

Sistem Rekomendasi Destinasi Wisata di Kota Surabaya Menggunakan Metode *Content Based Filtering* dan *Neural Collaborative Filtering*

Zaky Izmi Syakura dan Kartika Fithriasari

Departemen Statistika, Institut Teknologi Sepuluh Nopember

e-mail: 5003201148@student.its.ac.id dan kartika_f@its.ac.id

Abstrak— Dalam menghadapi kesulitan akses informasi wisata di Kota Surabaya akibat pandemi dan penurunan ekonomi global, sektor pariwisata diidentifikasi sebagai opsi potensial untuk meningkatkan PDB, devisa, dan penyerapan tenaga kerja. Meskipun Surabaya memiliki kekayaan budaya dan alam yang menarik, minimnya promosi dan informasi menjadi kendala. Penelitian ini menggunakan pendekatan *neural collaborative filtering* dan *content based filtering* untuk menyusun sistem rekomendasi wisata yang lebih personal dan relevan. *Content based filtering* menunjukkan RMSE 3,1937 dan nilai *F-1 Score* 0,7 atau 70%, namun kurang efektif jika data pengguna sedikit. Sebaliknya, *neural collaborative filtering* dengan RMSE 1,4207 dan nilai *F-1 Score* 0,76 atau 76% memberikan rekomendasi 5 destinasi wisata dan menunjukkan hasil yang lebih baik. Hasilnya, *neural collaborative filtering* dapat meningkatkan kualitas rekomendasi dan mendukung pengembangan pariwisata di Surabaya..

Kata Kunci—Sistem Rekomendasi, *Content based Filtering*, *Neural Collaborative Filtering*, Surabaya, Wisatawan.

I. PENDAHULUAN

Sektor pariwisata saat ini menjadi alternatif unggulan untuk mendorong perekonomian Indonesia, terutama setelah penurunan di sektor industri dan perdagangan. Penurunan ekonomi global, perang dagang antara Tiongkok dan AS, serta pandemi COVID-19 telah berdampak buruk pada ekonomi negara tujuan ekspor Indonesia. Oleh karena itu, pariwisata, dengan kontribusinya terhadap PDB, devisa, dan tenaga kerja, dianggap sebagai solusi untuk memulihkan ekonomi nasional [1].

Indonesia, dengan 17.001 pulau yang menawarkan keberagaman budaya dan lanskap alam yang menakjubkan, memiliki potensi pariwisata yang besar. Data Badan Pusat Statistik (BPS) tahun 2022 menunjukkan bahwa meskipun terjadi peningkatan jumlah wisatawan mancanegara dari 2011 hingga 2019, jumlah ini menurun drastis pada 2020 akibat pandemi COVID-19 dan kebijakan *lockdown* [2].

Dalam era informasi berlimpah, sistem rekomendasi menjadi penting untuk membantu wisatawan menemukan destinasi yang sesuai dengan preferensi mereka. Sistem ini menggunakan data pengguna, analisis pola perilaku, dan teknologi kecerdasan buatan untuk menyajikan rekomendasi yang relevan. Daerah yang membutuhkan pengembangan pariwisata adalah Kota Surabaya yang meskipun memiliki

banyak potensi wisata, masih kurang terkenal karena minimnya promosi dan informasi yang tersedia secara luas [3].

Untuk mengatasi masalah ini, maka diperlukan sistem rekomendasi yang dapat menyaring informasi berlebih dan memberikan saran wisata yang sesuai. Tiga kategori model sistem rekomendasi yang umum digunakan adalah *collaborative filtering*, *content based filtering*, dan *hybrid recommender system*. *Neural collaborative filtering*, pengembangan dari *collaborative filtering* menggunakan jaringan saraf tiruan, menunjukkan hasil yang lebih akurat dalam memberikan rekomendasi [4].

Penelitian ini menggunakan metode *neural collaborative filtering* dan *content based filtering* untuk merekomendasikan destinasi wisata di Surabaya. Penelitian ini bertujuan untuk mendapatkan perbandingan kedua metode tersebut dan menghasilkan rekomendasi destinasi wisata yang lebih baik di Surabaya, sehingga dapat meningkatkan jumlah kunjungan wisatawan mancanegara dan nusantara ke kota ini.

II. TINJAUAN PUSTAKA

A. Sistem Rekomendasi

Sistem rekomendasi adalah teknologi yang memberikan saran kepada pengguna berdasarkan preferensi mereka terhadap *item-item* seperti produk, konten media, atau lokasi wisata, dengan memanfaatkan ulasan dan data dari berbagai sumber. Kelebihan sistem rekomendasi adalah mengatasi isu-isu terkait akurasi dan kecepatan dalam mengakses informasi yang relevan bagi pengguna dengan memberikan rekomendasi yang sesuai dengan karakteristik dan preferensi penggunaan masing-masing [5].

B. Text Preprocessing

Preprocessing dalam *text mining* adalah serangkaian langkah untuk mengubah data mentah menjadi format terstruktur yang cocok untuk analisis. Langkah-langkah umumnya meliputi *case folding*, penghapusan tanda baca, *tokenizing*, *filtering* kata-kata yang tidak penting, dan *stemming* [6]. Langkah-langkah ini penting untuk mempersiapkan data teks agar sesuai dengan kebutuhan analisis yang dilakukan.

C. Term Frequency

Term Frequency (TF) mengukur seberapa sering sebuah kata muncul dalam sebuah teks atau dokumen tertentu. Semakin

banyak kata muncul, semakin tinggi bobot yang diberikan pada kata tersebut. Dalam *binary TF* yang digunakan dalam penelitian ini nilai bobot kata adalah 1 jika kata ada dalam dokumen dan 0 jika tidak [7]. Dengan demikian, frekuensi kemunculan kata dalam dokumen tidak mempengaruhi nilai TF dalam metode ini.

D. Content based Filtering

Dalam sistem rekomendasi menggunakan *metode content based filtering*, rekomendasi *item* untuk pengguna didasarkan pada kesamaan fitur atau atribut dengan *item* yang telah disukai atau dipilih sebelumnya oleh pengguna [8]. Fitur-fitur ini mencakup atribut-atribut dari *item* yang mempengaruhi penilaian pengguna, yang digunakan untuk membangun matriks *user-item*. Matriks ini merepresentasikan hubungan antara pengguna dan item dengan menyimpan *rating* yang diberikan oleh pengguna pada setiap *item* atau tempat. Pada matriks *user-item* tersebut dilakukan normalisasi pada nilai *rating*. Metode normalisasi nilai *rating* menggunakan *Min-Max Scaler*. Nilai *rating* yang telah dinormalisasi dapat dinyatakan sebagai r_{scaled} . Nilai asli *rating* sebelum dinormalisasi dilambangkan sebagai $r_{i,j}$. Nilai maksimum *rating* dinyatakan sebagai r_{max} , sementara nilai minimum *rating* dinyatakan sebagai r_{min} .

$$r_{scaled} = \frac{r_{i,j} - r_{min}}{r_{max} - r_{min}}, \quad (1)$$

Dengan memanfaatkan bobot fitur kata dan matriks *item features*, sistem menghasilkan tabel *item features* untuk setiap pengguna. Tabel ini kemudian digunakan untuk menghitung bobot pada metode *content based filtering* mengoptimalkan rekomendasi dengan mempertimbangkan preferensi dan histori pengguna terhadap *item* tertentu [9]. Berikut adalah rumus yang digunakan untuk menentukan nilai bobot dari masing-masing fitur dengan menggunakan tabel *rating* dan *item features* pada setiap *user*. $b(u_i, D_k)$ adalah bobot pada *user* ke- i terhadap fitur kata ke- k . J_i adalah jumlah *item* yang diberi *rating* oleh *user* ke- i . $d_{k,j}$ adalah *binary term frequency* untuk fitur kata ke- k dalam *item* ke- j dan $r_{i,j}$ adalah *rating* yang diberikan oleh *user* ke- i terhadap *item* ke- j .

$$b(u_i, D_k) = \frac{1}{|J_i|} \sum_{j \in J_i} d_{k,j} \times r_{i,j}, \quad (2)$$

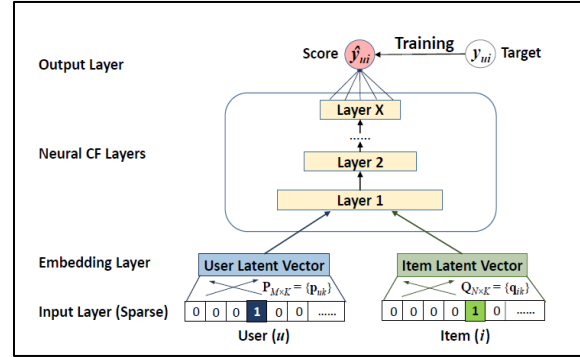
Di bawah ini terdapat formula yang bisa dipakai untuk mengestimasi *rating* yang akan diberikan oleh seorang pengguna pada suatu *item*. $\hat{r}(i, j)$ adalah estimasi *rating* *user* ke- i terhadap *item* ke- j . D_j adalah jumlah fitur kata yang muncul dalam *item* ke- j .

$$\hat{r}(i, j) = \frac{1}{|D_j|} \sum_{k \in D_j} b(u_i, D_k), \quad (3)$$

E. Neural Collaborative Filtering

Pada penelitian ini menggunakan metode pendekatan *Neural Collaborative Filtering* (NCF) yang menggabungkan teknik *matrix factorization* (MF) dan *multi-layer perceptron* (MLP) untuk meningkatkan kinerja sistem rekomendasi. Model NCF yang digunakan dalam penelitian ini terdiri dari beberapa *layer*, termasuk *input layer* untuk dapat menerima data masukan, *embedding layer* untuk mengubah data kategorikal menjadi

representasi vektor yang kontinu, dan MLP yang bertanggung jawab untuk memproses vektor representasi ini untuk memprediksi preferensi pengguna terhadap *item* [10].



Gambar 1. Arsitektur Neural Collaborative Filtering

Input layer pada metode NCF terbagi menjadi dua bagian, satu untuk destinasi wisata dan satu lagi untuk pengguna. Dengan menggunakan representasi *one-hot encoding* untuk merepresentasikan identitas unik mereka. Ini memastikan informasi dasar tentang destinasi dan pengguna dioptimalkan dalam proses rekomendasi. Selanjutnya, *embedding layer* mengubah matriks *rating user* dan *item* yang kompleks menjadi vektor dengan dimensi lebih rendah. Hasilnya adalah matriks *embedding weights* yang terdiri dari matriks *embedding user* dan *item* memetakan pengguna dan *item* ke dalam ruang vektor *embedding* dengan dimensi laten yang ditentukan awalnya diinisialisasi secara acak berdistribusi normal untuk memulai proses pembelajaran.

Neural Collaborative Filtering (NCF) mengintegrasikan dua pendekatan untuk memodelkan *user* dan *item* dengan menggabungkan vektor mereka. Pendekatan ini mengusulkan penambahan *hidden layer* dengan menggunakan *multi-layer perceptron* atau *fully connected layer* untuk memahami interaksi yang lebih kompleks dan *non-linear* antara faktor laten *user* dan *item*. \mathbf{z} adalah matriks *input* untuk layer berikutnya yang merupakan gabungan vektor faktor laten *user* ke- i dan *item* ke- j . \mathbf{h}_t adalah matriks *output hidden layer* ke- t . $\hat{\mathbf{R}}_{IJ}$ adalah vektor prediksi *rating*. \mathbf{a}_t adalah fungsi aktivasi *hidden layer* ke- t . \mathbf{a}_o adalah fungsi aktivasi *output*. \mathbf{P}_i adalah matriks faktor laten *user* ke- i . \mathbf{Q}_j adalah matriks faktor laten *item* ke- j . \mathbf{W}_t adalah matriks bobot pada *hidden layer* ke- t . \mathbf{W}_o adalah vektor bobot pada *output*. \mathbf{c}_t adalah matriks bias pada *hidden layer* ke- t dan \mathbf{c}_o adalah vektor bias pada *output*.

$$\mathbf{z} = (\mathbf{P}_i, \mathbf{Q}_j) = \begin{bmatrix} \mathbf{P}_i \\ \mathbf{Q}_j \end{bmatrix}, \quad (4)$$

$$\mathbf{h}_t = \mathbf{a}_t(\mathbf{z}^T \mathbf{W}_t + \mathbf{c}_t), \quad (5)$$

$$\hat{\mathbf{R}}_{IJ} = \mathbf{a}_o(\mathbf{a}_t \mathbf{W}_o + \mathbf{c}_o), \quad (6)$$

$$\hat{\mathbf{R}}_{IJ} = \begin{bmatrix} \hat{r}_{1,1} \\ \hat{r}_{1,2} \\ \vdots \\ \hat{r}_{IJ} \end{bmatrix} \quad (7)$$

Backpropagation adalah algoritma kunci dalam pelatihan

jaringan saraf yang digunakan untuk menyesuaikan bobot dan bias berdasarkan kesalahan antara keluaran aktual jaringan dan target yang diinginkan. Proses ini terjadi dari lapisan *output* ke lapisan *input* memungkinkan jaringan untuk mengoptimalkan kinerjanya dengan mengurangi *error* pada hasil prediksi. *Backpropagation* menggunakan turunan parsial dari *error* terhadap bobot menghitung penyesuaian yang diperlukan, sehingga meningkatkan kemampuan jaringan dalam membuat prediksi yang lebih akurat dalam domain *deep learning*. Perhitungan E_{total} pada *output layer* menggunakan *loss function mean squared error (MSE)*. E_{total} adalah total *error*. w_s^t adalah bobot ke- s pada *layer* ke- t dan h^t adalah *output* pada *layer* ke- t .

$$\frac{\partial E_{total}}{\partial w_s^t} = \frac{\partial E_{total}}{\partial \hat{r}_{ij}} \times \frac{\partial \hat{r}_{ij}}{\partial h^o} \times \dots \times \frac{\partial h^t}{\partial w_s^t} \quad (8)$$

$$MSE = \frac{\sum_{i=1}^I \sum_{j=1}^J (r_{ij} - \hat{r}_{ij})^2}{N} \quad (9)$$

Dengan menggunakan algoritma *backward propagation*, *optimizer* seperti *stochastic gradient descent (SGD)* menghitung turunan parsial dari fungsi kerugian terhadap bobot. Proses ini memungkinkan pembaharuan bobot dilakukan dengan memperhatikan *learning rate* dan *hyperparameter* lainnya untuk mengatur kecepatan dan arah perubahan bobot. SGD dikenal karena kesederhanaannya dan stabilitasnya yang baik dalam menghadapi dataset besar, serta mempengaruhi konsistensi dan konvergensi model dalam pelatihan jaringan saraf. $w_{s,g+1}^t$ adalah bobot ke- s pada *layer* ke- t pada iterasi ke- g dan γ adalah parameter *learning rate*.

$$w_{s,g+1}^t = w_{s,g}^t - \gamma \frac{\partial E_{total}}{\partial w_s^t} \quad (10)$$

F. Root Mean Square Error (RMSE)

Salah satu metode evaluasi yang umum digunakan adalah *Root Mean Square Error (RMSE)*. RMSE digunakan untuk mengukur seberapa baik model mampu memprediksi nilai sebenarnya dengan menghitung akar dari rata-rata kuadrat dari selisih antara nilai prediksi dan nilai yang sebenarnya. Semakin rendah nilai RMSE yang diperoleh, semakin baik kualitas prediksi yang dilakukan oleh sistem rekomendasi tersebut. Pada persamaan berikut simbol I menunjukkan banyaknya *user*. J menunjukkan banyaknya *item* dan N menunjukkan jumlah total data yang tersedia.

$$RMSE = \sqrt{\frac{\sum_{i=1}^I \sum_{j=1}^J (r_{ij} - \hat{r}_{ij})^2}{N}} \quad (11)$$

G. Confusion Matrix

Confusion matrix adalah metode untuk menilai kinerja sistem dengan membandingkan hasil klasifikasi sistem dengan hasil yang seharusnya. *Confusion matrix* penelitian ini terdiri dari TP (*True Positive*), TN (*True Negative*), FP (*False Positive*), dan FN (*False Negative*). TP dan TN mencerminkan klasifikasi yang benar, sementara FP dan FN mencerminkan klasifikasi yang salah. Metode ini digunakan mengukur kinerja klasifikasi meliputi *accuracy*, *recall*, *precision*, dan *F-1 Score* [12].

Pada penelitian ini *confusion matrix* menggunakan klasifikasi

prediksi *rating*. Prediksi *rating* yang akan diklasifikasikan menjadi "Direkomendasikan" dan "Tidak Direkomendasikan" berdasarkan nilai batas (*threshold*) 2,5. Jika prediksi atau *rating* aktual $\geq 2,5$, maka masuk ke kelas "Direkomendasikan". Jika $< 2,5$, masuk ke kelas "Tidak Direkomendasikan".

Tabel 1.
Confusion Matrix

Actual	Predict	
	Direkomendasikan	Tidak Direkomendasikan
Direkomendasikan	TP	FN
Tidak Direkomendasikan	FP	TN

G. Pariwisata

Pariwisata merujuk pada kegiatan perjalanan seseorang ke tempat lain untuk tujuan rekreasi, bisnis, pendidikan, atau lainnya. Fenomena ini mencakup aspek-aspek seperti perjalanan sementara tanpa tekanan, eksplorasi berbagai tujuan dengan beragam tujuan, dan pengalaman untuk merasakan kepuasan atau mendapatkan pengetahuan baru. Meskipun memberikan dampak positif seperti peningkatan ekonomi dan pengembangan infrastruktur, pariwisata juga dapat menyebabkan masalah lingkungan dan sosial. Sebagai sektor ekonomi yang berkembang pesat secara global, pariwisata memainkan peran penting dalam pertukaran budaya dan pertumbuhan ekonomi suatu negara [12].

III. METODOLOGI

A. Sumber Data

Penelitian ini menggunakan sumber data sekunder dari dataset "Indonesia Tourism Destination" yang tersedia di Kaggle. Data ini fokus pada Kota Surabaya dan terdiri dari dua bagian utama. Pertama, informasi tentang 46 destinasi wisata di Kota Surabaya, termasuk id, nama, deskripsi, dan kategori tempat wisata. Kedua, data *rating* dari 293 wisatawan yang memberikan penilaian terhadap destinasi tersebut. Jumlah total wisatawan yang memberikan *rating* pada dataset ini adalah 300. Secara keseluruhan, data yang digunakan dalam penelitian ini mencakup 1050 baris dari total 10.000 baris dalam keseluruhan dataset.

B. Variabel Penelitian

Variabel yang digunakan dalam penelitian ini sebagai berikut.

Tabel 2.
Variabel Data

No	Simbol	Variabel	Keterangan	Skala
1	U	$User_Id$	Kode identitas wisatawan dimana setiap wisatawan memiliki Id yang berbeda	Nominal
2	X_1	$Place_Id$	Kode identitas tempat wisata di Kota Surabaya dimana setiap tempat memiliki Id yang berbeda	Nominal
3	X_2	$Place_Name$	Nama tempat wisata di Kota Surabaya	Nominal

Tabel 2.

Variabel Data (Lanjutan)

No	Simbol	Variabel	Keterangan	Skala
4	X_3	Description	Penjelasan deskripsi informasi mengenai tempat wisata di Kota Surabaya	Nominal
5	X_4	Category	Kategori tempat wisata di Kota Surabaya yang terdiri dari 6 kategori wisata	Nominal
6	R	Rating	Penilaian atau rating dari masing-masing wisatawan terhadap tempat wisata di Kota Surabaya dengan skala (1-5)	Interval

C. Struktur Data

Struktur data yang digunakan pada penelitian ini ada 2, yaitu struktur data tempat wisata pada Tabel 3 dan struktur data rating wisatawan pada Tabel 4.

Tabel 3.

Struktur Data Tempat Wisata

Place_Id(X_1)	Place_Name(X_2)	Description(X_3)	Category(X_4)
$x_{1,1}$	$x_{2,1}$	$x_{3,1}$	$x_{4,1}$
$x_{1,2}$	$x_{2,2}$	$x_{3,2}$	$x_{4,2}$
$x_{1,3}$	$x_{2,3}$	$x_{3,3}$	$x_{4,3}$
...
$x_{1,46}$	$x_{2,46}$	$x_{3,46}$	$x_{4,46}$

Tabel 4.

Struktur Data Rating Wisatawan

User_Id(U)	Place_Id(X_1)	Rating(R)
u_1	$x_{1,1}$	$r_{1,1}$
	$x_{1,2}$	$r_{1,2}$

	x_{1,n_1}	r_{1,n_1}
u_2	$x_{2,1}$	$r_{2,1}$
	$x_{2,2}$	$r_{2,2}$

	x_{2,n_2}	r_{2,n_2}
...
u_{293}	$x_{293,1}$	$r_{293,1}$
	$x_{293,2}$	$r_{293,2}$

	$x_{293,n_{293}}$	$r_{293,n_{293}}$

D. Langkah Analisis

Berikut merupakan langkah analisis yang dilakukan pada penelitian ini.

1. Unduh data *Indonesian Tourism Destination* dari Kaggle, termasuk dataset *tourism_rating* dan *tourism_with_id*.
2. Lakukan *preprocessing data* dengan memilih atribut yang relevan dari kedua dataset, seperti *User_Id*, *Place_Id*, *Place_Ratings*, *Place_Name*, *Description*, *Category*, dan *City*. Gabungkan dataset berdasarkan *Place_Id* dan saring data untuk Kota Surabaya.
3. Implementasikan metode *content based filtering* dengan memilih atribut nama tempat, deskripsi, dan kategori.

Lakukan transformasi teks, *preprocessing (case folding, remove punctuation, tokenizing, stopwords removal,*

4. *stemming*), dan buat matriks *item features* serta matriks *user-item* dan menghitung prediksi *rating* dan evaluasi dengan RMSE.
5. Terapkan metode *neural collaborative filtering* dengan membangun arsitektur model, membagi data menjadi *training*, *validation*, dan *test set*, melatih model, menghitung prediksi *rating*, melakukan *hyperparameter tuning*, dan menghitung evaluasi dengan RMSE.
6. Bandingkan hasil evaluasi antara metode *content based filtering* dan *neural collaborative filtering*.
7. Buat rekomendasi berdasarkan sistem yang dikembangkan.
8. Tarik kesimpulan dari hasil penelitian untuk mendukung tujuan dan temuan yang telah disampaikan.

IV. HASIL DAN PEMBAHASAN

A. Metode Content Based Filtering

Pada metode *content based filtering* melakukan *preprocessing* data untuk mengatur struktur data mentah yang tidak teratur. *Preprocessing* ini diperlukan agar data dapat digunakan dalam analisis lebih lanjut, terutama difokuskan pada data destinasi wisata dari Tabel 3.2 dan Tabel 3.3. hasil dari *preprocessing* dengan menghapus beberapa tempat destinasi wisata yang memiliki *rating* sedikit dan wisatawan yang memberikan *rating* sedikit sehingga pada data terdapat 38 destinasi wisata dan 288 wisatawan. Setelah itu dilanjutkan dengan *text preprocessing*.

B. Text Preprocessing

Pada tahap *preprocessing* teks data diubah menjadi format yang siap dianalisis melalui beberapa langkah *case folding* untuk mengubah semua teks menjadi huruf kecil, penghapusan tanda baca dan angka, *tokenizing* untuk memecah teks menjadi kata-kata individual, *filtering* untuk menghapus *stopwords* yang tidak relevan, dan *stemming* untuk mengubah kata-kata menjadi bentuk dasarnya. Hasil dari proses ini adalah teks yang lebih bersih dan siap untuk analisis lebih lanjut dalam sistem rekomendasi. Hasil dari *preprocessing* teks sebagai berikut.

Tabel 5.

Hasil Text Preprocessing

Place_Id	Hasil
1	[ekowisata, mangrove, wonorejo, hutan, wisata, mangrove, surabaya, wisata, gabung, wisata, rekreasi, edukasi, area, luas, hektar, tanam, tanam, bakau, lokasi, wisata, kelola, perintah, kota, surabaya, wisata, mati, unjung, tata, rapi, bersih, aktivitas, wisata, keliling, hutan, mangrove, perahu, jalan, kaki, wisata, edukasi, ekosistem, cagar, alam]
2	[taman, harmoni, putih, taman, harmoni, putih, surabaya, milik, gudang, koleksi, bunga, warna, warni, dulunya, lokasi, buang, sampah, tpa, transformasi, wisata, taman, cantik, cantik, bunga, bunga, mekar, taman, harmoni, instagramable, heran, mayoritas, unjung, kaum, kawula, muda, giat, sosial, media, lantasi, budget, pasuk, kawasan, taman, harmoni, surabaya, nggak, khawatir, harga, tiket, masuk, taman, harmoni, putih, surabaya, gratis, koeck, bayar, karcis, parkir, cagar, alam]

Tabel 5.
Hasil Text Preprocessing (Lanjutan)

Place Id	Hasil
...	...
46	[gereja, perawan, maria, dosa, surabaya, gereja, katolik, lahir, santa, perawan, maria, gereja, tua, kota, surabaya, jawa, timur, indonesia, lokasi, jalan, kepanjen, surabaya, bangun, religius, damping, gedung, sekolah, katolik, frater, surabaya, ibadah]

C. Matriks Item Features

Tahap awal dalam prediksi *rating* adalah membentuk matriks *item features* yang merepresentasikan *item* dalam bentuk matriks. Setiap baris mewakili satu destinasi wisata dan setiap kolom mewakili kata tertentu dari destinasi tersebut. Nilai biner digunakan "1" menunjukkan keberadaan kata pada item tersebut, sedangkan "0" menunjukkan ketidadaannya. Matriks ini digunakan untuk menentukan kesamaan atau relevansi antara *item* dalam sistem rekomendasi. Matriks *item features* ditampilkan pada Tabel 6.

Tabel 6.
Matriks Item Features

Features Place Id	abadi	...	air	...	zeven
1	0	...	0	...	0
2	0	...	0	...	0
3	0	...	1	...	0
4	0	...	0	...	0
5	0	...	0	...	0
...
34	0	...	0	...	0
35	0	...	1	...	0
36	0	...	1	...	0
37	0	...	0	...	0
38	0	...	0	...	0

D. Matriks User Item

Tahap berikutnya adalah pembentukan matriks *user-item*, yang menggambarkan hubungan antara wisatawan dan destinasi wisata. Matriks ini digunakan sebagai dasar untuk analisis dalam sistem rekomendasi, dengan celah kosong untuk memprediksi *rating* yang hilang. Nilai kosong tersebut diberi tanda * yang menunjukkan bahwa *item* tersebut belum diberi *rating* oleh *user*. Kemudian *rating* tersebut dinormalisasi dengan rentang 0 hingga 1 sehingga hasilnya sebagai berikut.

Tabel 7.
Matriks User Item

Place Id User Id	1	2	3	...	36	37	38
1	1	1	0,6	...	*	*	*
2	*	*	*	...	*	*	*
3	*	*	*	...	*	*	*
4	*	*	*	...	*	*	*
5	*	*	*	...	*	*	*
...
284	*	*	*	...	*	*	*
285	*	*	*	...	*	0,4	*
286	*	*	*	...	*	0,2	*
287	*	*	*	...	*	*	*
288	*	*	*	...	*	*	*

E. User Profile

User Profile dalam *Content based Filtering* (CBF) dapat menggambarkan preferensi dan karakteristik pengguna terhadap *item*, berdasarkan *rating* yang diberikan pengguna terhadap kata-kata destinasi wisata. Profil ini mencakup kata-kata yang disukai dan bobot kepentingannya, yang membantu menghasilkan rekomendasi yang sesuai dengan minat pengguna. Setelah memperoleh matriks *item features* dan *user-item*, langkah berikutnya adalah membentuk matriks *user profile*. Berdasarkan hasil perhitungan bobot kata pada setiap *user* dengan menggunakan persamaan (1). Berikut adalah hasil perhitungan *user profile* untuk wisatawan dengan *Id* 5 yang ditampilkan pada Tabel 8.

Tabel 8.
User Profile Wisatawan Id 5

User	abadi	abdurrahman	...	yogyakarta	zaman	zeven
5	0	0	...	0,1	0	0

F. Perhitungan Prediksi Rating Content based Filtering

Sistem rekomendasi kemudian memprediksi *rating* yang mungkin diberikan oleh user terhadap *item* yang belum mereka nilai dengan menggunakan persamaan (3). Berikut adalah contoh perhitungan prediksi *rating* untuk wisatawan dengan *Id* 5 terhadap *item* 5.

$$\hat{r}(5,5) = \frac{0,25 + 0,13 + 0,33 + 0,125 + \dots + 0,13 + 0,375 + 0,25}{77} = \frac{3,475}{77}$$

$$\hat{r}(5,5) = 0,04513$$

G. Evaluasi Metode Content based Filtering

Setelah prediksi *rating* dilakukan, langkah selanjutnya adalah mengevaluasi metode *Content based Filtering* (CBF) melalui beberapa percobaan *split* data. Evaluasi ini mengukur kinerja CBF dalam menghasilkan rekomendasi yang sesuai dengan preferensi pengguna berdasarkan hasil prediksi *rating*. Data akan dibagi menjadi beberapa rasio *split*, yaitu 80:20, 75:25, 70:30, 65:35, dan 60:40. Hasil prediksi akan dievaluasi menggunakan *Root Mean Squared Error* (RMSE) untuk membandingkan *rating* aktual dengan prediksi. Berikut adalah hasil perbandingan RMSE dari berbagai pembagian data tersebut. Berdasarkan hasil evaluasi, rasio pembagian data terbaik adalah 70:30 karena nilai RMSE *test* yang kecil dan jarak antara RMSE *train* dan *test* tidak terlalu jauh, meminimalkan *overfitting*.

Tabel 9.
Evaluasi Hasil Metode Content based Filtering

Rasio Pembagian Data	RMSE Train	RMSE Test
80:20	1,2705	2,9845
75:25	1,0077	2,9134
70:30	1,2483	2,9310
65:35	1,2593	3,0036
60:40	1,2622	3,0739

Sistem rekomendasi dievaluasi menggunakan *confusion matrix* yang memerlukan klasifikasi prediksi *rating*. Prediksi diklasifikasikan menjadi "Direkomendasikan" dan "Tidak Direkomendasikan" berdasarkan nilai batas (*threshold*) 2,5.

Dari klasifikasi tersebut diperoleh hasil nilai-nilai *confusion matrix* sebagai berikut.

Tabel 10.

Hasil Perhitungan *Confusion Matrix Content based Filtering*

Rasio Data	Accuracy	Precision	Recall	F-1 Score
70:30	0,97	0,95	0,56	0,70

H. Hasil Rekomendasi Metode Content based Filtering

Prediksi *rating* dapat dilakukan untuk destinasi wisata yang belum pernah dikunjungi atau diberi *rating* oleh pengguna. Sebagai contoh, prediksi *rating* dilakukan untuk wisatawan dengan *Id* 5. Model ini memberikan estimasi *rating* tertinggi untuk destinasi wisata yang belum pernah dinilai oleh pengguna tersebut.

Tabel 11.

Hasil Rekomendasi Pada Wisatawan *Id* 5 Metode CBF

Place Id	Predict Rating
14	4,99
11	4,44
15	4,30
45	4,25
8	4,19
41	4,09
6	4,03
40	3,93

Berikut hasil prediksi *rating* untuk wisatawan dengan *Id* 5. Tempat wisata dengan *Id* 14 (Surabaya *North Quay*) mendapatkan *rating* tertinggi sebesar 4,99. Diikuti oleh *Food Junction Grand Pakuwon* (*Id* 11) dengan *rating* 4,44, dan Kebun Bibit Wonorejo (*Id* 15) dengan *rating* 4,30. Rekomendasi lainnya termasuk Taman Flora Bratang Surabaya (*Id* 45) dan Taman Pelangi (*Id* 8).

I. Metode Neural Collaborative Filtering

Metode *Neural Collaborative Filtering* (NCF) menggabungkan *collaborative filtering* tradisional dengan *neural networks* untuk meningkatkan akurasi rekomendasi. Penelitian ini memusatkan penggunaan metode NCF dengan fokus pada *embedding layer* dan *fully connected layer* untuk meningkatkan kinerja sistem rekomendasi destinasi wisata. *Embedding layer* digunakan untuk mewakili pengguna dan destinasi wisata dalam vektor berdimensi rendah yang memungkinkan model untuk menangkap informasi dengan lebih efisien. Selanjutnya, *fully connected layer* digunakan untuk memodelkan interaksi *non-linear* antara pengguna dan destinasi wisata memperkuat kemampuan sistem dalam merekomendasikan destinasi berdasarkan preferensi pengguna dengan lebih akurat. Arsitektur model NCF yang digunakan terdapat pada Gambar 2.

J. Hyperparameter Tuning

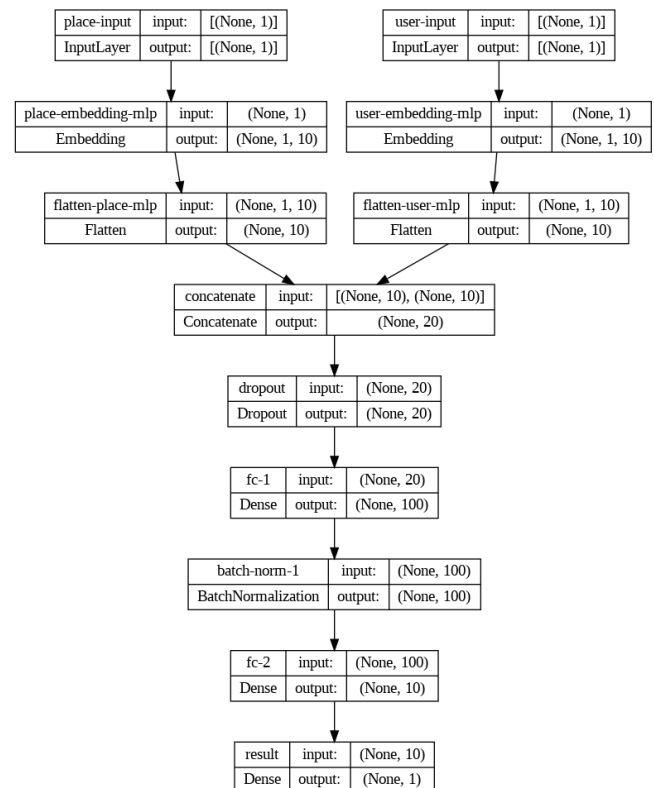
Pada tahap *hyperparameter tuning* dalam penelitian ini, fokus diberikan pada faktor laten 5, 10, dan 50 dengan penggunaan *optimizer Stochastic Gradient Descent* (SGD) dan variasi *learning rate* 0.01, 0.001, dan 0.0001. Setiap kombinasi *hyperparameter* diuji pada data *train*, *validation*, dan *test* yang dibagi sesuai proporsi yang telah ditetapkan. Untuk meningkatkan efisiensi pelatihan, digunakan *callback*

EarlyStopping yang menghentikan pelatihan lebih awal jika tidak ada peningkatan pada *loss validation* selama 10 *epoch* berturut-turut, serta mengembalikan bobot model ke keadaan terbaik saat mencapai *loss validation* terendah. Evaluasi model dilakukan berdasarkan *mean squared error* pada data *train* dan *validation*. Berikut hasil *hyperparameter tuning*.

Tabel 12.

Hyperparameter Tuning

Faktor Laten	Learning Rate	RMSE Train	RMSE Test
5	0,01	0,9122	1,4738
	0,001	1,2712	1,4731
	0,0001	1,3183	1,4850
10	0,01	0,9622	1,4260
	0,001	1,2317	1,3943
	0,0001	1,3067	1,4207
50	0,01	0,7666	1,5195
	0,001	1,2350	1,4846
	0,0001	1,2271	1,4842

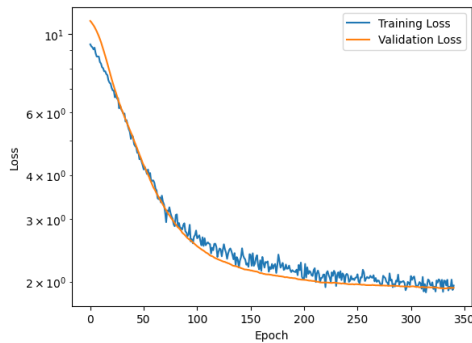


Gambar 2. Model Neural Collaborative Filtering

K. Evaluasi Metode Neural Collaborative Filtering

Dari hasil *hyperparameter tuning* ditemukan kombinasi terbaik pada faktor laten 10 dan *learning rate* 0,0001 dengan nilai RMSE terendah 1,4207. Model ini menunjukkan performa optimal dalam memprediksi *rating* dengan kesalahan prediksi yang minimal dibandingkan dengan konfigurasi lainnya.

Grafik pelatihan menunjukkan bahwa baik *training loss* maupun *validation loss* awalnya tinggi tetapi menurun secara bertahap seiring bertambahnya *epoch*, menandakan model sedang belajar dan menyesuaikan bobot untuk meminimalkan kesalahan. *Training loss* turun cepat di awal dan stabil dengan fluktuasi kecil setelah sekitar 100 *epoch*, sementara *validation loss* mengikuti tren yang serupa meskipun sedikit lebih tinggi



dari *training loss* menunjukkan tidak adanya *overfitting*.

Gambar 3. Proses *Training Model Terbaik*

Pada akhir pelatihan, keduanya stabil mendekati nilai minimum, menunjukkan bahwa model telah konvergen. Grafik ini menyiratkan pada data validasi yang kemudian dilakukan *callback EarlyStopping* untuk mencegah *overfitting* dan mengembalikan model terbaik. Kemudian dievaluasi menggunakan *confusion matrix* yang memerlukan klasifikasi prediksi *rating* dan hasil kebaikan model sebagai berikut.

Tabel 13.

Hasil Perhitungan *Confusion Matrix Neural Collaborative Filtering*

Faktor Laten	Learning Rate	Accuracy	Precision	Recall	F-1 Score
10	0,0001	0,65	0,67	0,88	0,76

L. Hasil Rekomendasi Neural Collaborative Filtering

Dengan menggunakan model terbaik ini dapat dilakukan prediksi *rating* dengan hasil yang lebih baik. Hasil prediksi *rating* dengan menggunakan model terbaik ini pada data *test* ditampilkan pada Tabel 14.

Tabel 14.

Hasil Prediksi Model Terbaik Pada Data *Test*

User Id	Actual Rating	Predict Rating
377	2	2,41
823	3	3,45
382	3	2,67
576	4	4,39
259	1	1,82
837	5	4,10
650	4	3,88
764	3	3,07
809	2	2,00

Selanjutnya, prediksi *rating* dilakukan untuk wisatawan dengan *Id 5* menggunakan model terbaik hasil dari *hyperparameter tuning*. Model ini mampu memberikan estimasi *rating* untuk destinasi wisata yang belum pernah dinilai oleh pengguna berdasarkan data historis dan preferensi mereka. Berikut adalah hasil prediksi *rating* tertinggi untuk wisatawan *Id 5* terhadap destinasi wisata.

Tabel 15.

Hasil Rekomendasi Pada Wisatawan *Id 5* Metode NCF

Place Id	Predict Rating
29	4,46
36	4,12
45	4,02
42	4,00
16	3,93
17	3,88
35	3,87

Hasil penelitian menunjukkan bahwa untuk wisatawan

dengan *Id 5*, Surabaya Museum (*Id 29*) mendapat *rating* tertinggi sebesar 4,46, diikuti oleh Patung Buddha Empat Rupa (*Id 36*) dengan *rating* 4,12, dan Taman Flora Bratang Surabaya (*Id 45*) dengan *rating* 4,02. Rekomendasi wisata lainnya adalah Museum Mpu Tantular, Taman Ekspresi dan Perpustakaan, Kebun Binatang Surabaya, dan Balai Kota Surabaya.

M. Perbandingan Metode Sistem Rekomendasi

Pembuatan sistem rekomendasi dengan *metode content based filtering* dan *neural collaborative filtering* dilakukan perbandingan performansi dari kedua metode tersebut. Perbandingan performansi dilakukan dengan membandingkan hasil evaluasi nilai *Root Mean Square Error* (RMSE). Perbandingan ini bertujuan untuk menentukan metode manakah yang paling baik. Hasil evaluasi kedua metode tersebut dikumpulkan pada Tabel 16.

Tabel 16.

Perbandingan Hasil Evaluasi

Metode	RMSE	F-1 Score
Content based Filtering (CBF)	2,9310	0,70
Neural Collaborative Filtering (NCF)	1,4207	0,76

Berdasarkan hasil evaluasi, metode *neural collaborative filtering* menunjukkan kinerja yang jauh lebih baik dibandingkan metode *content based filtering* dalam sistem rekomendasi destinasi wisata di Kota Surabaya. *Neural collaborative filtering* memiliki RMSE sebesar 1,4207 dan nilai *F-1 Score* 76% sedangkan *content based filtering* memiliki RMSE sebesar 3,1937 dan nilai *F-1 Score* 70%. Hal ini menegaskan bahwa pendekatan *neural collaborative filtering* lebih efektif dan digunakan dalam memberikan rekomendasi pada penelitian ini.

V. KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan hasil pembahasan diperoleh kesimpulan pada penelitian ini sebagai berikut.

1. Pada penelitian dari kedua metode diperoleh hasil sebagai berikut.
 - a. Metode *content based filtering* pada penelitian ini mampu memberikan prediksi dengan RMSE sebesar 2,9310 dan *F-1 Score* 0,70. Metode ini cukup baik dalam memberikan rekomendasi, tetapi metode ini memiliki keterbatasan jika data pengguna tidak mencukupi untuk profil yang komprehensif.
 - b. *Neural collaborative filtering* dengan parameter *learning rate* 0,0001 dan 10 faktor laten menghasilkan prediksi yang akurat dengan RMSE 1,4207 dan *F-1 Score* 0,76. Metode ini efektif dalam menangkap kompleksitas interaksi antara pengguna dan *item* meningkatkan kualitas rekomendasi.
2. Perbandingan menunjukkan bahwa *neural collaborative filtering* lebih baik dalam memprediksi *rating* dibandingkan *content based filtering* dengan RMSE dan *F-1 Score* lebih rendah dan kemampuan lebih baik dalam merekomendasikan produk yang relevan.
3. Hasil rekomendasi memberikan destinasi wisata di Kota Surabaya direkomendasikan kepada wisatawan sebagai contoh wisatawan *Id 5*, yaitu Surabaya Museum, Patung Buddha Empat Rupa, Taman Flora Bratang Surabaya,

Museum Mpu Tantular, Taman Ekspresi dan Perpustakaan, Kebun Binatang Surabaya, dan Balai Kota Surabaya.

B. Saran

Saran untuk penelitian selanjutnya adalah mengoptimalkan metode *content based filtering* dengan menggunakan lebih banyak data untuk mengurangi matriks *sparse*. Penambahan variabel seperti umur dan tempat tinggal pengguna juga dapat meningkatkan kualitas sistem rekomendasi. Selain itu, penelitian diharapkan untuk mengeksplorasi metode lain yang lebih inovatif dan efektif untuk mengatasi keterbatasan yang dimiliki pada metode yang digunakan.

DAFTAR PUSTAKA

- [1] Maulana, A. (2019). Kajian Perhitungan Global Gross Travel Propensity (Gtp) Dan Country Potential Generation Index (Cpgi) 2016. *Jurnal Pariwisata Pesona*, 4(2), 139–148.
- [2] Badan Pusat Statistik Provinsi Jawa Timur. (2022). *Statistik Pariwisata Provinsi Jawa Timur 2021*. Badan Pusat Statistik.
- [3] Adityaji, R. (2018). Formulasi Strategi Pengembangan Destinasi Pariwisata Dengan Menggunakan Metode Analisis Swot: Studi Kasus Kawasan Pecinan Kapasan Surabaya. *Jurnal Pariwisata Pesona*, 3(1). <https://doi.org/10.26905/jpp.v3i1.2188>
- [4] Zhang, L., Luo, T., Zhang, F., & Wu, Y. (2018). A Recommendation Model Based on Deep Neural Network. *IEEE Access*, 6, 9454–9463. <https://doi.org/10.1109/ACCESS.2018.2789866>
- [5] Ricci, F., Shapira, B., Rokach, L., & Kantor, P. B. (2011). *Recommender Systems Handbook*. Springer Science+Business Media.
- [6] Imbar, R. V., Ayub, M., Rehatta, A., Jurusan, S., Informasi, S., & Informatika, T. (2014). Implementasi Cosine Similarity dan Algoritma Smith-Waterman untuk Mendeteksi Kemiripan Teks. *Jurnal Informatika*, 31–42.
- [7] Siregar, R. R. A., Sinaga, F. A., & Arianto, R. (2017). Aplikasi Penentuan Dosen Penguji Skripsi Menggunakan Metode TF-IDF dan Vector Space Model. *Computatio: Journal of Computer Science and Information Systems*, 1(2), 171. <https://doi.org/10.24912/computatio.v1i2.1014>
- [8] Ricci, F., Rokach, L., & Shapira, B. (2011). *Introduction to Recommender Systems Handbook* (F. Ricci, Ed.). Springer Science+Business Media.
- [9] Uluaymur, M., Cataltepe, Z., & Tayfur, E. (2012). Content based movie recommendation using different feature sets. *Lecture Notes in Engineering and Computer Science*, 1, 517–521.
- [10] He, X., Liao, L., Zhang, H., Nie, L., Hu, X., & Chua, T. S. (2017). Neural collaborative filtering. *26th International World Wide Web Conference, WWW 2017*, 173–182. <https://doi.org/10.1145/3038912.3052569>
- [11] Han, J., Kamber, M., & Pei, J. (2012). *Data Mining: Concepts and Techniques* (3rd ed.).
- [12] Riani, N. K. (2021). Pariwisata Adalah Pisau Bermata 2. *Jurnal Inovasi Penelitian*, 2(5), 1469–1474.