the wicked Queen. The	0.435799986	1
	6	18937	Quest for Camelot	following her late father as a Knight of the Round Table. The evil	0.428200006	1
	7	62177	Brave	princess of a kingdom ruled by King Fergus and Queen Elinor. An	0.410600007	1
	8	12155	Alice in Wonderland	dunce of an English nobleman. At her engagement party, she escapes	0.406100005	1
	9	8840	DragonHeart	skies, a knight named Bowen comes face to face and heart to heart	0.400299996	1
sea adventure	0	7364	Sahara	as usual for a thrill-seeking underwater adventurer and his	0.622799993	1
	1	15511	Pirates Who Don't Do	embark on a fun and fresh pirate adventure with their trademark	0.577499986	1
	2	57800	Drift	continent is set adrift. Using an iceberg as a ship, they encounter sea	0.574599981	1
	3	152747	All Is Lost	(Redford) wakes to find his 39-foot yacht taking on water after a	0.568099976	1
	4	11968	Into the Blue	thing Jared and Sam expect to see below the shark-infested waters is	0.527199984	1
	5	9488	of Lost Dreams	have now joined the family spy business as Level 2 OSS agents.	0.526499987	0
	6	9457	Deep Rising	South Pacific Ocean to loot it, only to do battle with a series of large-	0.518999994	1
	7	72197	Adventure with	somewhat less-than-successful, terror of the High Seas. With a rag-	0.515699983	1
	8	22	Caribbean: The Curse	Caribbean Sea, butts heads with a rival pirate bent on pillaging the	0.512000024	1
	9	1907	The Beach	possession of a strange map. Rumours state that it leads to a solitary	0.479299992	1

Lampiran III Hasil Uji Coba

Rata-rata Hasil Uji Coba *Query* Pendek *Preprocessing* Top-5 dan Top-10

Top-K	Average of Precision	Average of MAP	Average of MRR	Average of nDCG
5	0.990	0.990	1	0.995
10	0.965	0.980	1	0.992
Grand Total	**0.977**	**0.985**	**1**	**0.993**

Rata-rata Hasil Uji Coba *Query* Panjang *Preprocessing* Top-5 dan Top-10

Top-K	Average of Precision	Average of MAP	Average of MRR	Average of nDCG
5	0.980	0.989	1	0.996
10	0.950	0.980	1	0.993
Grand Total	**0.965**	**0.984**	**1**	**0.994**

Rata-rata Hasil Uji Coba *Query* Pendek Tanpa *Preprocessing* Top-5 dan Top-10

Top-K	Average of Precision	Average of MAP	Average of MRR	Average of nDCG
5	0.990	1	1	1
10	0.980	0.9901	1	0.9969
Grand Total	**0.985**	**0.995**	**1**	**0.998**

Rata-rata Hasil Uji Coba *Query* Panjang Tanpa *Preprocessing* Top-5 dan Top-10

Top-K	Average of Precision	Average of MAP	Average of MRR	Average of Ndcg
5	0.960	0.984	1	0.993
10	0.940	0.973	1	0.989
Grand Total	**0.950**	**0.979**	**1**	**0.991**