# BAB 2 TINJAUAN PUSTAKA

## 2.1 Penelitian Terkait

Penelitian ini mengkaji beberapa studi sebelumnya yang menggunakan metode Content-Based Filtering dan cosine similarity dalam sistem rekomendasi wisata. Tinjauan ini menjadi dasar pengembangan sistem yang lebih spesifik pada wisata alam di Sulawesi Utara. Rangkuman penelitian sejenis disajikan pada Tabel 2.1.

Tabel 2.1 Rangkuman Penelitian Sejenis

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Penulis & Tahun** | **Judul (Sumber)** | **Algoritma & Dataset** | **Hasil Utama** | **GAP (Celah Penelitian)** |
| 1 | Wafiqi dkk. (2024) | Prediksi Tingkat Stres pada Mahasiswa UNUGHA Cilacap Menggunakan KNN | K-Nearest Neighbor; survei 90 mahasiswa (84 train, 6 test) | Akurasi 83,33% menggunakan KNN | Data satu institusi (90 responden) dan hanya KNN yang digunakan; tidak dibandingkan dengan metode atau data lain, sehingga generalisasi terbatas. |
| 2 | Alfiyan Aldi dkk. (2020) | Deteksi Dini Stres Mahasiswa Menggunakan ID3 dan KNN | ID3 & KNN; kuesioner 300 mahasiswa | KNN ≈ 91%, ID3 ≈ 61% | Hanya ID3 dan KNN diuji; algoritma lain (mis. SVM, Random Forest) tidak dieksplorasi. Detail fitur tidak dijelaskan, sehingga efektivitas metode lain belum diketahui. |
| 3 | Almira V. Fadhila dkk. (2023) | Deteksi Tingkat Stres Berdasarkan Kualitas Tidur (SaYoPillow – Kaggle) | KNN, SVM, Decision Tree; dataset 630 sampel (8 fitur) | KNN & SVM 100%, Decision Tree 99% | Hanya benchmarking, tanpa optimasi parameter. Tidak ada interpretasi mendalam (SHAP) atau penyimpanan model |
| 4 | Kurniawan dkk. (2024) | Analisis Kesehatan Mental Mahasiswi Akhir (SVM) | SVM (kernel RBF & polinomial); kuesioner 249 mahasiswi | Akurasi tertinggi 68% (RBF γ=1, C=100) | Sampel terbatas (249 mahasiswi satu kampus) dan hanya SVM diuji. Akurasi rendah (68%) menandakan model kurang optimal; perlu eksplorasi algoritma lain. |
| 5 | Harahap & Yamasari (2024) | Klasifikasi Stres Mahasiswa Menggunakan ANN (RMSProp, SMOTE-N) | ANN (RMSProp optimizer, dgn/tdk SMOTE-N); data mahasiswa | Akurasi 96% (dgn SMOTE-N) vs 88% (tanpa) | Hanya ANN (RMSProp) diuji; optimizer atau model lain tidak dibandingkan. Detail dataset tidak dijelaskan, sehingga hasil sangat tergantung pada oversampling SMOTE-N. |
| 6 | Muhammad Reza (2024) | Prediksi Kesehatan Mental Tingkat Stres Mahasiswa (C4.5) | Decision Tree (C4.5); kuesioner 135 mahasiswa | Akurasi 88,8%[repository.ar-raniry.ac.id](https://repository.ar-raniry.ac.id/39140/1/Muhammad%20Reza_190212070.pdf#:~:text=stres%20dengan%20akurasi%20sebesar%2088%2C8,Penelitian%20ini%20menyimpulkan%20bahwa%20algoritma) | Hanya Decision Tree tanpa benchmarking algoritma modern; idak ada preprocessing lanjutan (label encoding, handling missing values) |

## 2.2 Sistem Rekomendasi

Sistem rekomendasi merupakan sistem cerdas yang berfungsi untuk memberikan saran atau pilihan kepada pengguna berdasarkan analisis terhadap data yang dikumpulkan sebelumnya, seperti riwayat interaksi, preferensi, dan perilaku pengguna. Sistem ini dirancang untuk mengurangi beban pengguna dalam memilih informasi yang sesuai dari lautan data yang tersedia. Dalam dunia teknologi informasi modern, sistem rekomendasi telah menjadi komponen penting dalam banyak aplikasi digital seperti e-commerce (misalnya Tokopedia, Amazon), layanan media streaming (seperti Netflix, Spotify), dan platform media sosial (seperti YouTube dan TikTok) (Wahyudi, 2017).

Sistem rekomendasi memanfaatkan pendekatan algoritmik untuk menganalisis dan memprediksi item yang kemungkinan besar disukai pengguna. Prediksi ini dilakukan secara otomatis dengan memanfaatkan kecerdasan buatan dan machine learning. Sistem semacam ini tidak hanya menyajikan informasi, tetapi juga secara aktif mempersonalisasi pengalaman pengguna melalui rekomendasi yang bersifat adaptif terhadap perubahan preferensi.

Dalam konteks teknologi informasi, sistem rekomendasi memiliki peranan strategis yang luas. Salah satu tujuan utama dari sistem rekomendasi adalah untuk meningkatkan kepuasan pengguna. Hal ini dicapai dengan menyajikan informasi atau produk yang relevan dengan kebutuhan atau minat mereka secara tepat waktu. Selain itu, sistem ini membantu mempercepat proses pencarian informasi, mengurangi beban kognitif pengguna, serta meningkatkan efisiensi dalam eksplorasi data.

Sistem rekomendasi juga dapat meningkatkan loyalitas pengguna terhadap platform digital. Ketika pengguna merasa bahwa sistem mampu memahami dan mengakomodasi preferensi mereka, kemungkinan besar mereka akan terus menggunakan platform tersebut. Di sisi bisnis, sistem rekomendasi mendukung optimalisasi strategi pemasaran, meningkatkan penjualan, dan menciptakan pengalaman pengguna yang dipersonalisasi (Suarjaya et al., 2024; Firdaos & Sujjada, 2025).

Dalam tataran teknis, sistem rekomendasi memanfaatkan big data dan machine learning untuk mendeteksi pola perilaku pengguna. Sistem ini kemudian menggunakan pola tersebut untuk menyarankan konten secara dinamis. Di era transformasi digital, pendekatan seperti ini menjadi fondasi penting dalam pengembangan layanan berbasis teknologi informasi (Padila, 2023).

## 2.3 *Content-Based Filtering*

Content-Based Filtering (CBF) merupakan pendekatan sistem rekomendasi yang mempersonalisasi pengalaman pengguna dengan menyarankan item berdasarkan kemiripan karakteristik antara item yang pernah disukai oleh pengguna dengan item yang tersedia. Prinsip utama dari metode ini adalah bahwa "jika pengguna menyukai suatu item, maka dia juga akan menyukai item yang serupa". Pendekatan ini secara eksplisit memanfaatkan data konten dari item, seperti teks deskripsi, kategori, atribut fitur, dan metadata lainnya (Nugroho et al., 2025).

CBF bekerja secara independen terhadap pengguna lain. Ini berbeda dengan collaborative filtering yang mengandalkan interaksi kolektif. Oleh karena itu, CBF sangat bermanfaat dalam sistem yang bersifat personal dan tidak memiliki basis pengguna yang besar. Dalam domain pariwisata, pendekatan ini menjadi sangat tepat karena objek wisata memiliki atribut yang dapat dikuantifikasi dan diolah, seperti lokasi geografis, jenis wisata (alam, sejarah, budaya, kuliner), serta ulasan dan deskripsi layanan.

CBF membentuk *profil pengguna* berdasarkan aktivitas historis yang dikumpulkan dari sistem, seperti pencarian, klik, atau pemberian rating terhadap suatu item. Profil ini menggambarkan preferensi pengguna dalam format yang terstruktur, misalnya vektor kata-kata kunci (keyword vector) atau nilai numerik dari fitur tertentu. Kemudian, sistem mencocokkan profil tersebut dengan *fitur konten* dari item lain menggunakan teknik seperti cosine similarity, TF-IDF (Term Frequency-Inverse Document Frequency), word embedding, atau bahkan deep learning untuk ekstraksi fitur lanjutan (Maulida & Hasrullah, 2025).

Sebagai contoh, jika seorang pengguna menyukai tempat wisata pegunungan yang menyediakan aktivitas hiking dan camping, maka sistem akan merekomendasikan objek wisata lain dengan fitur konten serupa, seperti taman nasional atau lokasi glamping. Proses ini berlangsung secara otomatis dan bersifat adaptif karena profil pengguna dapat terus diperbarui seiring waktu.

## 2.4 *Cosine Similarity*

Cosine Similarity adalah teknik populer dalam bidang machine learning dan information retrieval untuk mengukur tingkat kemiripan antara dua vektor dalam ruang multidimensi. Metode ini didasarkan pada konsep sudut antara dua vektor: semakin kecil sudutnya, semakin besar nilai kemiripan. Cosine similarity mengabaikan magnitudo dan hanya mempertimbangkan arah dari vektor, sehingga dua vektor yang searah akan memiliki nilai kemiripan maksimum, yaitu 1, sedangkan vektor yang saling tegak lurus (tidak memiliki kemiripan) akan bernilai 0.

Rumus dasar cosine similarity dituliskan sebagai:

Cosine Similiarity =

Dalam konteks sistem rekomendasi, vektor A dan B dapat mewakili profil item atau pengguna, tergantung pada pendekatan yang digunakan. Misalnya, dalam pendekatan content-based, vektor dapat merepresentasikan fitur konten dari dua item (misalnya dua tempat wisata, dua lagu, atau dua produk e-commerce), sementara dalam pendekatan collaborative filtering, vektor bisa mewakili pola rating dari dua pengguna yang dibandingkan (Primadewi et al., 2025).

Cosine similarity menjadi komponen penting dalam sistem rekomendasi karena efisiensinya dalam mengukur kemiripan antar objek berdasarkan fitur yang termuat dalam vektor. Dengan menghitung sudut antar dua vektor, sistem dapat dengan cepat menentukan seberapa mirip dua item atau dua pengguna tanpa harus mempertimbangkan perbedaan skala atau magnitudo antar fitur.

Dalam sistem rekomendasi berbasis content-based filtering, cosine similarity digunakan untuk membandingkan representasi fitur konten antar item. Misalnya, dua destinasi wisata yang memiliki deskripsi mirip misalnya sama-sama mencantumkan kata “pantai”, “sunset”, dan “foto” akan memiliki nilai cosine similarity yang tinggi. Hal ini memungkinkan sistem untuk menyarankan tempat wisata yang mirip dengan tempat yang sebelumnya disukai oleh pengguna (Budiman & Firmansyah, 2025).

Cosine similarity juga sangat cocok digunakan pada data hasil transformasi teks seperti TF-IDF. Representasi teks dalam bentuk vektor spasial ini memudahkan perhitungan kemiripan secara matematis, sehingga metode ini menjadi sangat populer dalam sistem rekomendasi artikel, buku, film, atau produk berbasis deskripsi teks.

## 2.5 Pendekatan Machine Learning dalam Sistem Rekomendasi

Machine learning merupakan cabang dari kecerdasan buatan (Artificial Intelligence) yang berfokus pada pengembangan algoritma yang mampu belajar dari data dan mengambil keputusan atau prediksi tanpa instruksi eksplisit. Dalam sistem rekomendasi, machine learning digunakan untuk mengenali pola interaksi antara pengguna dan item, serta menggeneralisasi pola tersebut untuk menghasilkan rekomendasi baru secara otomatis dan berkelanjutan (Aggarwal, 2016).

Dalam sistem rekomendasi, terdapat dua pendekatan utama berbasis machine learning, yaitu:

1. Collaborative Filtering (CF)

Pendekatan ini menggunakan data historis interaksi pengguna (seperti rating atau klik) tanpa memperhatikan konten dari item itu sendiri. Algoritma ini mengandalkan kesamaan perilaku antar pengguna atau antar item untuk membuat rekomendasi. Pendekatan ini banyak digunakan, namun memiliki kelemahan seperti cold start problem dan data sparsity (Koren et al., 2009).

1. Content-Based Filtering (CBF)

Pendekatan ini berfokus pada fitur atau atribut konten dari item. Sistem mempelajari preferensi pengguna berdasarkan konten item yang pernah disukai, lalu merekomendasikan item serupa berdasarkan kesamaan kontennya. Pada pendekatan ini, setiap item direpresentasikan sebagai vektor fitur, dan teknik seperti cosine similarity digunakan untuk mengukur kemiripan antar item. Meskipun cosine similarity bukan model pembelajaran dalam arti supervised learning, namun tetap dianggap bagian dari pendekatan machine learning karena beroperasi dalam kerangka vektorisasi dan pemrosesan data numerik (Ricci et al., 2015).

Dalam penelitian ini, pendekatan content-based filtering menjadi pilihan utama karena sistem difokuskan pada pemrosesan deskripsi konten destinasi wisata alam. Vektor fitur dari destinasi dibentuk dari data deskriptif seperti kategori, aktivitas, dan fasilitas, lalu dihitung tingkat kemiripannya menggunakan cosine similarity. Proses ini tidak hanya efisien, tetapi juga mampu beradaptasi dengan variasi data tanpa memerlukan data dari pengguna lain.

Machine learning juga memungkinkan sistem rekomendasi untuk bersifat adaptif terhadap perubahan preferensi pengguna dan dinamika data. Misalnya, jika ada destinasi baru dengan atribut yang serupa dengan destinasi yang sering dipilih pengguna, maka sistem dapat langsung memasukkannya ke dalam daftar rekomendasi. Hal ini menunjukkan bahwa sistem tidak bersifat statis, melainkan responsif terhadap penambahan dan perubahan data secara real-time (Zhang et al., 2019).

Walaupun penelitian ini belum menggunakan model pembelajaran kompleks seperti deep learning atau clustering berbasis unsupervised learning (misalnya K-Means atau DBSCAN), namun prinsip dasar dari machine learning tetap menjadi pondasi penting dalam membangun sistem yang cerdas, relevan, dan berskala. Penerapan cosine similarity dalam konteks ini juga menunjukkan bagaimana pendekatan sederhana dari machine learning dapat memberikan hasil rekomendasi yang efektif, terutama ketika dikombinasikan dengan pemodelan konten yang kaya dan deskriptif.

## 2.6 Penggunaan Python dalam Pengembangan Sistem Rekomendasi

Python merupakan bahasa pemrograman tingkat tinggi yang dirancang dengan sintaksis yang sederhana namun fleksibel, menjadikannya salah satu bahasa paling populer dalam pengembangan sistem berbasis data, termasuk sistem rekomendasi. Popularitas Python dalam dunia akademik dan industri teknologi didukung oleh ekosistem pustaka (library) yang kaya, khususnya di bidang data science, machine learning, dan natural language processing (Müller & Guido, 2017).

Dalam pengembangan sistem rekomendasi, Python memberikan kemudahan dalam berbagai tahap pembangunan sistem, mulai dari proses pembersihan dan transformasi data, representasi fitur dalam bentuk vektor, pengukuran kemiripan antar item, hingga evaluasi performa sistem. Pendekatan ini menjadikan Python tidak hanya alat bantu pemrograman, tetapi juga platform eksplorasi dan eksperimen ilmiah dalam bidang informasi dan kecerdasan buatan.

Salah satu pustaka yang sangat berpengaruh dalam sistem rekomendasi berbasis konten adalah scikit-learn, yang menyediakan berbagai modul penting seperti:

* TfidfVectorizer: untuk mentransformasikan teks deskripsi destinasi menjadi vektor berbobot berdasarkan frekuensi kata.
* cosine\_similarity: untuk mengukur tingkat kesamaan antara dua vektor item, seperti antara dua destinasi wisata berdasarkan konten deskriptifnya.
* train\_test\_split dan metrics: untuk membagi dataset dan menghitung metrik evaluasi seperti precision, recall, dan F1-score (Pedregosa et al., 2011).

Selain itu, library seperti pandas dan numpy memainkan peran penting dalam manipulasi data tabular dan perhitungan numerik, sedangkan matplotlib dan seaborn digunakan untuk visualisasi hasil evaluasi, heatmap kemiripan, serta distribusi preferensi pengguna terhadap konten tertentu.

Dalam konteks sistem rekomendasi wisata berbasis content-based filtering, Python digunakan untuk:

* Ekstraksi fitur konten destinasi seperti kategori wisata, aktivitas, dan fasilitas.
* Konversi teks menjadi vektor numerik menggunakan teknik TF-IDF.
* Pengukuran kemiripan antar destinasi menggunakan cosine similarity.
* Membangun interface prototipe sistem rekomendasi berbasis lokal atau web (menggunakan Flask/Django untuk implementasi lebih lanjut).
* Evaluasi hasil rekomendasi untuk mengukur tingkat relevansi hasil sistem terhadap ground truth atau preferensi pengguna.

Python juga memungkinkan pengembangan yang cepat dan fleksibel, sehingga sangat cocok untuk penelitian eksploratif yang mengharuskan pengujian banyak variasi algoritma. Hal ini menjadikan Python alat utama dalam pengembangan sistem rekomendasi berskala kecil hingga besar, serta dalam pengujian sistem akademik dan industri (García-Gil et al., 2020).

## 2.7 Google Colab

A logo with orange circles

AI-generated content may be incorrect.

Gambar 2.1 Logo Google Colab

Sumber :[googlecolab.com](https://colab.research.google.com/?hl=id)

Google Colaboratory, atau Google Colab, adalah dokumen eksekusi yang memungkinkan penulisan, pengeditan, dan berbagi program *Python* yang tersimpan di Google Drive atau yang baru dibuat, tanpa perlu menginstal perangkat lunak tambahan. Hanya membutuhkan akun Google untuk *login* dan menyimpan *file* Python. Google Colab cocok untuk *programmer* maupun pemula yang ingin belajar Python, serta mendukung kolaborasi tim seperti Google Docs. Mirip dengan Jupyter Notebook yang berbasis browser, Google Colab memungkinkan menjalankan *program* Python tanpa perlu instalasi atau unduhan editor teks, karena seluruh *setup* dilakukan di *cloud* (Anendya, 2023).

Salah satu keunggulan utama Google Colab adalah kemampuan kolaborasi *real-time*. Seperti halnya berbagi dokumen di Google Docs, pengguna dapat berbagi *notebook* mereka dengan orang lain untuk bekerja bersama secara simultan. *Notebook* ini disimpan di Google Drive, sehingga dapat diakses dari perangkat apa pun dan kapan saja, mirip dengan dokumen lain dalam ekosistem Google. Dalam konteks analisis data, Google Colab menyediakan akses ke berbagai *library* Python yang sudah terinstal sebelumnya, seperti TensorFlow, PyTorch, dan OpenCV.

## 2.8 Visual Studio Code

A blue triangle with a cross

AI-generated content may be incorrect.

Gambar 2.2 Logo Visual Studio Code

Sumber : [visualstudiocode.com](https://code.visualstudio.com/)

Visual Studio Code adalah editor kode sumber gratis yang ringan namun kuat, tersedia untuk Windows, macOS, Linux, Raspberry Pi OS, serta versi web. Editor ini mendukung JavaScript, TypeScript, dan Node.js secara bawaan, serta memiliki ekosistem ekstensi yang luas untuk bahasa pemrograman lain, *runtime*, lingkungan, dan *platform cloud*. Visual Studio Code juga menyediakan fitur seperti *IntelliSense* untuk melengkapi kode, *debugging* grafis, linting, pengeditan multi-kursor, petunjuk *parameter*, navigasi dan *refactoring* kode, serta kontrol versi bawaan termasuk dukungan Git. Banyak fitur ini diadaptasi dari teknologi Visual Studio (Heller, 2022).

## 2.9 Streamlit



Gambar 2.3 Logo Streamlit

Sumber : [streamlit.io](https://streamlit.io/)

Streamlit adalah kerangka kerja sumber terbuka untuk pembuatan prototipe cepat dan pembuatan visualisasi serta dasbor di Python tanpa memerlukan pengetahuan tentang teknologi *front-end* seperti JS atau HTML. Hanya dengan beberapa baris kode, Anda dapat membuat aplikasi yang dinamis dan menarik secara visual yang memungkinkan pengguna untuk memasukkan data dan melihat hasilnya secara *real-time*. Ketika Anda menjalankan aplikasi Streamlit, kode Python dieksekusi dan menghasilkan kode HTML, CSS, dan JavaScript yang diperlukan untuk menampilkan aplikasi di peramban *web* (Wawrzyński, 2023).

## 2.10 *Flowchart*

Flowchart atau diagram alir adalah diagram yang terdiri atas gambar dan

simbol untuk menggambarkan langkah-langkah penyelesaian suatu permasalahan

yang ada pada sistem untuk mempermudah pengguna dalam mema- hami informasi

dan rangkaian proses yang ada pada suatu prosedur. Simbol- simbol yang ada pada

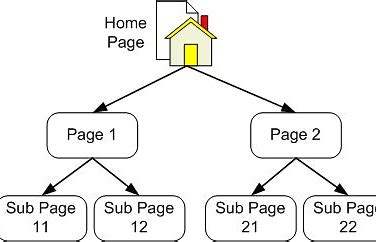
flowchart beserta keterangannya dapat dilihat pada gambar 2.3

A table of text and symbols

AI-generated content may be incorrect.

Gambar 2.4 Simbol dan keterangan flowchart

## 2.11 Struktur Navigasi



Gambar 2.5 Contoh Struktur Navigasi

Sumber : [medium.com](https://medium.com/@jkvsuga9/ux-blue-print-memanfaatkan-untuk-keunggulan-desain-61db058d2fa9)

Struktur navigasi website menggambarkan cara berbagai halaman di situs diatur dan saling terhubung. Sebelum membangun *website*, desainer *web* dan ahli *SEO* biasanya membuat *wireframe*, yang merupakan rancangan awal struktur situs. Struktur navigasi ini menentukan urutan klik pengguna, mengarahkan mereka dari satu halaman ke halaman berikutnya. Struktur navigasi yang baik harus mudah dipahami oleh pengguna dan tidak membingungkan. Selain itu, struktur navigasi mengorganisir konten secara logis dan hierarkis, sehingga pengguna memahami hubungan antara berbagai bagian dari situs atau aplikasi (Gunawan, 2021). Struktur ini juga memfasilitasi perpindahan antar halaman atau bagian yang berbeda dengan cepat dan efisien. Berikut adalah macam – macam struktur navigasi pada *website.*