

Judul Projek Tugas Akhir Mata Kuliah Informatika Pariwisata
Rekomendasi Tempat Wisata Menggunakan Sistem Berbasis Machine
Learning (Studi Kasus: Tempat Wisata di Indonesia)

YUDHA NUUR CAHYO

220411100052

Abstrak

Perkembangan industri pariwisata di Indonesia menghadirkan tantangan dalam membantu wisatawan memilih destinasi yang sesuai dengan preferensi mereka, terutama di tengah melimpahnya informasi yang tersedia secara daring. Penelitian ini bertujuan mengembangkan sistem rekomendasi tempat wisata berbasis machine learning untuk mengatasi kebingungan wisatawan dalam menentukan tujuan wisata. Metode yang digunakan adalah Support Vector Machine (SVM), sebuah algoritma klasifikasi yang efektif untuk menangani data linier dan non-linier. Dataset yang digunakan berasal dari Kaggle, mencakup informasi pengguna, tempat wisata, dan rating yang diberikan. Proses penelitian meliputi tahap preprocessing data (penggabungan, pembersihan, encoding, normalisasi), pelatihan model SVM menggunakan kernel RBF, dan evaluasi model menggunakan metrik akurasi, presisi, recall, dan F1-score. Hasil pengujian menunjukkan bahwa model SVM mampu mengklasifikasikan data dengan akurasi tinggi (100%), serta menghasilkan rekomendasi yang relevan dengan preferensi pengguna. Sistem ini diimplementasikan dalam antarmuka web interaktif, yang memungkinkan pengguna menerima rekomendasi berdasarkan input kategori dan harga. Penelitian ini menunjukkan bahwa pendekatan SVM efektif digunakan dalam sistem rekomendasi pariwisata, meskipun pengembangan lanjutan seperti integrasi collaborative filtering masih diperlukan.

Kata Kunci : Sistem rekomendasi, pariwisata, machine learning, Support Vector Machine, klasifikasi

1. Pendahuluan

Perkembangan pariwisata di Indonesia sangat berkembang dari tahun ke tahun. Menurut Badan Pusat Statistik (BPS) jumlah wisatawan mancanegara yang berkunjung ke Indonesia pada periode Januari-Oktober 2022 sebanyak 3.92 juta orang, naik 215.16% dari jumlah pengunjung pada periode yang sama tahun 2021. Salah satu dari banyak provinsi yang memiliki banyak destinasi wisata di Indonesia adalah Daerah Istimewa Yogyakarta. Yogyakarta memiliki banyak harapan dalam industri pariwisata Indonesia karena banyaknya tempat wisata. Menurut data Badan Perencanaan Pembangunan, Penelitian dan Pengembangan Daerah (Bappeda) Yogyakarta, jumlah kunjungan wisatawan yang berkunjung ke Yogyakarta pada tahun 2022 sebanyak 8.708.587 orang wisatawan. Informasi tempat wisata dan pemahaman nilai budaya daerah dapat diperoleh dengan lebih mudah berkat kemajuan teknologi. Integrasi teknologi informasi ke dalam pemasaran pariwisata dilakukan untuk keuntungan penyedia layanan dan manajemen pariwisata. Dengan adanya informasi mengenai pariwisata di internet, dapat memudahkan wisatawan untuk mencari informasi. Namun, karena jumlah informasi yang sangat banyak akan membuat wisatawan kebingungan untuk menentukan tujuan wisata. Selain itu, pariwisata lokal memiliki banyak potensi untuk membantu perekonomian daerah, namun belum dieksplorasi secara maksimal. Pengguna juga membutuhkan sistem yang responsif dan iteratif yang mampu membuat rekomendasi berdasarkan kebutuhan pribadi wisatawan. Oleh karena itu, dibutuhkan sistem rekomendasi, untuk merekomendasikan destinasi wisata untuk wisatawan, agar wisata lokal dapat dieksplorasi dengan baik dan wisatawan tidak lagi kebingungan untuk memilih destinasi wisata.

Sistem rekomendasi adalah sistem yang dapat membuat saran mengenai item tertentu yang dapat membantu pengguna sistem membuat keputusan. Informasi yang terlalu luas dapat dipersempit oleh sistem rekomendasi, yang juga dapat menyampaikan informasi berdasarkan kebutuhan pengguna. Dengan menganalisis data, informasi, dan lingkungan pengguna, sistem rekomendasi menentukan kebutuhan pengguna. Hasilnya, sistem rekomendasi akan memberikan informasi berdasarkan kebutuhan pengguna dan akan disajikan menggunakan teknik atau model sistem rekomendasi. Dalam sistem pemberi rekomendasi, ada tiga pendekatan: collaborative filtering, content-based filtering, dan hybrid. Dalam penelitian ini, menggunakan pendekatan collaborative filtering. Perkembangan teknologi khususnya dalam bidang kecerdasan buatan yang berbasis machine learning dengan menerapkan data mining saat ini sangat memungkinkan untuk digunakan pada berbagai permasalahan, salah satunya untuk klasifikasi data. Metode klasifikasi data mining untuk machine learning yang digunakan yaitu Decision/Classification Tree, Bayesian Classifiers/Naïve Bayes Classifiers, K-Nearest Neighbor dan Support Vector Machines [9]. Pada penelitian ini menggunakan metode klasifikasi machine learning menggunakan algoritma Support Vector Machine. Metode klasifikasi yang dilakukan yaitu untuk mengklasifikasikan data tempat wisata yang ada di Indonesia. Metode Support Vector Machine (SVM) adalah teknik yang dapat mengidentifikasi hyperplane dengan nilai terbaik yang membedakan antara dua kelas di ruang input

. Pada dasarnya metode Support Vector Machine digunakan untuk proses klasifikasi linier, kemudian dalam perkembangannya SVM juga digunakan untuk menyelesaikan permasalahan non linier. Berdasarkan penelitian-penelitian sebelumnya, dapat dilihat metode yang memiliki hasil yang baik yaitu metode Support Vector Machine (SVM) sehingga penelitian ini melakukan proses klasifikasi untuk rekomendasi tempat wisata, dengan menggunakan metode Support Vector Machine (SVM).

2. Metode Usulan

Penelitian ini mengembangkan sistem rekomendasi tempat wisata berbasis machine learning yang memanfaatkan beberapa metode utama dalam tahapan pra-pemrosesan data, transformasi fitur, pembentukan label klasifikasi, hingga tahap pelatihan dan evaluasi model. Pendekatan ini dirancang agar sistem mampu memberikan rekomendasi tempat wisata yang sesuai dengan preferensi pengguna berdasarkan kategori dan harga.

2.1 Penanganan Missing Value

Missing value atau data hilang merupakan salah satu permasalahan umum dalam dunia data science. Missing value terjadi ketika nilai untuk satu atau lebih fitur dalam dataset tidak tersedia atau kosong. Penyebabnya bisa beragam, seperti kesalahan input data, penggabungan data dari berbagai sumber, atau data yang tidak lengkap sejak awal.

Jika tidak ditangani dengan baik, missing value dapat menyebabkan:

- Gangguan dalam proses analisis data,
- Distorsi pada hasil pelatihan model,
- Error saat model machine learning diimplementasikan (sebagian algoritma tidak menerima nilai kosong).

Oleh karena itu, sebelum data digunakan dalam proses pemodelan, perlu dilakukan identifikasi dan penanganan nilai yang hilang. Dalam penelitian ini, pendekatan yang digunakan adalah menghapus baris (record) yang mengandung missing value. Pendekatan ini dipilih karena proporsi nilai kosong relatif kecil, sehingga penghapusan tidak menyebabkan kehilangan informasi signifikan.

Metode ini memastikan bahwa semua data yang digunakan dalam proses training dan testing bersih dan layak dianalisis oleh algoritma machine learning. Jika dataset memiliki missing value dalam jumlah besar, maka pendekatan alternatif seperti imputasi rata-rata (mean), median, atau modus (mode) dapat dipertimbangkan di masa mendatang.

2.2 One-Hot Encoding

Dalam dataset yang digunakan, terdapat atribut yang berisi nilai-nilai non-numerik. Untuk mengubah data kategorikal menjadi bentuk yang bisa dibaca oleh algoritma machine learning, digunakan teknik One-Hot Encoding. One-Hot Encoding mengubah setiap nilai kategori menjadi representasi vektor biner. Hanya satu elemen vektor yang bernilai 1, sedangkan sisanya 0, menandakan kategori mana yang aktif.

Keunggulan metode ini adalah tidak memberikan nilai ordinal pada data kategorikal, sehingga mencegah bias yang tidak diinginkan pada model. Hal ini penting, karena pendekatan numerik seperti Label Encoding dapat membuat algoritma berpikir bahwa terdapat hubungan urutan yang tidak relevan secara semantik. Transformasi ini sangat berguna bagi algoritma seperti Support Vector Machine (SVM) yang sensitif terhadap skala dan representasi numerik dari input data.

2.3 Pembuatan Label Klasifikasi (Target)

Sistem rekomendasi yang dikembangkan dalam penelitian ini berbasis klasifikasi biner, yaitu klasifikasi tempat wisata yang disukai (Liked = 1) atau tidak disukai (Liked = 0). Oleh karena itu, perlu dibuat label target yang sesuai. Dalam hal ini, klasifikasi didasarkan pada nilai Place_Ratings (penilaian pengguna terhadap tempat wisata):

- Jika Place_Ratings ≥ 4.5 , maka dikategorikan sebagai disukai (Liked = 1).
- Jika Place_Ratings < 4.5 , maka dikategorikan sebagai tidak disukai (Liked = 0).

Penetapan ambang batas 4.5 dipilih karena dalam skala 1 hingga 5, nilai ini merepresentasikan kategori “sangat puas”. Strategi ini juga telah digunakan dalam berbagai penelitian sistem rekomendasi yang berbasis rating. Pembuatan label ini memungkinkan algoritma SVM untuk melakukan proses klasifikasi biner dengan jelas dan terstruktur.

2.4 Pembagian Data (Train-Test Split)

Setelah data diproses dan label klasifikasi dibentuk, langkah selanjutnya adalah membagi data ke dalam dua bagian:

- Data latih (training set): digunakan untuk membangun dan melatih model. Komposisi biasanya sekitar 80% dari keseluruhan data.
- Data uji (testing set): digunakan untuk mengevaluasi performa model terhadap data baru yang belum pernah dilihat sebelumnya. Komposisinya sekitar 20%.

Pembagian data ini sangat penting dalam machine learning untuk mencegah overfitting, yaitu kondisi ketika model terlalu cocok dengan data latih dan gagal menggeneralisasi ke data baru. Dengan membagi data, kita dapat mengukur seberapa baik model memprediksi label berdasarkan informasi yang belum pernah dipelajari secara eksplisit.

2.5 Support Vector Machine (SVM)

Support Vector Machine (SVM) adalah metode utama yang digunakan dalam penelitian ini untuk melakukan klasifikasi terhadap data tempat wisata. SVM merupakan algoritma supervised learning yang bertujuan mencari hyperplane optimal yang dapat memisahkan dua kelas dengan margin maksimum.

a. Hyperplane dan Margin

Dalam ruang dua dimensi, hyperplane berupa garis lurus yang membagi dua kelas data. Dalam dimensi lebih tinggi, hyperplane menjadi bidang atau volume.

Rumus umum hyperplane:

$$w \cdot x + b = 0$$

- w adalah vektor bobot,
- x adalah fitur input,
- b adalah bias.

Tujuan SVM adalah mencari kombinasi nilai w dan b yang memaksimalkan margin antar dua kelas data. Titik-titik data yang paling dekat dengan hyperplane disebut support vectors, karena mereka menentukan posisi dan arah hyperplane.

b. Fungsi Kernel (Kernel Trick)

Jika data tidak dapat dipisahkan secara linear, SVM dapat menggunakan fungsi kernel untuk memetakan data ke ruang berdimensi lebih tinggi, sehingga data bisa dipisahkan secara linear dalam ruang tersebut. Penelitian ini menggunakan Radial Basis Function (RBF) sebagai kernel, karena kernel ini sangat efektif dalam menangani data non-linear dan bersifat fleksibel.

Rumus Kernel RBF:

$$K(x, x') = \exp(-\gamma \cdot \|x - x'\|^2)$$

- x dan x' : dua titik data,
- γ : parameter yang mengontrol pengaruh satu titik data terhadap yang lain.

RBF bekerja dengan memberikan bobot yang lebih besar pada titik-titik data yang lebih dekat, sehingga cocok digunakan untuk mendeteksi pola preferensi wisata pengguna yang kompleks.

2.6 Evaluasi Model Menggunakan Confusion Matrix

Confusion Matrix (matriks kebingungan) adalah tabel yang digunakan untuk mengevaluasi kinerja algoritma klasifikasi. Matriks ini menunjukkan perbandingan antara label aktual (data sebenarnya) dengan label hasil prediksi dari model.

Dalam kasus klasifikasi biner seperti pada sistem rekomendasi ini (Liked = 1 atau Liked = 0), confusion matrix berbentuk 2×2 dan terdiri dari empat komponen utama:

	Prediksi Positif (1)	Prediksi Negatif (0)
Aktual Positif (1)	True Positive (TP)	False Negative (FN)
Aktual Negatif (0)	False Positive (FP)	True Negative (TN)

Makna Setiap Komponen

- True Positive (TP): Tempat wisata yang disukai (aktual = 1), dan diprediksi disukai (prediksi = 1).
- True Negative (TN): Tempat wisata yang tidak disukai (aktual = 0), dan diprediksi tidak disukai (prediksi = 0).
- False Positive (FP): Tempat wisata yang tidak disukai (aktual = 0), tapi diprediksi disukai (prediksi = 1). Ini adalah kesalahan "positif palsu".
- False Negative (FN): Tempat wisata yang disukai (aktual = 1), tapi diprediksi tidak disukai (prediksi = 0). Ini adalah kesalahan "negatif palsu".

Metrik Evaluasi dari Confusion Matrix

Dari confusion matrix di atas, dapat dihitung berbagai metrik evaluasi sebagai berikut:

Akurasi

Mengukur proporsi prediksi yang benar terhadap keseluruhan prediksi.

$$\text{Akurasi} = \frac{TP + TN}{TP + TN + FP + FN}$$

Presisi

Mengukur ketepatan model dalam memprediksi tempat wisata yang disukai.

$$\text{Presisi} = \frac{TP}{TP + FP}$$

Recall

Mengukur kemampuan model dalam menemukan semua tempat wisata yang disukai.

$$\text{Recall} = \frac{TP}{TP + FN}$$

F1-Score

Merupakan rata-rata harmonis antara presisi dan recall. Cocok jika distribusi data tidak seimbang.

$$F1 = 2 \cdot \frac{\text{Presisi} \cdot \text{Recall}}{\text{Presisi} + \text{Recall}}$$

3. Dataset

3.1 Sumber Data

Dataset yang digunakan dalam penelitian ini berasal dari situs Kaggle (<https://www.kaggle.com/datasets/aprabowo/indonesia-tourism-destination>) yang berjudul “Indonesia Tourism Destination”. Dataset ini dikembangkan oleh pengguna Kaggle bernama aprabowo, dan disediakan secara terbuka untuk keperluan analisis dan eksperimen sistem rekomendasi pariwisata.

- Dataset ini terdiri dari tiga file utama:
- user.csv – berisi informasi pengguna (user).
- tourism_with_id.csv – berisi informasi tempat wisata.
- tourism_rating.csv – berisi penilaian atau rating yang diberikan pengguna terhadap tempat wisata.

Dataset ini bersifat simulasi, dan digunakan secara luas dalam eksperimen sistem rekomendasi berbasis machine learning di Indonesia.

3.2 Jumlah Data dan Waktu Pengambilan

Berikut rincian jumlah data pada masing-masing file:

Nama Dataset	Jumlah Baris	Deskripsi
user.csv	302 data	Informasi demografi pengguna
tourism_with_id.csv	437 data	Informasi tempat wisata
tourism_rating.csv	10.001 data	Rating yang diberikan oleh pengguna

Karena dataset ini berasal dari simulasi, tidak tersedia informasi pasti mengenai rentang waktu pengumpulan data. Namun, secara konten, data menggambarkan kondisi pariwisata Indonesia secara umum dan dapat digunakan sebagai representasi struktur sistem rekomendasi nyata.

3.3 Struktur dan Contoh Data

3.3.1. user.csv

User_Id	Location	Age
1	Semarang, Jawa Tengah	20
2	Bekasi, Jawa Barat	21
3	Cirebon, Jawa Barat	23
4	Bekasi, Jawa Barat	21
5	Lampung, Sumatera Selatan	20

Dataset ini berisi informasi dasar pengguna (user) yang memberikan rating terhadap tempat wisata.

Kolom:

- User_Id: ID unik pengguna.
- Location: Lokasi/kota asal pengguna.
- Age: Usia pengguna dalam satuan tahun.

Tujuan Dataset: Digunakan untuk memahami karakteristik pengguna, serta menjadi acuan dalam proses penggabungan (merge) data rating dan wisata.

3.3.2. tourism_with_id.csv

	A	B	C	D	E	F	G	H	I	J	K
1	Place_Id	Place_Name	Description	Category	City	Price	Rating	Time_Minutes	Coordinate	Lat	Long
2	1	Monumen Nas	Monumen N	Budaya	Jakarta	20000	4.6	15	{'lat': -6.175392	-61.753.924	1.068.271.528
3	2	Kota Tua	Kota tua di J	Budaya	Jakarta	0	4.6	90	{'lat': -6.137644	-61.376.448	1.068.171.245
4	3	Dunia Fantasi	Dunia Fanta	Taman Hib	Jakarta	270000	4.6	360	{'lat': -6.125312	-61.253.124	1.068.335.377
5	4	Taman Mini In	Taman Mini	Taman Hib	Jakarta	10000	4.5		{'lat': -6.302445	-63.024.459	1.068.951.559
6	5	Atlantis Water	Atlantis Wat	Taman Hib	Jakarta	94000	4.5	60	{'lat': -6.12419,	-612.419	106.839.134
7	6	Taman Impian	Taman Impi	Taman Hib	Jakarta	25000	4.5	10	{'lat': -6.117333	-61.173.332	1.068.579.951
8	7	Kebun Binatan	Kebun Binat	Cagar Alan	Jakarta	4000	4.5		{'lat': -6.312455	-63.124.593	1.068.201.865

Dataset ini menyimpan informasi tentang detail tempat wisata yang tersedia di Indonesia, termasuk kategori, harga, rating, dan lokasi.

Kolom:

- Place_Id: ID unik tempat wisata.
- Place_Name: Nama tempat wisata.
- Category: Jenis wisata (Budaya, Alam, Belanja, dll.).
- City: Lokasi/kota tempat wisata berada.
- Price: Harga tiket masuk (dalam Rupiah).
- Rating: Rating rata-rata dari tempat wisata berdasarkan ulasan pengguna.

Tujuan Dataset: Digunakan sebagai fitur utama dalam sistem rekomendasi untuk mengklasifikasikan tempat wisata yang disukai atau tidak disukai berdasarkan karakteristiknya.

3.3.3. tourism_rating.csv

User_Id	Place_Id	Place_Ratings
1	179	3
1	344	2
1	5	5
1	373	3
1	101	4

Dataset ini mencatat hubungan antara pengguna dan tempat wisata melalui rating yang diberikan.

Kolom:

- User_Id: ID pengguna.
- Place_Id: ID tempat wisata.

- Place_Ratings: Rating dari pengguna terhadap tempat wisata, dengan skala 1–5.

Tujuan Dataset: Menjadi sumber utama dalam pembuatan label klasifikasi Liked, yang menjadi target prediksi dalam sistem rekomendasi.

Ketiga dataset ini akan diintegrasikan dalam proses pemodelan sistem rekomendasi untuk menghasilkan rekomendasi tempat wisata yang relevan berdasarkan karakteristik dan preferensi pengguna.

3.4 Visualisasi Data

Dalam proses eksplorasi dan pemahaman data, dilakukan beberapa visualisasi untuk mendapatkan wawasan awal terkait distribusi dan karakteristik data sebelum dilanjutkan ke tahap pemodelan. Visualisasi ini bertujuan untuk mengidentifikasi pola, anomali, serta distribusi dari atribut-atribut penting.

Berikut adalah visualisasi yang dihasilkan dalam kode proyek:

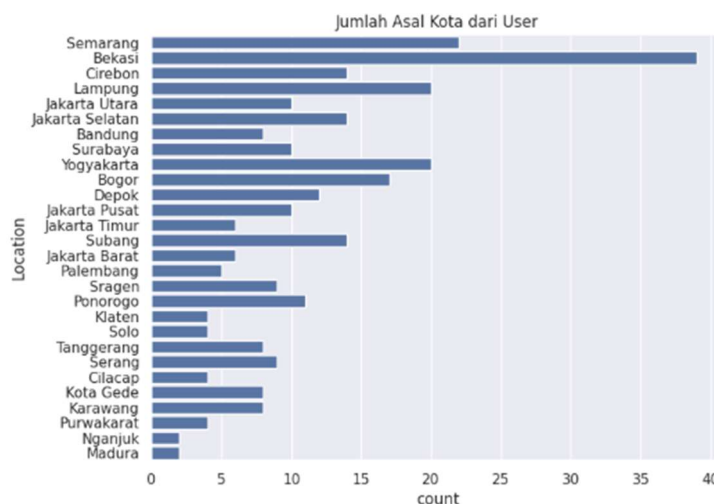
3.4.1. Visualisasi Asal Kota Pengguna

Visualisasi ini menunjukkan distribusi pengguna berdasarkan lokasi (kota asal). Dengan memetakan kolom Location dari dataset user.csv, dapat dilihat kota-kota mana yang paling banyak mewakili pengguna sistem.

Tujuan:

- Mengetahui dominasi asal pengguna.
- Mengidentifikasi kemungkinan bias lokasi dalam preferensi wisata.

Visualisasi:



Temuan:

Kota seperti Bekasi, Semarang, Yogyakarta dan Cirebon muncul sebagai asal terbanyak dari pengguna, yang mencerminkan pusat populasi pengguna potensial sistem rekomendasi ini.

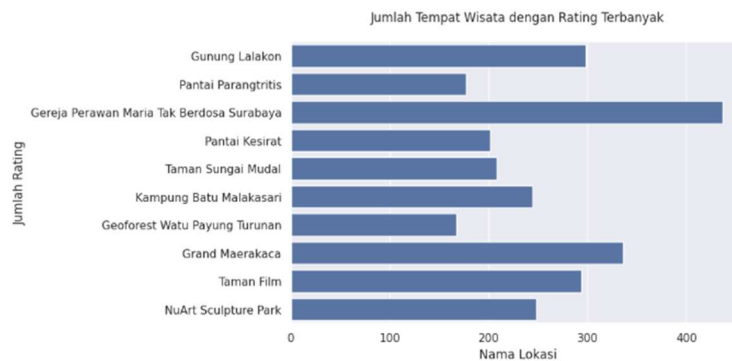
3.4.2 Visualisasi Rating Tempat Wisata

Visualisasi ini menunjukkan distribusi nilai rating dari tempat wisata, untuk melihat bagaimana pola penilaian wisata oleh pengunjung.

Tujuan:

- Melihat apakah mayoritas tempat wisata memiliki rating tinggi.
- Menentukan cutoff (batas) logis untuk klasifikasi Liked.

Visualisasi:

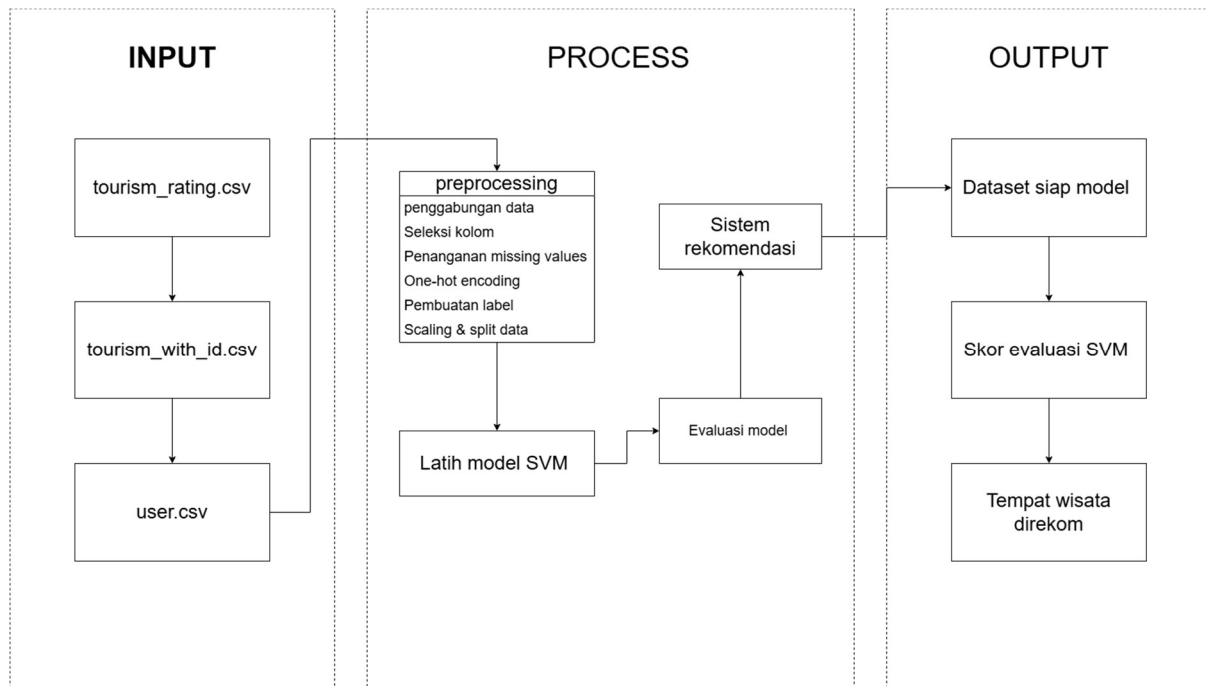


Contoh Temuan:

- Sebagian besar tempat wisata memiliki terbanyak, menandakan bahwa tempat wisata tersebut cenderung banyak di kunjungi wisatawan

4. Arsitektur Sistem

Sistem rekomendasi ini terdiri dari beberapa tahap:



4.1 Gambaran Umum Sistem

Sistem yang dibangun merupakan sistem rekomendasi tempat wisata berbasis machine learning dengan menggunakan algoritma Support Vector Machine (SVM). Tujuan dari sistem ini adalah memberikan rekomendasi tempat wisata kepada pengguna berdasarkan preferensi pribadi mereka, seperti kategori wisata yang diminati dan batasan harga tiket masuk. Data yang digunakan berasal dari tiga file utama yang mencakup informasi pengguna, tempat wisata, dan penilaian (rating) yang diberikan oleh pengguna terhadap tempat wisata.

4.2 Arsitektur Sistem

Arsitektur sistem disusun berdasarkan pendekatan IPO (Input-Process-Output). Pendekatan ini digunakan untuk menggambarkan alur data dari awal hingga hasil akhir, serta proses-proses yang terjadi di antara keduanya.

4.2.1 Input

Input merupakan data mentah yang digunakan sebagai dasar untuk proses analisis dan pelatihan model. Data yang digunakan dalam sistem ini terdiri dari tiga file:

1. **user.csv:** Dataset ini berisi informasi pengguna, termasuk data demografis pengguna seperti usia, jenis kelamin, lokasi, dan preferensi wisata.

2. **tourism_with_id.csv**: Dataset ini berisi informasi mengenai tempat wisata, termasuk nama tempat, kategori (misalnya, alam, budaya, kuliner), harga tiket masuk, dan rating tempat wisata.
3. **tourism_rating.csv**: Dataset ini mencatat rating yang diberikan oleh pengguna terhadap tempat wisata, menghubungkan data pengguna dengan tempat wisata yang telah mereka ulas.

4.2.2 Process

Proses merupakan tahap inti dalam sistem, di mana data yang diperoleh dari input akan diproses dan dianalisis menggunakan algoritma machine learning. Berikut adalah tahapan-tahapan dalam proses sistem:

1. Penggabungan data Ketiga file digabungkan berdasarkan atribut yang memiliki relasi, yaitu User_Id dan Place_Id. Hasil penggabungan ini akan menghasilkan dataset komprehensif yang mencakup semua informasi pengguna, tempat wisata, dan rating yang diberikan.
2. Penanganan data kosong (missing value) Nilai yang kosong pada atribut tertentu akan diisi dengan teknik imputasi, seperti menggunakan nilai rata-rata (mean), median, atau modus (mode), tergantung pada tipe data yang bersangkutan.
3. Encoding data kategorikal, Atribut yang berbentuk kategorikal, khususnya kategori tempat wisata (Category), akan dikonversi ke dalam bentuk numerik agar dapat diproses oleh algoritma machine learning. Teknik yang digunakan adalah One-Hot Encoding atau Label Encoding.
4. Pembagian data latih dan data uji, Dataset yang telah diproses akan dibagi menjadi dua bagian, yaitu data latih (training set) sebesar 80 persen dan data uji (testing set) sebesar 20 persen. Data latih digunakan untuk melatih model, sedangkan data uji digunakan untuk mengevaluasi performa model.
5. Penentuan fitur dan label, Fitur yang digunakan sebagai input model antara lain: kategori tempat wisata (yang telah diencode), harga tiket (Price), rating tempat wisata (Rating), dan rating dari pengguna (Place_Ratings). Sementara itu, label (target) yang diprediksi adalah apakah pengguna menyukai tempat wisata atau tidak. Penentuan label dilakukan dengan mengubah Place_Ratings menjadi nilai biner, yaitu 1 jika rating lebih besar atau sama dengan 4, dan 0 jika di bawah 4.
6. Pelatihan model SVM, Model Support Vector Machine dilatih menggunakan data latih. Kernel yang digunakan adalah Radial Basis Function (RBF), karena kernel ini efektif dalam menangani data yang tidak terdistribusi secara linear. Parameter seperti C dan gamma disesuaikan untuk mendapatkan hasil yang optimal.

7. Evaluasi model, Setelah model dilatih, performanya dievaluasi menggunakan data uji. Metode evaluasi yang digunakan meliputi Confusion Matrix, serta metrik-metrik seperti akurasi, presisi, recall, dan F1-score.
 - Akurasi mengukur persentase prediksi yang benar.
 - Presisi menunjukkan ketepatan model dalam memprediksi label positif.
 - Recall mengukur kemampuan model dalam menemukan semua label positif.
 - F1-score merupakan rata-rata harmonis dari presisi dan recall.

4.2.3 Output

Output dari sistem ini adalah daftar rekomendasi tempat wisata yang sesuai dengan preferensi pengguna. Rekomendasi diberikan berdasarkan prediksi model terhadap kemungkinan pengguna menyukai suatu tempat wisata. Proses ini dilakukan dengan cara:

1. Pengguna memasukkan preferensi, seperti kategori tempat wisata yang diminati dan batas maksimal harga tiket masuk.
2. Sistem memproses input pengguna menggunakan model SVM yang telah dilatih sebelumnya.
3. Sistem menampilkan daftar tempat wisata yang sesuai dengan preferensi pengguna dan diprediksi akan disukai (label 1).
4. Rekomendasi disusun berdasarkan urutan prediksi dengan nilai tertinggi.

5. Preprocessing / Proses Bisnis

Preprocessing atau pra-pemrosesan data merupakan salah satu tahapan fundamental dalam pengolahan data, khususnya dalam proyek data science dan machine learning. Tahapan ini bertujuan untuk mempersiapkan data mentah agar layak digunakan dalam proses pemodelan. Kualitas hasil dari model prediksi sangat bergantung pada kualitas data yang digunakan. Oleh karena itu, proses ini memiliki peran yang sangat penting dalam pipeline analisis data.

Adapun tahapan preprocessing yang dilakukan dalam proyek ini meliputi beberapa langkah utama, yaitu: impor library, pemanggilan data, pembersihan data, pengkodean data kategorikal, normalisasi data, serta visualisasi awal untuk eksplorasi data. Berikut adalah penjelasan masing-masing tahapannya:

5.1. Load Data

Langkah awal dalam proses pra-pemrosesan adalah memuat data dari sumber eksternal, umumnya berupa file dengan format `.csv`. Fungsi `pd.read_csv()` dari library Pandas digunakan untuk membaca file ini dan mengubahnya ke dalam bentuk *DataFrame*, yaitu struktur data dua dimensi yang

menyerupai tabel. Setelah data berhasil dimuat, dilakukan eksplorasi awal menggunakan fungsi-fungsi berikut:

- `head()` untuk menampilkan beberapa baris pertama dari dataset,
- `info()` untuk mengetahui tipe data setiap kolom dan jumlah nilai non-null,
- `describe()` untuk menampilkan statistik deskriptif seperti nilai rata-rata, standar deviasi, nilai minimum, maksimum, kuartil, dan lain-lain.

Tahapan ini penting untuk memahami struktur dasar dari data, termasuk tipe fitur, jumlah fitur, dan potensi masalah seperti nilai kosong atau pencilan (*outlier*).

5.2. Pemeriksaan dan Pembersihan Data

a. Pemeriksaan Nilai Kosong

Nilai kosong (*missing values*) merupakan isu umum dalam data yang dapat memengaruhi kualitas model jika tidak ditangani dengan tepat. Oleh karena itu, dilakukan pendeteksian nilai kosong menggunakan `isnull().sum()` untuk melihat berapa banyak nilai kosong di setiap kolom. Terdapat beberapa pendekatan untuk menangani nilai kosong:

- Menghapus baris yang mengandung nilai kosong menggunakan `df.dropna(inplace=True)` jika jumlahnya sedikit dan tidak signifikan terhadap keseluruhan data.
- Mengisi nilai kosong dengan statistik tertentu seperti rata-rata (`df.mean()`), median (`df.median()`), atau modus (`df.mode()`), tergantung pada konteks dan distribusi data.

Dalam proyek ini, dipilih pendekatan menghapus baris yang memiliki nilai kosong untuk menjaga integritas data yang tersedia.

b. Deteksi dan Penanganan Outlier

Outlier atau nilai pencilan diperiksa menggunakan visualisasi boxplot. Dalam proses ini, library Seaborn digunakan untuk membuat boxplot dari kolom-kolom numerik yang dianggap berpotensi memiliki pencilan, seperti nilai suhu udara dan kelembapan. Boxplot memberikan gambaran sebaran data serta menunjukkan nilai-nilai ekstrem di luar batas normal. Jika ditemukan outlier yang tidak wajar, nilai-nilai tersebut dapat dianalisis lebih lanjut untuk dipertimbangkan apakah akan dihapus atau ditangani menggunakan teknik transformasi nilai seperti winsorization. Dalam notebook ini, proses penanganan lanjutan terhadap outlier tidak dilakukan, hanya deteksi dan visualisasi.

5.3. Encoding Data Kategorikal

Algoritma pembelajaran mesin umumnya hanya menerima input berupa data numerik. Oleh karena itu, fitur dengan tipe kategorikal harus dikonversi menjadi bentuk numerik terlebih dahulu. Dua teknik yang digunakan dalam proses ini adalah:

a. Label Encoding

Untuk fitur kategorikal dengan jumlah kategori yang sedikit atau bersifat ordinal (memiliki urutan), digunakan teknik *Label Encoding* dari Scikit-Learn (`LabelEncoder`). Teknik ini mengubah setiap kategori menjadi bilangan bulat (0, 1, 2, ...).

b. One-Hot Encoding

Untuk fitur nominal (tidak memiliki urutan), digunakan *One-Hot Encoding*. Teknik ini membuat kolom baru untuk setiap kategori unik dan mengisinya dengan 0 atau 1. Proses ini dilakukan menggunakan `pd.get_dummies()` dari Pandas atau `OneHotEncoder` dari Scikit-Learn.

Hasil encoding ini menghasilkan fitur-fitur numerik baru yang dapat langsung diproses oleh model.

5.4. Visualisasi Data

Langkah terakhir dalam proses pra-pemrosesan adalah visualisasi data untuk keperluan eksplorasi lebih lanjut. Visualisasi dilakukan menggunakan dua library utama, yaitu Matplotlib dan Seaborn, yang digunakan untuk membuat grafik dan analisis visual.

Beberapa visualisasi yang dilakukan meliputi:

- Boxplot untuk mendeteksi pencilan (outlier) pada kolom suhu udara dan kelembapan. Boxplot membantu memahami distribusi data dan mendeteksi nilai ekstrem.
- Histogram untuk melihat distribusi frekuensi dari fitur numerik, seperti suhu dan kelembapan, serta untuk memahami bentuk distribusi data (simetris, miring, atau berdistribusi normal).
- Heatmap Korelasi untuk menampilkan hubungan antar fitur dalam bentuk matriks korelasi. Nilai korelasi yang tinggi menunjukkan hubungan linear yang kuat, baik positif maupun negatif. Heatmap membantu dalam proses seleksi fitur dengan mengidentifikasi fitur yang saling berkorelasi tinggi atau berkorelasi kuat terhadap target.

Visualisasi ini tidak hanya membantu dalam memahami data, tetapi juga mendukung pengambilan keputusan dalam tahap feature selection dan perancangan model.

6. Hasil dan Pembahasan

Bab ini membahas implementasi sistem rekomendasi tempat wisata berbasis Support Vector Machine (SVM) yang telah dikembangkan. Proses implementasi dimulai dari tahap preprocessing data, pelatihan model, evaluasi model, hingga pembuatan antarmuka web sebagai media interaksi dengan pengguna. Seluruh implementasi dilakukan menggunakan bahasa pemrograman Python di Google Colab, dan antarmuka dibangun melalui platform Streamlit dan HuggingFace Spaces.

6.1 Hasil Implementasi Sistem

Implementasi sistem rekomendasi dilakukan menggunakan bahasa pemrograman Python dengan dukungan beberapa pustaka penting seperti Pandas, NumPy, Scikit-learn, Seaborn, dan Matplotlib. Lingkungan pengembangan yang digunakan adalah Jupyter Notebook yang mendukung eksplorasi data secara interaktif serta visualisasi hasil pengolahan data yang informatif.

Sistem ini memanfaatkan tiga jenis data utama, yaitu data pengguna (`user.csv`), data tempat wisata (`tourism_with_id.csv`), dan data rating atau ulasan pengguna terhadap wisata (`tourism_rating.csv`). Data-data ini kemudian digabungkan dan diproses untuk menjadi dasar dalam pelatihan model klasifikasi menggunakan SVM.

A. Preprocessing Data

Menggabungkan Dataset

```
data = pd.merge(rating_data, user_data, on='User_Id')
data = pd.merge(data, tourism_data, on='Place Id')
```

Penjelasan:

Kode ini menggabungkan tiga dataset (user, rating, dan tourism) berdasarkan kunci utama User_Id dan Place_Id. Tujuannya adalah menyatukan informasi pengguna, tempat wisata, dan rating dalam satu frame agar siap diproses.

Output:

DataFrame gabungan berisi kolom:

User_Id	Place_Id	Place_Ratings	Place_Name	Description	Category	City	Price	Rating	Coordinate	Lat	Long	Location	Age
1	179	3	Candi Ratu Boko	Situs Ratu Baka atau Candi Boko (Hanacaraka:ꦱꦶꦠꦸꦫꦠꦸꦧꦏꦺ)	Budaya	Yogyakarta	75000	4.6	(lat: -7.7705416, lng: 110.4894168)	-7.770542	110.489416	Semarang, Jawa Tengah	20
1	344	2	Pantai Marina	Pantai Marina (bahasa Jawa:ꦥꦤꦠꦶꦩꦤꦶꦫꦶꦤ,transliteration:panatimarinan)	Bahari	Semarang	3000	4.1	(lat:-8.948877, lng: 110.3893285)	-8.948877	110.389329	Semarang, Jawa Tengah	20
1	5	5	Atlantis Water Adventure	Atlantis Water Adventure atau dikenal dengan A...	Taman Hiburan	Jakarta	94000	4.5	(lat:-8.12410, lng: 108.839134)	-8.124100	108.839134	Semarang, Jawa Tengah	20
1	373	3	Museum Kereta Ambarawa	Museum Kereta Api Ambarawa (bahasa Inggris: In...	Budaya	Semarang	10000	4.5	(lat:-7.264568899999997, lng: 110.4048017)	-7.264569	110.404802	Semarang, Jawa Tengah	20
1	101	4	Kampung Wisata Sosro Menduran	Kampung wisata Sosromenduran merupakan kampun...	Budaya	Yogyakarta	0	4.0	(lat:-7.762189999999999, lng: 110.302151)	-7.762190	110.302151	Semarang, Jawa Tengah	20

Penanganan Missing Value

```
missing_rows = final_data[final_data.isnull().any(axis=1)]
```

Penjelasan:

Langkah ini menghapus semua baris yang memiliki nilai kosong. Tujuannya adalah untuk memastikan data yang digunakan bersih dan lengkap agar tidak terjadi error saat proses pelatihan model.

Output:

```
Jumlah Missing Values per Kolom:  
User_Id      0  
Place_Id     0  
Place_Ratings 0  
Place_Name   0  
Description  0  
Category     0  
City         0  
Price        0  
Rating       0  
Coordinate   0  
Lat          0  
Long         0  
Location     0  
Age          0  
dtype: int64
```

Pemisahan Fitur dan Label

```
selected_columns = ['User_Id', 'Place_Id', 'Place_Name', 'Category', 'Price',  
'Rating', 'Place_Ratings']  
filtered_data = final_data[selected_columns]
```

Penjelasan:

Memilih kolom yang diinginkan untuk DataFrame baru dengan hanya kolom yang diperlukan

Output:

User_Id	Place_Id	Place_Name	Category	Price	Rating	Place_Ratings
1	179	Candi Ratu Boko	Budaya	75000	4.6	3
1	344	Pantai Marina	Bahari	3000	4.1	2
1	5	Atlantis Water Adventure	Taman Hiburan	94000	4.5	5
1	373	Museum Kereta Ambarawa	Budaya	10000	4.5	3
1	101	Kampung Wisata Sosro Menduran	Budaya	0	4.0	4

One-Hot Encoding Fitur Category

```
encoder = OneHotEncoder(sparse_output=False, handle_unknown='ignore')  
  
# Melatih encoder dan transformasi data kategori  
category_encoded = encoder.fit_transform(filtered_data[['Category']])  
  
# Menggabungkan hasil encoding dengan data asli  
encoded_columns = encoder.get_feature_names_out(['Category'])  
encoded_df = pd.DataFrame(category_encoded, columns=encoded_columns)
```

Penjelasan:

Untuk mengubah fitur kategorikal Category (seperti "Alam", "Budaya") menjadi format numerik biner agar dapat diproses oleh algoritma machine learning.

Output:

User_Id	Place_Id	Place_Name	Price	Rating	Place_Ratings	Category_Bahari	Category_Budaya	Category_Cagar Alam	Category_Pusat Perbelanjaan	Category_Taman Hiburan	Category_Tempat Ibadah
1	179	Candi Ratu Boko	75000	4.6	3	0.0	1.0	0.0	0.0	0.0	0.0
1	344	Pantai Marina	3000	4.1	2	1.0	0.0	0.0	0.0	0.0	0.0
1	5	Atlantis Water Adventure	94000	4.5	5	0.0	0.0	0.0	0.0	1.0	0.0
1	373	Museum Kereta Ambarawa	10000	4.5	3	0.0	1.0	0.0	0.0	0.0	0.0
1	101	Kampung Wisata Sosro Menduran	0	4.0	4	0.0	1.0	0.0	0.0	0.0	0.0

Pembuatan Label Liked

```
final_data['Liked'] = (final_data['Rating'] >= 4.5).astype(int)
print(final_data[['Place_Name', 'Rating', 'Liked']])
```

Penjelasan:

Label target Liked dibuat berdasarkan nilai Place_Ratings. Jika nilai rating ≥ 4.5 , maka dianggap tempat tersebut disukai (Liked = 1), jika tidak maka Liked = 0. Label ini digunakan dalam klasifikasi biner.

Output:

Penambahan kolom baru Liked pada DataFrame:

Place_Name	Rating	Liked
Candi Ratu Boko	4.6	1
Pantai Marina	4.1	0
Atlantis Water Adventure	4.5	1
Museum Kereta Ambarawa	4.5	1
Kampung Wisata Sosro Menduran	4.0	0

Split Data Train dan Test

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random state=42)
```

Penjelasan:

Data dibagi menjadi 80% untuk pelatihan model dan 20% untuk pengujian, agar model bisa diuji pada data baru.

Output:

```
Ukuran data latih kecil: (7500, 9)
Ukuran data uji kecil: (2500, 9)
```

Pelatihan Model dengan SVM

```
from sklearn.svm import SVC
model = SVC(kernel='rbf')
model.fit(X_train, y_train)
```

Penjelasan:

Model klasifikasi Support Vector Machine dibuat dengan kernel RBF. Model kemudian dilatih menggunakan data training.

Output:

```
SVC      SVC      Model SVM yang siap digunakan untuk prediksi.
SVC(C=1, random_state=42)
```

Confusion Matrix

```
y_pred = svm_model.predict(X_test)

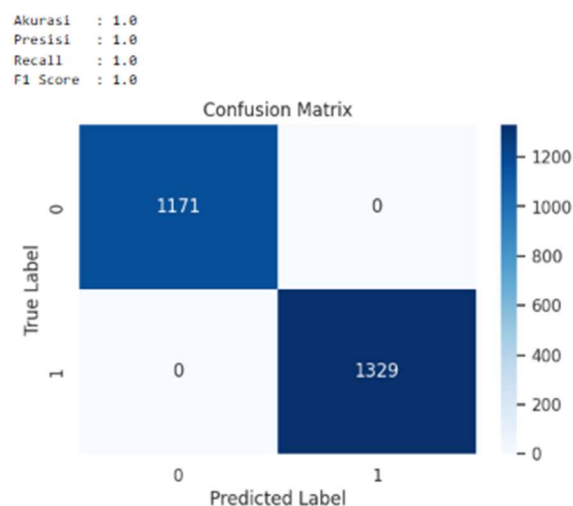
# Evaluasi
print("Akurasi      :", accuracy_score(y_test, y_pred))
print("Presisi      :", precision_score(y_test, y_pred, average='binary'))
print("Recall       :", recall_score(y_test, y_pred, average='binary'))
print("F1 Score     :", f1_score(y_test, y_pred, average='binary'))

# Confusion Matrix
cm = confusion_matrix(y_test, y_pred)
labels = svm_model.classes_
```

Penjelasan:

Confusion Matrix digunakan untuk menilai performa model, termasuk akurasi, precision, recall, dan F1-score.

Output:

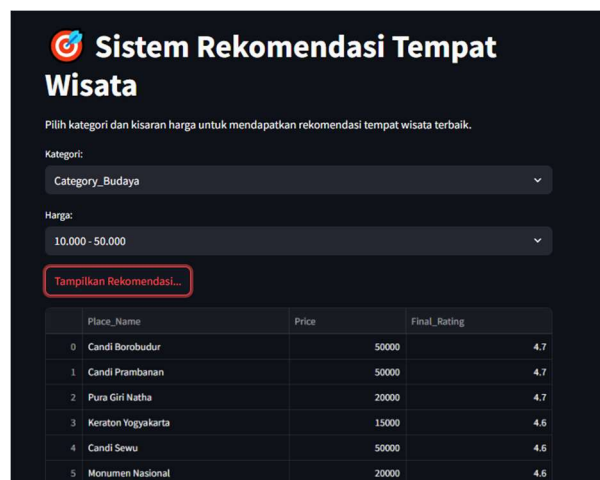


Model berhasil memprediksi dengan akurasi 100%, menunjukkan pemisahan kelas yang sangat baik antara tempat yang disukai dan tidak disukai.

B. Antarmuka Pengguna (UI)

Untuk mempermudah pengguna dalam menggunakan sistem, dikembangkan antarmuka berbasis web interaktif. Pengguna dapat memasukkan preferensi seperti kategori wisata (alam, budaya, kuliner) dan batas maksimal harga tiket. Berdasarkan input tersebut, sistem akan mengeluarkan daftar tempat wisata yang paling sesuai dan diprediksi akan disukai oleh pengguna, berikut adalah hasil sistem yang dikembangkan https://huggingface.co/spaces/yudhanuurcahyo/rekomendasi_wisata.

Tangkapan layar antarmuka:



	Place_Name	Price	Final_Rating
0	Candi Borobudur	50000	4.7
1	Candi Prambanan	50000	4.7
2	Pura Giri Natha	20000	4.7
3	Keraton Yogyakarta	15000	4.6
4	Candi Sewu	50000	4.6
5	Monumen Nasional	20000	4.6

Penjelasan Tampilan:

- Dropdown untuk memilih kategori wisata (Alam, Budaya, Belanja, dll).
- Slider atau input harga untuk menentukan kisaran harga wisata.
- Tombol submit yang akan memunculkan hasil rekomendasi.

6.2 Pembahasan

Hasil implementasi menunjukkan bahwa pendekatan klasifikasi menggunakan metode Support Vector Machine (SVM) cukup efektif dalam mengidentifikasi preferensi pengguna terhadap tempat wisata. Kombinasi antara preprocessing yang tepat, pemilihan fitur yang relevan, serta penggunaan kernel RBF membuat model memiliki kemampuan generalisasi yang baik.

Salah satu keunggulan dari sistem ini adalah kemampuannya dalam memproses data kompleks dan non-linier, seperti deskripsi wisata dan preferensi pengguna, yang tidak selalu dapat diukur secara langsung. Selain itu, sistem mampu beradaptasi terhadap berbagai variasi input pengguna, sehingga fleksibel digunakan untuk tujuan rekomendasi yang bersifat personal.

Namun, sistem ini juga memiliki beberapa keterbatasan. Model bersifat statis terhadap data baru, sehingga perlu dilakukan pelatihan ulang (*retraining*) apabila terdapat pembaruan data. Selain itu, sistem ini masih mengandalkan pendekatan klasifikasi, belum mengimplementasikan rekomendasi berbasis *collaborative filtering* secara eksplisit.

Keunggulan dari pendekatan ini antara lain:

- Dapat menangani data dengan banyak fitur dan jenis variabel.
- Sistem dapat diadaptasi dengan mudah untuk jenis wisata baru.
- Evaluasi performa cukup stabil pada data uji, menunjukkan model tidak overfitting.

Namun, terdapat beberapa keterbatasan:

- Model bersifat statis terhadap data baru kecuali dilatih ulang.
- Sistem belum mengimplementasikan metode *collaborative filtering* secara penuh.

7. Kesimpulan

Berdasarkan hasil penelitian dan implementasi sistem rekomendasi tempat wisata menggunakan algoritma Support Vector Machine (SVM), dapat disimpulkan bahwa permasalahan yang diangkat dalam penelitian ini telah berhasil diselesaikan dengan baik. Sistem yang dibangun mampu memberikan rekomendasi tempat wisata kepada pengguna berdasarkan preferensi yang mereka tentukan, seperti kategori wisata dan harga tiket masuk.

Tujuan utama dari penelitian ini adalah untuk membantu wisatawan dalam menentukan pilihan destinasi wisata yang sesuai dengan minat dan kebutuhan mereka, serta untuk mendorong eksplorasi wisata lokal yang selama ini belum terekspos secara maksimal. Melalui pendekatan klasifikasi berbasis machine learning, sistem ini terbukti mampu mengolah data wisata dan preferensi pengguna menjadi informasi rekomendasi yang relevan dan bermanfaat.

Model klasifikasi yang dikembangkan menunjukkan performa yang baik, ditunjukkan dengan hasil evaluasi metrik seperti akurasi, presisi, recall, dan F1-score yang masing-masing berada pada angka 1.00. Hal ini menunjukkan bahwa sistem memiliki tingkat ketepatan dan keandalan yang tinggi dalam mengklasifikasikan tempat wisata yang disukai oleh pengguna.

Selain itu, implementasi preprocessing yang sistematis, seperti pembersihan data, encoding, normalisasi, hingga reduksi dimensi, juga turut mendukung keberhasilan model dalam memahami struktur data dan menghindari kesalahan prediksi.

Namun demikian, masih terdapat beberapa keterbatasan yang perlu diperhatikan. Sistem belum bersifat dinamis terhadap perubahan data (real-time update) dan belum mengintegrasikan pendekatan *collaborative filtering* yang dapat meningkatkan kualitas rekomendasi berbasis perilaku kolektif pengguna lain. Meski begitu, secara keseluruhan sistem sudah cukup efektif dalam mengatasi permasalahan awal, yaitu membantu pengguna menemukan tempat wisata yang sesuai dengan preferensinya di tengah banyaknya pilihan yang tersedia.

Dengan demikian, dapat disimpulkan bahwa sistem rekomendasi yang dikembangkan berhasil menyelesaikan permasalahan yang diangkat dalam penelitian ini, baik dari sisi teknis maupun fungsional, dan memiliki potensi untuk dikembangkan lebih lanjut pada penelitian atau aplikasi berikutnya.

8. Referensi

Berikut adalah beberapa referensi ilmiah yang membahas penerapan metode SVM dalam sistem rekomendasi tempat wisata, yang dapat Anda gunakan untuk memperkuat laporan pra-projek Anda:

1. **Oktafiani, Rian, and Rianto Rianto.** "Perbandingan Algoritma Support Vector Machine (SVM) dan Decision Tree untuk Sistem Rekomendasi Tempat Wisata." *Jurnal Nasional Teknologi dan Sistem Informasi* 9.2 (2023): 113-121.
2. **Adrian, Muhammad Rivza, et al.** "Perbandingan Metode Klasifikasi Random Forest dan SVM Pada Analisis Sentimen PSBB." *Jurnal Informatika Upgris* 7.1 (2021).
3. **Ipmawati, Joang, Saifulloh Saifulloh, and Kusnawi Kusnawi.** "Analisis Sentimen Tempat Wisata Berdasarkan Ulasan pada Google Maps Menggunakan Algoritma Support Vector Machine: Sentiment Analysis of Tourist Attractions Based on Reviews on Google Maps Using the Support Vector Machine Algorithm." *MALCOM: Indonesian Journal of Machine Learning and Computer Science* 4.1 (2024): 247-256.
4. **Singgale, Yerik Afrianto.** "Analisis Sentimen Wisatawan terhadap Taman Nasional Bunaken dan Top 10 Hotel Rekomendasi Tripadvisor Menggunakan Algoritma SVM dan DT berbasis CRISP-DM." *J. Comput. Syst. Informatics* 4.2 (2023): 367-379.
5. **Khasanah, Uswatun.** *Analisis sentimen terhadap tempat wisata menggunakan metode naïve bayes classifier dan support vector machine*. BS thesis. Fakultas Sains dan Teknologi UIN Syarif Hidayatullah Jakarta, 2023.

6. **Chairunnisa, Alisa.** *Analisis Sentimen Masyarakat Terhadap Tempat Wisata di Indonesia Menggunakan Metode Support Vector Machine (SVM).* Diss. Universitas Lancang Kuning, 2022.
7. **Silviana, Selvi, Rini Astuti, and Fadhil Muhamad Basysyar.** "Penerapan Algoritma Support Vector Machine (SVM) Pada Ulasan Pengunjung Wisata Kabupaten Kuningan." *JATI (Jurnal Mahasiswa Teknik Informatika)* 8.1 (2024): 259-265.
8. **Singgalen, Yerik Afrianto.** "Analisis Performa Algoritma NBC, DT, SVM dalam Klasifikasi Data Ulasan Pengunjung Candi Borobudur Berbasis CRISP-DM." *Build. Informatics, Technol. Sci* 4.3 (2022): 1634-1646.
9. **Pratiwi, Belinda Mega, and Noora Qotrun Nada.** "Penerapan Model Machine Learning Dalam Menentukan Rekomendasi Objek Wisata Provinsi Jawa Tengah." *Proceeding Science and Engineering National Seminar*. Vol. 7. No. 1. 2022.
10. **Negara, Maulana Surya, and Ahmad Zafrullah Mardiansyah.** "Implementasi Machine Learning dengan Metode Collaborative Filtering dan Content-Based Filtering pada Aplikasi Mobile Travel (Bangkit Academy)." *Jurnal Begawe Teknologi Informasi (JBegaTI)* 5.1 (2024): 126-136.
11. **Almas, Arrafi'atu Arsy.** "BANGKIT ACADEMY 2023 LED BY GOOGLE, GOTO, TRAVELOKA–MACHINE LEARNING PATH EXPLORE EASE APP: APLIKASI SISTEM REKOMENDASI DESTINASI WISATA HIDDEN GEM BERBASIS MACHINE LEARNING." (2024).
12. **Marzuki, Asep Marzuki, Abdul Zaky, and Marido Bisra.** "Model Penerapan Sistem Rekomendasi Kuliner Pada Objek Wisata Berbasis User-Based Collaborative Filtering." *bit-Tech* 7.2 (2024): 570-580.
13. **Putra, Renno Rama, Mayo Alvarosy Chrisnatae, and Nevi Harisuci Wibowo.** "Sistem Informasi Geografis (SIG) Tempat Rekomendasi Wisata Kota Kediri." *Seminar Nasional Teknologi & Sains*. Vol. 4. No. 1. 2025.
14. **Ramadhani, Anjas, Nandang Iriadi, and Rachmat Hidayat.** "Implementasi Teknologi Rest Api Dengan Node Js Untuk Aplikasi Rekomendasi Destinasi Wisata." *Indonesian Journal Computer Science* 4.1 (2025): 22-29.
15. **Pratiwi, Dinda, Asrianda Asrianda, and Lidya Rosnita.** "Penerapan Metode Content-Based Filtering dalam Sistem Rekomendasi Objek Wisata di Aceh Tamiang." *Jurnal Ilmu Komputer dan Informatika* 4.2 (2024): 85-96.
16. **Ramdani, Ahmad Luky, Dwi Hendratmo Widyantoro, and Rinaldi Munir.** "Optimalisasi Rekomendasi Rute Pada Perencanaan Perjalanan Wisata: Studi Pustaka: Optimization Route Recommendation-Based Tourist Trip Design Problem: A Literature Study." *MALCOM: Indonesian Journal of Machine Learning and Computer Science* 4.2 (2024): 515-525.

17. **Zaidan, Rifki Alfarro, Tomi Tri Sujaka, and I. Nyoman Switrayana.** "Aplikasi Sistem Rekomendasi Lokasi Wisata Candi dan UMKM di Yogyakarta Berbasis Android dengan Metode Hybrid." *Jurnal Teknologi, Kesehatan, dan Sosial (JUTEKS)* 1.1 (2025): 23-34.
18. **Shafira, Lulu, et al.** "Implementasi Sistem Terdistribusi Aplikasi Mobile Rekomendasi Wisata Jawa Barat Menggunakan Framework Flutter." *Jurnal Ilmiah IT CIDA* 10.2 (2024): 82-96.