# BAB I

# PENDAHULUAN

## **Latar Belakang**

Perkembangan teknologi informasi, khususnya dalam bidang pemrosesan bahasa alami (*natural language processing*), telah mengubah cara pengguna berinteraksi dengan berbagai platform digital, termasuk toko buku daring. Meningkatnya jumlah konten digital menyebabkan fenomena kelebihan informasi (*information overload*), yang membuat pengguna kesulitan dalam menyaring dan menemukan informasi yang benar-benar relevan dengan kebutuhannya.

Di Indonesia, beberapa platform buku digital besar telah menyediakan ribuan judul buku yang dapat diakses secara daring. Namun, banyaknya pilihan justru sering kali menyulitkan pengguna dalam menemukan buku yang sesuai dengan minat atau kebutuhan pribadi. Hal ini menunjukkan perlunya sistem yang dapat membantu pengguna dalam menjelajahi dan memilih buku dengan cara yang lebih cerdas dan terpersonalisasi.

Salah satu pendekatan yang dapat digunakan untuk mengatasi masalah tersebut adalah sistem rekomendasi berbasis konten. Sistem ini bekerja dengan menganalisis elemen teks seperti judul dan deskripsi buku untuk mengukur tingkat kesamaan antar buku. Melalui pendekatan semantik, sistem tidak hanya mencocokkan kata kunci secara literal, tetapi juga memahami konteks makna dari setiap teks.

Kemajuan dalam pengembangan model pemrosesan bahasa berbasis *transformer* telah membuka peluang baru dalam menghasilkan representasi teks yang lebih akurat dan bermakna. Model-model ini mampu menangkap konteks kalimat secara mendalam sehingga dapat digunakan untuk mengukur kemiripan antar buku berdasarkan makna, bukan sekadar kemiripan kata.

Namun, sebagian besar platform buku daring di Indonesia masih mengandalkan sistem pencarian konvensional berbasis kata kunci. Sistem seperti ini cenderung mengabaikan konteks dan makna semantik yang terkandung dalam teks. Di sisi lain, struktur data yang tidak seragam, seperti judul dan deskripsi buku yang beragam, menimbulkan tantangan dalam mengolah informasi secara efektif. Akibatnya, sistem rekomendasi yang dihasilkan menjadi kurang akurat dan tidak responsif terhadap preferensi pengguna.

Untuk mengatasi masalah tersebut, penelitian ini mengembangkan sistem rekomendasi buku berbasis representasi semantik yang diperoleh dari model embedding teks. Sistem ini menghitung kemiripan antar buku menggunakan teknik perhitungan berbasis vektor, sehingga dapat merekomendasikan buku-buku yang secara makna memiliki kedekatan satu sama lain. Prototipe sistem dikembangkan dalam bentuk aplikasi web menggunakan platform yang sederhana dan interaktif, sehingga pengguna dapat dengan mudah memperoleh rekomendasi buku yang sesuai dengan preferensinya.

Penelitian ini diharapkan dapat memberikan solusi terhadap permasalahan pencarian dan pemilihan buku secara daring. Selain itu, penelitian ini juga memberikan kontribusi dalam pengembangan sistem informasi berbasis pemrosesan bahasa alami, serta mendukung transformasi digital dalam bidang literasi dan edukasi.

## **Identifikasi Masalah**

1. Representasi teks dari data buku yang tidak terstruktur, seperti judul dan deskripsi, masih kurang bermakna secara semantik.
2. Metode yang digunakan belum mampu menghasilkan embedding teks berkualitas tinggi untuk menangkap makna yang mendalam.
3. Proses perhitungan kesamaan antar buku masih belum efisien, sehingga menghambat kualitas sistem rekomendasi.
4. Sistem rekomendasi belum terintegrasi secara optimal ke dalam antarmuka web yang ramah pengguna dan interaktif.

## **Batasan Masalah**

1. Penelitian hanya menggunakan pendekatan sistem rekomendasi berbasis konten dengan sumber data berupa judul dan deskripsi buku.
2. Dataset yang digunakan terbatas pada data buku dari Gramedia dengan jumlah 1.877 entri setelah proses *preprocessing*.
3. Teknologi pengembangan terbatas pada Python dengan penggunaan pustaka Streamlit, Pandas, NumPy, Google GenAI, dan Scikit-learn.
4. Sistem tidak mempertimbangkan atribut lain seperti kategori buku, nama penulis, atau ulasan pengguna.
5. Evaluasi sistem terbatas pada pengujian fungsionalitas, tanpa mencakup evaluasi kuantitatif seperti *precision*, *recall*, atau *F1-score*.

## **Rumusan Masalah**

1. Bagaimana merancang pipeline *preprocessing* yang dapat menghasilkan representasi teks yang bermakna dari data buku?
2. Bagaimana mengembangkan metode untuk menghasilkan embedding teks menggunakan model *text-embedding-004*?
3. Bagaimana menganalisis kesamaan antar buku dengan menggunakan metode *cosine similarity* untuk keperluan rekomendasi?
4. Bagaimana merancang aplikasi web berbasis Streamlit yang mampu mengintegrasikan fitur *semantic search* dan sistem rekomendasi?

## **Tujuan Penelitian**

1. Merancang pipeline *preprocessing* yang dapat menyiapkan data teks dari buku secara terstruktur dan bermakna.
2. Mengembangkan metode untuk menghasilkan embedding teks dengan model *text-embedding-004*.
3. Melakukan analisis kesamaan antar buku menggunakan metode *cosine similarity* untuk menghasilkan rekomendasi yang relevan.
4. Merancang dan mengimplementasikan aplikasi web berbasis Streamlit yang mendukung *semantic search* dan sistem rekomendasi buku.

## **Manfaat Penelitian**

### Manfaat Praktis

1. Meningkatkan efisiensi dalam proses pencarian dan rekomendasi buku pada platform Gramedia.
2. Meningkatkan kepuasan pengguna melalui sistem yang menyediakan rekomendasi secara personal dan semantik.
3. Mendukung potensi peningkatan penjualan buku melalui pemberian rekomendasi yang relevan dan sesuai kebutuhan pengguna.

### Manfaat Teoritis

1. Memberikan kontribusi terhadap pengembangan sistem rekomendasi berbasis pemrosesan bahasa alami (NLP).
2. Menyediakan studi kasus penerapan semantic search dalam konteks toko buku daring.
3. Memperkaya literatur di bidang informatika mengenai integrasi embedding teks dengan antarmuka aplikasi web interaktif.

# BAB II

**TINJAUAN PUSTAKA**

## **A. Landasan Teori**

1. Sistem Rekomendasi  
   Sistem rekomendasi adalah sistem yang dirancang untuk memberikan saran kepada pengguna terkait item atau konten yang sesuai dengan preferensi mereka. Menurut Ricci et al. (2015), sistem rekomendasi dapat diklasifikasikan ke dalam tiga kategori utama: berbasis konten (content-based), berbasis kolaborasi (collaborative filtering), dan pendekatan hibrida (hybrid). Sistem rekomendasi berbasis konten memanfaatkan informasi intrinsik dari item, seperti deskripsi atau fitur, untuk mencari kemiripan antar item. Dalam konteks buku digital, sistem ini menggunakan teks judul dan deskripsi buku untuk mengukur relevansi antar buku.
2. Pemrosesan Bahasa Alami (Natural Language Processing/NLP)  
   *Natural Language Processing* (NLP) merupakan bidang dalam kecerdasan buatan yang berfokus pada interaksi antara komputer dan bahasa manusia. NLP mencakup berbagai teknik untuk memahami, menginterpretasi, dan menghasilkan bahasa alami (Jurafsky & Martin, 2021). Dalam sistem rekomendasi, NLP digunakan untuk mengekstraksi makna dari teks seperti deskripsi produk atau ulasan pengguna.
3. Representasi Teks dan Embedding  
   Representasi teks adalah proses mengubah teks menjadi format numerik agar dapat diproses oleh algoritma komputer. Teknik klasik seperti Bag-of-Words dan TF-IDF tidak mempertimbangkan konteks kata. Sebaliknya, teknik embedding modern seperti Word2Vec (Mikolov et al., 2013), GloVe (Pennington et al., 2014), dan BERT (Devlin et al., 2019) mampu menangkap makna kontekstual dari kata-kata. Sentence-BERT (Reimers & Gurevych, 2019) merupakan pengembangan dari BERT yang dioptimalkan untuk menghitung kesamaan antar kalimat atau dokumen. Model embedding terbaru seperti text-embedding-004 dari Google GenAI dirancang untuk efisiensi tinggi dengan mempertahankan kualitas semantik yang akurat.
4. Semantic Search  
   *Semantic search* adalah teknik pencarian informasi yang memahami makna konteks di balik permintaan pengguna, bukan sekadar mencocokkan kata kunci. Dengan menggunakan teknik embedding dan pengukuran seperti cosine similarity, semantic search mampu menemukan hasil yang secara makna lebih relevan (Zhang et al., 2020). Dalam aplikasi rekomendasi buku, pendekatan ini memungkinkan sistem menemukan buku-buku yang memiliki kedekatan semantik meskipun tidak menggunakan kata kunci yang sama.
5. Cosine Similarity  
   Cosine similarity adalah metode pengukuran kesamaan antara dua vektor dalam ruang multidimensi, yang banyak digunakan dalam sistem pencarian dan rekomendasi berbasis teks. Nilai cosine similarity berkisar antara -1 hingga 1, di mana nilai mendekati 1 menunjukkan tingkat kemiripan yang tinggi (Han et al., 2012). Dalam sistem rekomendasi berbasis embedding, cosine similarity digunakan untuk membandingkan vektor representasi antar buku.
6. Streamlit dan Teknologi Pendukung  
   Streamlit adalah framework berbasis Python untuk membangun aplikasi web interaktif secara cepat dan sederhana. Streamlit cocok untuk visualisasi data dan demonstrasi model machine learning. Dalam penelitian ini, Streamlit digunakan untuk membangun antarmuka sistem rekomendasi, didukung oleh pustaka seperti Pandas, NumPy, dan Scikit-learn untuk pemrosesan data dan analisis kesamaan.

## **B. Penelitian Relevan**

Penelitian oleh Zhang et al. (2020) melakukan survei komprehensif terhadap sistem rekomendasi berbasis deep learning dan menekankan pentingnya semantic embedding dalam meningkatkan relevansi hasil rekomendasi. Mereka menunjukkan bahwa penggunaan model berbasis transformer seperti BERT menghasilkan performa superior dalam menangkap konteks semantik.

Reimers dan Gurevych (2019) mengembangkan Sentence-BERT, sebuah model embedding yang disesuaikan untuk tugas pencocokan semantik antar kalimat atau dokumen. Model ini terbukti efisien dan akurat dalam penghitungan kemiripan semantik, menjadikannya ideal untuk sistem rekomendasi berbasis teks.

Liu et al. (2021) memperluas pendekatan ini dengan menggabungkan model embedding dan knowledge graph untuk meningkatkan performa rekomendasi dalam domain edukasi. Penelitian ini menyoroti pentingnya integrasi sumber data semantik dalam meningkatkan kualitas saran yang diberikan kepada pengguna.

Penelitian lain oleh Grand View Research (2022) memproyeksikan pertumbuhan signifikan pasar buku digital secara global, yang menuntut peningkatan kualitas sistem pencarian dan rekomendasi. Studi ini memberikan konteks relevansi penelitian dalam mendukung pengembangan sistem yang lebih adaptif dan responsif terhadap preferensi pengguna.

Meskipun banyak penelitian telah mengusulkan integrasi model embedding dalam sistem rekomendasi, masih terdapat kesenjangan pada penerapan praktis dalam konteks lokal seperti toko buku digital di Indonesia. Oleh karena itu, penelitian ini memberikan kontribusi dengan mengimplementasikan pendekatan semantic search berbasis text-embedding-004 dalam platform buku daring Gramedia, disertai pengembangan antarmuka interaktif berbasis Streamlit.

# BAB III

**METODE PENELITAN**

## **A. Waktu dan Tempat Penelitian**

Penelitian ini dilaksanakan selama lima bulan, dimulai pada bulan Mei 2024 hingga Maret 2025. Proses penelitian dilakukan secara daring dan luring. Pengumpulan data buku dilakukan melalui proses web scraping pada situs resmi Gramedia ([https://www.gramedia.com](https://www.gramedia.com/)) menggunakan ekstensi browser Automa, sedangkan proses pengolahan data dan pengembangan sistem dilakukan di rumah pribadi peneliti. Pemilihan platform Gramedia sebagai sumber data didasarkan pada tingginya jumlah koleksi buku digital yang tersedia serta popularitasnya sebagai salah satu toko buku daring terbesar di Indonesia. Lingkungan pengembangan perangkat lunak dan eksperimen dilakukan pada perangkat laptop dengan spesifikasi Intel Core i5 Generasi ke-8, RAM 8 GB, dan sistem operasi Windows 10.

## **B. Tahapan Penelitian**

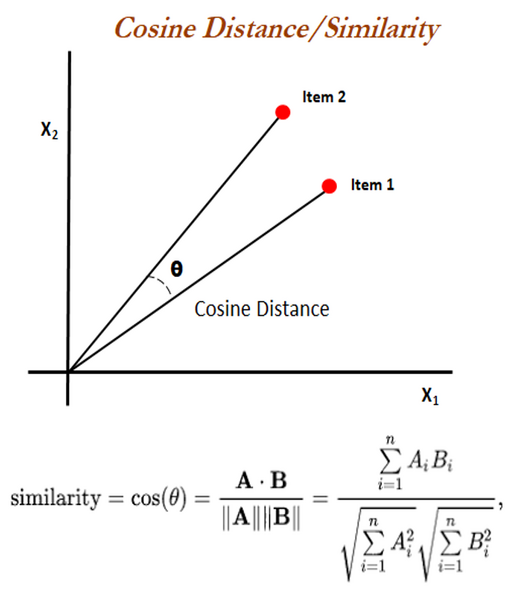
Tahapan dalam penelitian ini dilakukan secara sistematis dan terstruktur guna memastikan hasil yang diperoleh relevan, valid, dan dapat direproduksi. Adapun tahapan-tahapan tersebut dijelaskan sebagai berikut:

* 1. Studi Literatur  
     Peneliti mengumpulkan berbagai referensi dari jurnal ilmiah, buku, artikel online, dan publikasi lain yang relevan dengan sistem rekomendasi berbasis konten, NLP, semantic search, dan model embedding seperti BERT dan Sentence-BERT. Studi literatur ini bertujuan untuk memahami pendekatan-pendekatan yang telah digunakan dalam penelitian serupa serta untuk mengidentifikasi celah penelitian yang dapat dijadikan fokus.
  2. Pengumpulan Data  
     Data dikumpulkan dengan teknik web scraping menggunakan ekstensi Automa untuk mengotomatisasi ekstraksi data dari situs Gramedia. Informasi yang dikumpulkan meliputi judul buku, deskripsi buku, penulis, dan tautan gambar. Hasil scraping menghasilkan 1.877 entri buku yang menjadi dasar dari eksperimen.
  3. Pra-pemrosesan Data  
     Data yang diperoleh dari scraping selanjutnya diproses agar dapat digunakan dalam model embedding. Tahapan ini mencakup proses *lowercasing*, penggabungan (*concatenation*) antara judul dan deskripsi buku, serta pembersihan teks dari karakter-karakter khusus yang tidak relevan. Hasil pra-pemrosesan ini disimpan dalam format CSV untuk keperluan analisis selanjutnya.
  4. Embedding Teks dengan Model text-embedding-004  
     Setelah data siap, teks buku diubah menjadi representasi vektor menggunakan model *text-embedding-004* dari Google GenAI. Model ini dipilih karena kemampuannya menghasilkan representasi semantik berkualitas tinggi, sehingga cocok untuk mendeteksi kemiripan makna antar dokumen. Setiap entri teks diubah menjadi vektor berdimensi tetap yang kemudian digunakan dalam tahap perhitungan kemiripan.
  5. Penghitungan Kemiripan dengan Cosine Similarity  
     Untuk mengukur kedekatan semantik antar buku, digunakan metode cosine similarity. Teknik ini menghitung sudut kosinus antara dua vektor embedding. Semakin kecil sudutnya (semakin mendekati 1), maka kedua vektor dianggap semakin mirip secara semantik. Hasil dari proses ini berupa daftar buku yang mirip secara semantik untuk setiap buku dalam dataset.
  6. Perancangan Alur Kerja Sistem  
     Alur kerja sistem dirancang dengan pendekatan modular, terdiri dari tahapan input data pengguna, proses embedding dan pencocokan kemiripan, hingga penyajian hasil rekomendasi. Diagram alur kerja menggambarkan arus data dari pengguna hingga sistem mengembalikan hasil rekomendasi. Perancangan ini penting untuk memastikan proses berjalan secara otomatis, efisien, dan terintegrasi.
  7. Desain Antarmuka Pengguna (*UI Design*)  
     Desain antarmuka pengguna dilakukan sebelum proses pengembangan aplikasi. Antarmuka dirancang dengan prinsip user-friendly, minimalis, dan responsif menggunakan tools seperti Figma. Komponen UI yang dirancang meliputi halaman utama, input pencarian buku, hasil rekomendasi, serta halaman detail buku. Fokus desain adalah memudahkan pengguna dalam memahami dan menggunakan fitur pencarian serta hasil rekomendasi yang ditampilkan.
  8. Pengembangan Aplikasi Web  
     Tahap terakhir adalah pengembangan antarmuka pengguna menggunakan framework Streamlit. Antarmuka ini memungkinkan pengguna untuk mencari buku dan menerima rekomendasi berdasarkan kesamaan semantik dengan input yang diberikan. Komponen penting pada aplikasi meliputi input teks, daftar rekomendasi, dan tampilan detail buku. Aplikasi dikembangkan dengan menggunakan bahasa pemrograman Python dan pustaka tambahan seperti Pandas, NumPy, dan Scikit-learn.

## **C. Algoritma**

Penelitian ini menerapkan pendekatan algoritmik berbasis *semantic similarity* menggunakan model embedding dan cosine similarity. Berikut penjelasan detail dari pendekatan algoritmik yang digunakan:

1. Embedding dengan text-embedding-004  
   Model ini bekerja dengan cara memetakan teks ke dalam vektor berdimensi tinggi. Berbeda dengan teknik tradisional seperti TF-IDF atau Word2Vec, text-embedding-004 mampu memahami konteks secara lebih dalam melalui pelatihan pada data berskala besar dengan arsitektur transformer. Model ini dapat menangkap relasi semantik antara dua kalimat meskipun secara leksikal tidak identik.
2. Cosine Similarity  
   Setelah embedding vektor diperoleh, kemiripan antar buku dihitung dengan rumus cosine similarity sebagai berikut:



*Gambar x.x Cosine Similarity*

**A · B:** Ini adalah *dot product* dai **A** dan **B.** *Dot product* mengukur sebarapa besar kedua vector tersebut mengaah ke arah yang sama. Semakin besar nilainya, semakin mirip arah kedua vektor.

**|A|** dan **|B|:** Ini adalah Panjang dair vekto A dan B. Panjang dihitung untu menormalkan nilai *dot product,* sehingga hasil *cosine similarity* tidak dipengaruhi oleh besanya vekto, tetapi hanya oleh sudut di antara mereka.

Hasil dari rumus ini, cos(θ, adalah kosinus dari sudut θ antara kedua vektor, dengan rentang nilai antara **-1** hingga **1**:

* **Nilai mendekati 1**: Menunjukkan bahwa kedua vektor hampir sejajar atau mengarah ke arah yang sama, yang berarti entri buku memiliki kemiripan yang tinggi berdasarkan embedding mereka.
* **Nilai mendekati 0**: Menunjukkan bahwa kedua vektor saling tegak lurus, yang mengindikasikan tidak ada kemiripan yang signifikan antara kedua entri buku.
* **Nilai mendekati -1**: Menunjukkan bahwa kedua vektor mengarah ke arah yang berlawanan, yang berarti entri buku tersebut sangat berbeda.

1. Ranking dan Rekomendasi  
   Berdasarkan hasil penghitungan cosine similarity, sistem menyusun daftar rekomendasi dengan memilih n buku teratas yang memiliki skor tertinggi. Daftar ini kemudian disajikan melalui antarmuka Streamlit kepada pengguna dalam bentuk interaktif dan dapat disesuaikan dengan input pencarian.

Pendekatan algoritmik ini dipilih karena bersifat efisien, relevan dalam konteks semantic search, serta telah terbukti efektif dalam berbagai studi sebelumnya terkait sistem rekomendasi berbasis konten.

Dengan pendekatan ini, diharapkan sistem rekomendasi yang dikembangkan mampu memberikan hasil yang akurat, relevan secara semantik, dan mudah digunakan oleh pengguna akhir.

Referensi:  
Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of NAACL-HLT. <https://arxiv.org/abs/1810.04805>

Grand View Research. (2022). E-book Market Size, Share & Trends Analysis Report. <https://www.grandviewresearch.com/industry-analysis/e-book-market>

Han, J., Kamber, M., & Pei, J. (2012). Data Mining: Concepts and Techniques (3rd ed.). Morgan Kaufmann.

Jurafsky, D., & Martin, J. H. (2021). Speech and Language Processing (3rd ed.). Draft. <https://web.stanford.edu/~jurafsky/slp3/>

Liu, J., Zhao, Z., & Chen, W. (2021). Improving Recommendations with Multi-Modal Knowledge Graphs. Information Processing & Management, 58(4), 102592. <https://doi.org/10.1016/j.ipm.2021.102592>

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. arXiv preprint arXiv:1301.3781.

Pennington, J., Socher, R., & Manning, C. D. (2014). GloVe: Global Vectors for Word Representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP).

Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In Proceedings of EMNLP-IJCNLP. <https://arxiv.org/abs/1908.10084>

Ricci, F., Rokach, L., & Shapira, B. (2015). Recommender Systems Handbook. Springer.

Zhang, S., Yao, L., Sun, A., & Tay, Y. (2020). Deep learning based recommender system: A survey and new perspectives. ACM Computing Surveys, 52(1), 1-38. <https://doi.org/10.1145/3285029>

**DAFTAR PUSTAKA**

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of NAACL-HLT*. <https://arxiv.org/abs/1810.04805>

Grand View Research. (2022). *E-book market size, share & trends analysis report*. <https://www.grandviewresearch.com/industry-analysis/e-book-market>

Liu, J., Zhao, Z., & Chen, W. (2021). Improving recommendations with multi-modal knowledge graphs. *Information Processing & Management, 58*(4), 102592. https://doi.org/10.1016/j.ipm.2021.102592

Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence embeddings using Siamese BERT-networks. *Proceedings of EMNLP-IJCNLP*. <https://arxiv.org/abs/1908.10084>

Zhang, S., Yao, L., Sun, A., & Tay, Y. (2020). Deep learning based recommender system: A survey and new perspectives. *ACM Computing Surveys, 52*(5), 1–38. https://doi.org/10.1145/3285029