

# Sistem Rekomendasi Destinasi Wisata di Kota Batu Menggunakan *Neural Collaborative Filtering*

Yuliana Aristantia<sup>1,\*</sup>, Yunifa Miftachul Arif<sup>2</sup>, Zainal Abidin<sup>3</sup>

Program Studi Magister Informatika, Universitas Islam Negeri Maulana Malik Ibrahim, Indonesia

<sup>1</sup>yulianatantia31@gmail.com; <sup>2</sup>yunif4@ti.uin-malang.ac.id; <sup>3</sup>zainal@ti.uin-malang.ac.id;

\* penulis korespondensi

## INFO ARTIKEL

### Sejarah Artikel

Diterima: 10 November 2024  
Direvisi: 30 Desember 2024  
Diterbitkan: 31 Desember 2024

### Kata Kunci

Destinasi Wisata  
*Neural Collaborative Filtering*  
Sistem Rekomendasi

## ABSTRAK

Penelitian ini bertujuan untuk mengembangkan sistem rekomendasi destinasi wisata di Kota Batu, Jawa Timur, menggunakan metode *Neural Collaborative Filtering* (NCF). Sistem ini dirancang untuk membantu wisatawan memilih destinasi yang sesuai dengan preferensi mereka, mengatasi tantangan keberagaman kebutuhan wisatawan dalam industri pariwisata yang terus berkembang. Dataset yang digunakan mencakup data pengunjung, preferensi wisata, dan informasi 14 lokasi wisata populer di Kota Batu. Proses implementasi melibatkan persiapan dataset, pembentukan *matriks embedding*, dan pelatihan model menggunakan TensorFlow. Evaluasi dilakukan menggunakan metrik akurasi, *precision*, *recall* berdasarkan *Confusion Matrix*. Hasil penelitian menunjukkan bahwa sistem rekomendasi berbasis NCF memiliki performa akurasi yang baik, dengan akurasi tertinggi sebesar 81% pada skenario pembagian *data training* 80% dan *data testing* 20%. Skenario lainnya, seperti 85:15, menghasilkan akurasi 80%, yang mengindikasikan kemampuan generalisasi model yang memadai. Namun, akurasi menurun pada skenario dengan proporsi *data training* yang lebih kecil, seperti 60:40 (67%) dan 70:30 (74%), menegaskan pentingnya dataset yang representatif untuk meningkatkan prediksi. Sistem ini efektif dalam memberikan rekomendasi yang personal dan akurat, sekaligus mendukung promosi pariwisata di Kota Batu. Penelitian ini diharapkan dapat mendorong peningkatan kunjungan wisatawan ke Kota Batu dan berkontribusi pada pertumbuhan industri pariwisata Indonesia.

## PENDAHULUAN

Sektor pariwisata di Indonesia telah menunjukkan pertumbuhan yang signifikan dalam beberapa tahun terakhir, baik dari segi jumlah wisatawan lokal maupun mancanegara [1]. Kota Batu, yang terletak di Jawa Timur, telah bertransformasi menjadi salah satu destinasi wisata unggulan di Indonesia, bersaing dengan destinasi populer lainnya seperti Bali dan Yogyakarta. Dengan keindahan alam, berbagai tempat rekreasi, dan budaya yang khas, Kota Batu menarik perhatian wisatawan dari berbagai kalangan. Pemerintah Kota Batu secara aktif mempromosikan potensi pariwisata daerahnya, yang bertujuan untuk meningkatkan jumlah kunjungan wisatawan serta memperkuat citra kota sebagai destinasi wisata utama di Indonesia. Namun, keberagaman preferensi wisatawan menciptakan tantangan tersendiri dalam memberikan pengalaman perjalanan yang personal dan memuaskan [2]. Banyaknya pilihan destinasi wisata sering kali membuat wisatawan kesulitan memilih lokasi yang paling sesuai dengan preferensi mereka [3]. Oleh karena itu, diperlukan sebuah sistem rekomendasi yang dapat memberikan solusi personal bagi wisatawan, berdasarkan pola dan preferensi individu. *Neural Collaborative Filtering* (NCF) adalah salah satu metode yang menjanjikan

untuk membangun sistem rekomendasi ini. Dengan menggabungkan kekuatan jaringan syaraf tiruan dan data kolaborasi pengguna, NCF mampu menangkap pola preferensi yang kompleks, memberikan rekomendasi yang lebih akurat dibandingkan metode tradisional [4].

Penelitian sebelumnya telah membuktikan keberhasilan NCF di berbagai domain seperti rekomendasi film [5][6], *e-book* [7], *marketplace* [8] dan sistem rekomendasi produk [9]. NCF juga terbukti handal dalam berbagai prediksi [10]. Dalam konteks pariwisata, penerapan NCF masih terbatas, namun menunjukkan potensi besar untuk memberikan rekomendasi destinasi wisata dengan akurasi yang tinggi. Penelitian ini bertujuan untuk mengembangkan sistem rekomendasi destinasi wisata di Kota Batu menggunakan metode NCF. Sistem ini diharapkan mampu merekomendasikan destinasi wisata yang tidak hanya sesuai dengan preferensi wisatawan tetapi juga menyertakan informasi tambahan seperti akomodasi yang berdekatan secara geografis. Selain memberikan solusi bagi wisatawan, penelitian ini juga diharapkan dapat mendukung upaya promosi pariwisata di Kota Batu dan berkontribusi pada pertumbuhan industri pariwisata di Indonesia. Dengan mengadopsi teknologi ini, Kota Batu dapat memperkuat posisinya sebagai destinasi unggulan, menarik wisatawan dari dalam maupun luar negeri, serta memberikan nilai tambah bagi sektor pariwisata nasional.

## METODE

Berdasarkan Gambar 1, penelitian diawali dengan persiapan dataset, membuat *crosstab*, menyusun *embedding matrix*, penyusunan *batch*, membuat *embedding network*, menyusun *schedule*, *training process* dan evaluasi.



Gambar 1. Alur Penelitian

### Persiapan Dataset

Pengumpulan data merupakan awal yang sangat penting dalam penelitian ini. Tanpa data penelitian ini tidak dapat berjalan. Dan persiapan untuk mengumpulkan data perlu dipersiapkan untuk mendapatkan data yang berkualitas. Data yang digunakan pada penelitian ini yaitu data yang diperoleh dari hasil observasi ke tempat wisata utama di Kota Wisata Batu. Data yang digunakan dalam penelitian ini diperoleh dalam kurun waktu 1 bulan pada bulan 5 Maret sampai 6 April 2024. Observasi dilakukan di beberapa titik lokasi utama yaitu area Jawa Timur Park, Alun-alun Kota Batu dan Coban Talun. Wawancara dilakukan kepada 227 pengunjung secara acak. Dari hasil wawancara tersebut dapat disimpulkan dan

dapat dipetakan 14 lokasi utama destinasi wisata di Kota Batu. Lokasi wisata tersebut disajikan pada Tabel 1.

Tabel 1. Daftar Destinasi Wisata

Output	Item Destinasi Wisata
Y1	Jawa Timur <i>Park 1 (Jatim Park 1)</i>
Y2	Jawa Timur <i>Park 2 (Jatim Park 2)</i>
Y3	Jawa Timur <i>Park 3 (Jatim Park 3)</i>
Y4	Museum Angkut
Y5	Taman Rekreasi Selecta
Y6	BNS ( <i>Batu Night Spectacular</i> )
Y7	EGP ( <i>Eco Green Park</i> )
Y8	Alun-alun Kota Batu
Y9	Kusuma Agro
Y10	Pemandian Cangar
Y11	Coban Talun
Y12	Pemandian Songgoriti
Y13	Coban Rais
Y14	PFP ( <i>Predator Fun Park</i> )

### Membuat *Crosstab*

Tahap membuat *crosstab* bertujuan untuk memvisualisasikan user dan item dalam mendapatkan rating tertinggi. Pembuatan *crosstab* mendefinisikan matrik interaksi item user dari hasil umpan balik user. Sehingga dalam kasus penelitian ini akan ditentukan kunjungan destinasi favorit *user* atau top destinasi dan top kategori dari masing-masing data yang telah dipersiapkan.

### Membuat *Embedding Matrix*

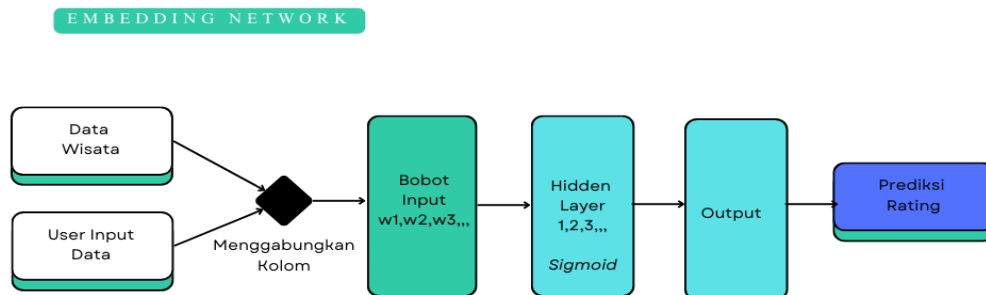
Tahapan yang dilakukan pada penyusunan *embedding matrix* bertujuan untuk mengubah bentuk data yang awalnya berupa *crosstab* menjadi bentuk matrik. *Embedding matrix* merepresentasikan bentuk data menjadi model yang mudah dipahami dalam melihat hubungan vektor data dan merupakan representasi dimensi yang lebih rendah.

### Menyusun *Batch*

Tahap penyusunan *batch* bertujuan membagi dataset menjadi bagian kecil dengan melakukan *split dataset*. Pembagian dataset nantinya akan digunakan selama proses *training*. Proses dalam penyusunan *batch* dilakukan dengan membuat parameter untuk merepresentasikan objek yang akan dibuat, yaitu  $x$  atau dataset dalam bentuk *embedding matrik*,  $y$  atau target rating. Jumlah data yang akan diambil pada setiap *batch* (*batch\_size*) dengan ukuran 10, mengambil sejumlah data secara acak, dan mengubah bentuk input  $x$  dan  $y$  yang semula matrik menjadi sebuah array. Matrik  $x$  akan diacak sepanjang indeks pertamanya, indeks  $x$  atau dataset dalam bentuk *embedding matrik* adalah 0. Lalu dilakukan *self inisialization* terhadap  $x$ ,  $y$ , *batch\_size*, *shuffle*, dan jumlah ukuran  $x$  terhadap *batch*, langkah tersebut dilakukan dengan tujuan untuk membuat kelas yang menunjukkan iterasi pada dataset satu persatu, kemudian dilakukan perulangan. Secara praktik bagian ini akan menggunakan *tensorflow* dalam implementasinya.

## Menyusun *Embedding Network*

Adapun proses pada *embedding network* secara lengkap adalah sebagaimana Gambar 2. Secara garis besar, proses skema *embedding network* memiliki kemiripan dalam proses membuat *embedding matriks* dan menyusun *batch* proses *training*. Kemiripan proses awal terutama bagian proses persiapan data sampai proses bobot *input* atau *weight data*.



Gambar 2. Skema *Embedding Network*

Pada Gambar 2, setiap proses akan dirangkum dalam penjelasan berikut:

### 1. Data Wisata

Data wisata merupakan data baku utama dalam proses implementasi embedding network. Untuk itu sebelum implementasi dimulai, *fixed data* sangat fundamental. Data awal yang berbentuk Microsoft Excel harus dikonversi dahulu ke bentuk format CSV file.

### 2. User Input Data

Bagian ini adalah keterlibatan peneliti (*user*) dalam implementasi. Data yang sudah jadi oleh user akan diinput dalam hal ini adalah proses memasukkan data wisata yang sudah jadi (*fixed data*) ke dalam sistem. Sistem yang digunakan adalah menggunakan *google colab*. Data wisata dan User Input Data ini adalah nilai  $x$  yang akan dihitung untuk mendapatkan nilai prediksi atau  $y$ .

### 3. Menggabungkan Kolom

Pada bagian ini, proses menggabungkan kolom merupakan langkah dimana data yang sudah ada (*fixed data*) akan ditambah kolom berupa UserId. Id user yang masuk dalam kategori pengelompokan tersebut bersifat unik. Data userId yang tidak kosong tersebut akan menjadi sebuah *Unique User*, kemudian unique user tersebut akan dijadikan bentuk data New User berdasarkan rating dan userId. Hal yang sama juga dilakukan dengan data wisata yang menjadi sebuah *Unique Wisata* kemudian dijadikan bentuk data New Wisata berdasarkan rating dan wisataId.

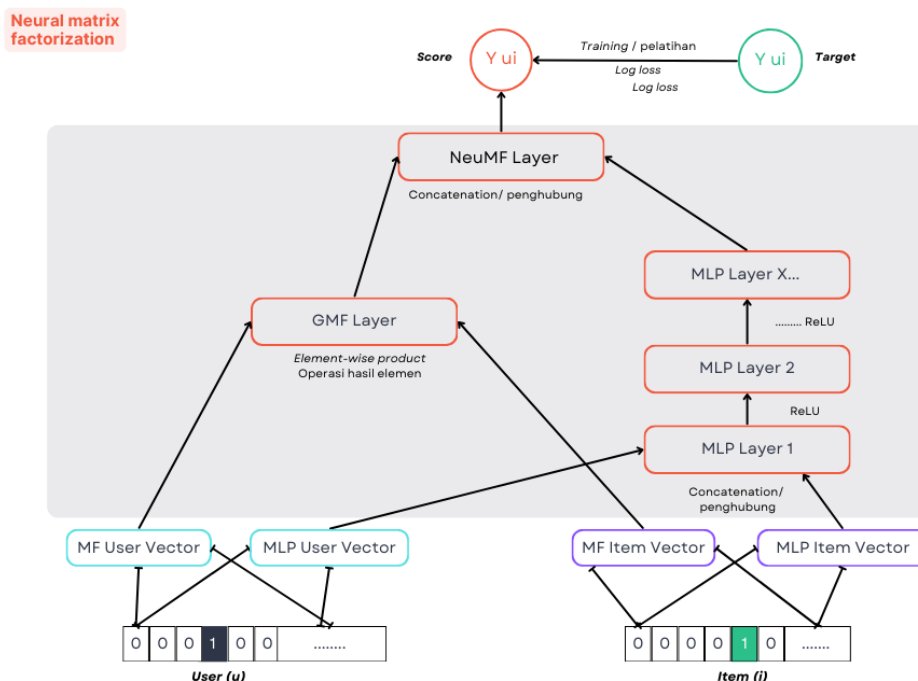
Seperti pada penjelasan tahapan sebelumnya data yang digabungkan ini akan membentuk nilai objek data atau  $x$ . Yang kemudian dihitung untuk mendapatkan nilai prediksi atau  $y$ .

### 4. Bobot *Input*

Bobot *input* atau *weight* yang juga dapat disingkat dengan  $w$ , merupakan proses perhitungan berat dari data per *user*. Nilai bobot ini berupa nilai vektor yang didapatkan dari gabungan dari *fixed data* dan user input data. Yang pada proses selanjutnya akan dilakukan *batch* proses perhitungan data latihan (*training*) dan data uji (*testing*).

### 5. Hidden Layer

Hidden layer merupakan lapisan (*layer*) yang terletak di antara lapisan input dan output jaringan saraf tiruan. Tugas lapisan ini adalah mengekstrak fitur yang kompleks dan abstrak dari data yang telah diteruskan oleh lapisan input. Layer ini digambarkan pada area yang berwarna abu-abu pada Gambar 3.



Gambar 3. Neural Matrix Factorization

Pada Gambar 3, proses *data training* dan *testing* akan di-breakdown berdasarkan layer. NeuMF Layer atau *neural matrix factorization* (NMF) adalah perpanjangan dari metode faktorisasi matriks tradisional yang menggabungkan jaringan saraf untuk menangkap pola dan hubungan yang lebih kompleks dalam data. Faktorisasi matriks (MF) biasanya digunakan dalam sistem pemfilteran dan rekomendasi kolaboratif [11], yang tujuannya adalah untuk memprediksi entri yang hilang dalam nilai matrik *user* dan *input*.

6. Output

Output dalam implementasi penelitian secara umum menggambarkan nilai atau score yang biasanya dalam bentuk prosentase. Dalam studi kasus ini, peneliti menggunakan confusion matrix dalam menggambarkan nilai output. Dalam gambar 3.5 digambarkan dalam nilai Y<sub>ui</sub>.

7. Prediksi Rating Nilai

Prediksi rating adalah hasil output dari proses neural collaborative filtering yang merupakan hasil dari olah nilai vektor pada proses sebelumnya.

**Evaluasi Kinerja Algoritma**

Evaluasi kinerja *Neural Collaborative Filtering* yaitu menggunakan beberapa parameter yaitu *accuracy*, *precision* dan *recall* dengan menggunakan Confusion Matrix untuk mengetahui seberapa banyak data yang berhasil di klasifikasi dan yang gagal diklasifikasi menggunakan metode NCF. Untuk bentuk dasar Confusion Matrix dapat dilihat pada Tabel 2.

Tabel 2. Confusion Matrix

Prediksi	Destinasi Disukai (Positive)	Destinasi Tidak Disukai (Negative)
Actual Negative	True Negative (TN)	False Positive (FP)
Actual Positive	False Negative (FN)	True Positive (TP)

Adapun metrik evaluasi yang dihitung antara lain:

1. *Accuracy*  
Merupakan rasio prediksi Benar (positif dan negatif) dengan keseluruhan data dengan perhitungan,  $accuracy = (TP + TN) / (TP + FP + FN + TN)$ .
2. *Precision*  
Merupakan rasio prediksi benar positif dibandingkan dengan keseluruhan hasil yang diprediksi positif. Perhitungannya adalah  $precision = (TP) / (TP + FP)$ .
3. *Recall (Sensitivitas)*  
Merupakan rasio prediksi benar positif dibandingkan dengan keseluruhan data yang benar positif. Recall menjawab pertanyaan dengan hitungan  $recall = (TP) / (TP + FN)$ .

## HASIL DAN PEMBAHASAN

Penelitian ini menggunakan Bahasa Pemrograman Python yang dijalankan pada *tools* Google Collaboration. Langkah penelitian dimulai dari *import library* atau modul yang digunakan dalam berbagai tahapan dalam penelitian ini. Tahapan yang dilakukan antara lain melakukan *pre-processing* dataset, yakni data wawancara wisata. Selanjutnya, dilakukan visualisasi data. Kemudian, melakukan modeling Algoritma *Neural Collaborative Filtering*. Hingga diakhiri dengan langkah terakhir berupa evaluasi menggunakan *confusion matrix*. Pre-processing data dilakukan dengan embedding matriks. Pada proses ini akan muncul nilai statistik dimana data akan tersaji berisi nilai tertinggi sampai nilai terendah dari lokasi favorit tempat wisata sebagaimana disajikan pada Gambar 4.

	count	mean	std	min	25%	50%	75%	max
Y1	227.0	0.612335	0.488294	0.0	0.0	1.0	1.0	1.0
Y2	227.0	0.638767	0.481420	0.0	0.0	1.0	1.0	1.0
Y3	227.0	0.568282	0.496410	0.0	0.0	1.0	1.0	1.0
Y4	227.0	0.502203	0.501100	0.0	0.0	1.0	1.0	1.0
Y5	227.0	0.511013	0.500983	0.0	0.0	1.0	1.0	1.0
Y6	227.0	0.374449	0.485050	0.0	0.0	0.0	1.0	1.0
Y7	227.0	0.286344	0.453051	0.0	0.0	0.0	1.0	1.0
Y8	227.0	0.541850	0.499347	0.0	0.0	1.0	1.0	1.0
Y9	227.0	0.140969	0.348759	0.0	0.0	0.0	0.0	1.0
Y10	227.0	0.268722	0.444275	0.0	0.0	0.0	1.0	1.0
Y11	227.0	0.193833	0.396173	0.0	0.0	0.0	0.0	1.0
Y12	227.0	0.171806	0.378046	0.0	0.0	0.0	0.0	1.0
Y13	227.0	0.105727	0.308167	0.0	0.0	0.0	0.0	1.0
Y14	227.0	0.083700	0.277550	0.0	0.0	0.0	0.0	1.0

Gambar 4. Hasil Data Statistik

Pada Gambar 4, sebenarnya sudah diketahui tempat wisata paling favorit sampai bukan yang favorit. Yang bisa digunakan untuk menyusun rating penilaian secara cepat. Proses selanjutnya adalah menyusun *schedule* yang mana digunakan untuk menyusun rating atau data yang sering muncul, sebelum proses implementasi *training* dan *testing* dilakukan. Sebelum implementasi *training* diatur jumlah proporsi data antara *data training* dan *testing*. Dalam penelitian ini menggunakan dataset sejumlah 227 data dengan output item 14 kategori. Penerapan *data training* dan *testing* adalah 80% dan 20% menggunakan model scikit learn dari python. Dalam uji coba train test split juga akan menggunakan beberapa uji coba persentase data 70% dan 30%, 90% dan 10%, 60% dan 40%, juga 85% dan 15%.

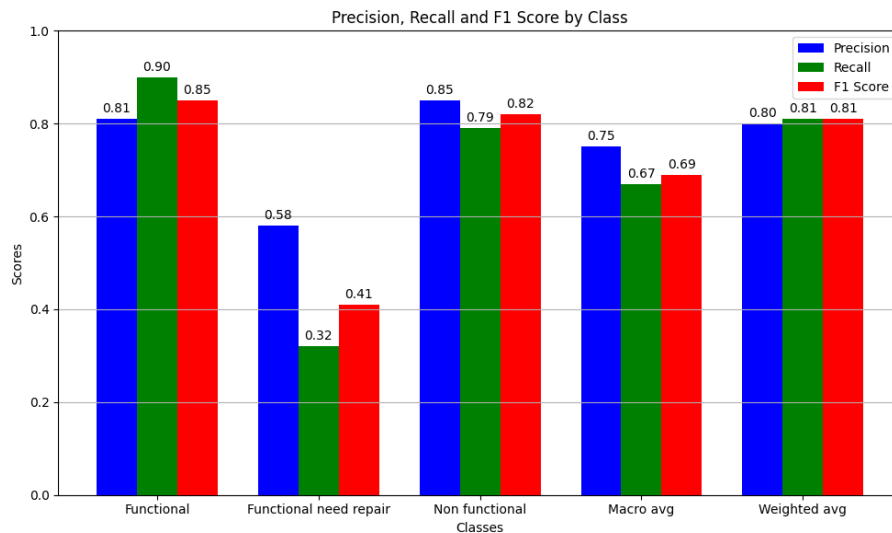
### Evaluasi Kinerja Algoritma

Hasil dari implementasi metode *Neural Collaborative Filtering* dapat evaluasi dari nilai *confusion matrix* yang menghitung nilai dari *precision*, *recall*, dan *f1 score*. Adapun detail nilai dari beberapa ujicoba adalah sebagai berikut:

#### Ujicoba persentase 80% *data training* dan 20% *data testing*

Tabel 3. Hasil Ujicoba 1

	<i>Precision</i>	<i>Recall</i>	<i>f1-score</i>
<i>Functional</i>	0.81	0.90	0.85
<i>Functional need repair</i>	0.58	0.32	0.41
<i>Non functional</i>	0.85	0.79	0.82
<i>Accuracy</i>			0.81
<i>Macro avg</i>	0.75	0.67	0.69
<i>Weighted avg</i>	0.81	0.81	0.81



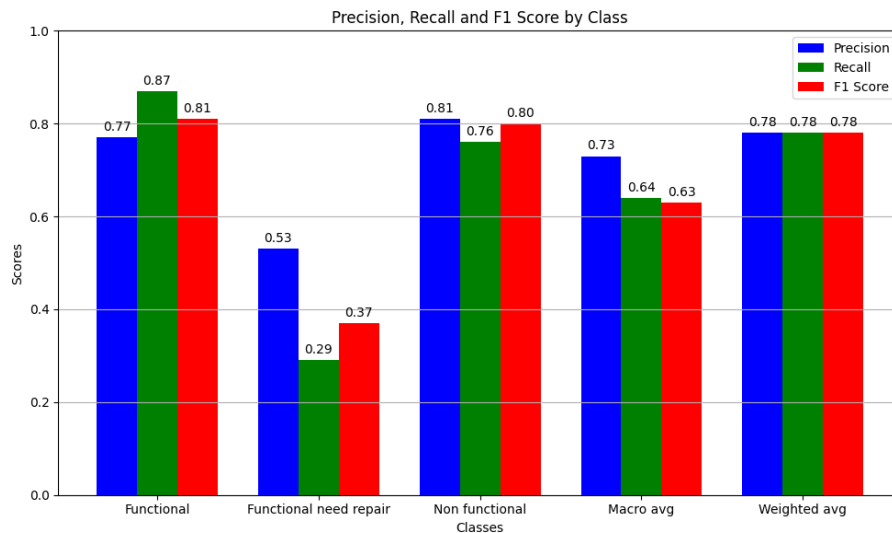
Gambar 5. Hasil Ujicoba 1

Data yang digunakan dalam Tabel 3 dan Gambar 5 menunjukkan data yang dieksekusi adalah 80% untuk data latih atau *training*. Dan untuk data uji atau *data testing* sebesar 20%. Hasil menunjukkan bahwa nilai akurasi yang didapatkan adalah sebesar 81% dan *weighted average* juga sebesar 81%. Sedangkan *macro average* tertinggi sebesar 75% dan terendah 67%. Untuk bagian *functional*, *functional need repair* dan *non functional* didapatkan nilai terendah adalah 32% dibagian *recall functional need repair*. Dan nilai tertinggi adalah 85% pada bagian *functional precision* dan *non functional f1-score*.

**Ujicoba persentase 90% data training dan 10% data testing**

Tabel 4. Hasil Ujicoba 2

	<i>Precision</i>	<i>Recall</i>	<i>f1-score</i>
<i>Functional</i>	0.77	0.87	0.81
<i>Functional need repair</i>	0.53	0.29	0.37
<i>Non functional</i>	0.81	0.76	0.80
<i>Accuracy</i>			0.78
<i>Macro avg</i>	0.73	0.64	0.63
<i>Weighted avg</i>	0.78	0.78	0.78



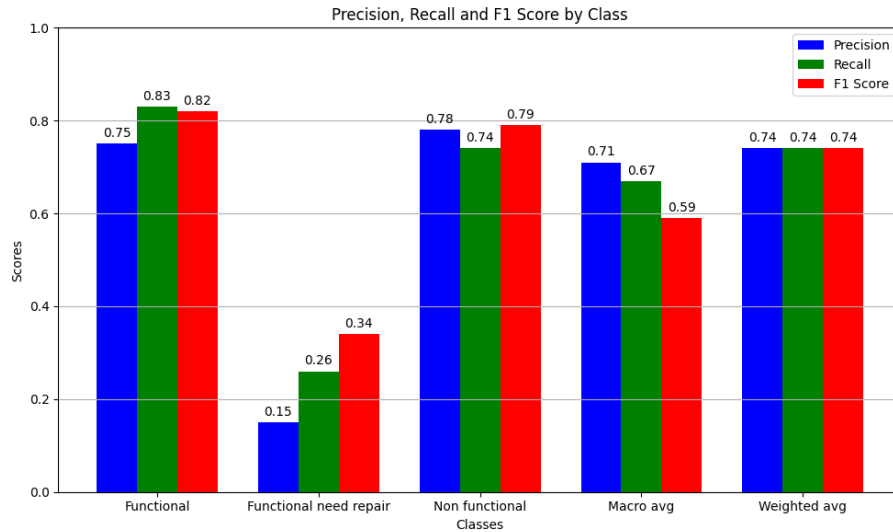
Gambar 6. Hasil Ujicoba 2

Pada Tabel 4 dan Gambar 6 menunjukkan bahwa data untuk *training* adalah 90% dan *testing* sebesar 10%. Dalam perhitungan confusion matrix dihasilkan nilai tertinggi akurasi adalah 78%. Sedangkan pada bagian *precision*, *recall*, *f1-score*, nilai tertinggi adalah 87% pada bagian *recall functional*. Dan nilai terendah adalah pada *recall functional need repair* dengan 29%. Bagian average baik pada bagian macro dan weighted didapatkan hasil 78% nilai tertinggi dan terendah 63% pada f1-score nilai *macro average*.

**Ujicoba persentase 70% data training dan 30% data testing**

Tabel 5. Hasil Uji coba 3

	<i>Precision</i>	<i>Recall</i>	<i>f1-score</i>
<i>Functional</i>	0.75	0.83	0.82
<i>Functional need repair</i>	0.15	0.26	0.34
<i>Non functional</i>	0.78	0.74	0.79
<i>Accuracy</i>			0.74
<i>Macro avg</i>	0.71	0.67	0.59
<i>Weighted avg</i>	0.74	0.74	0.74



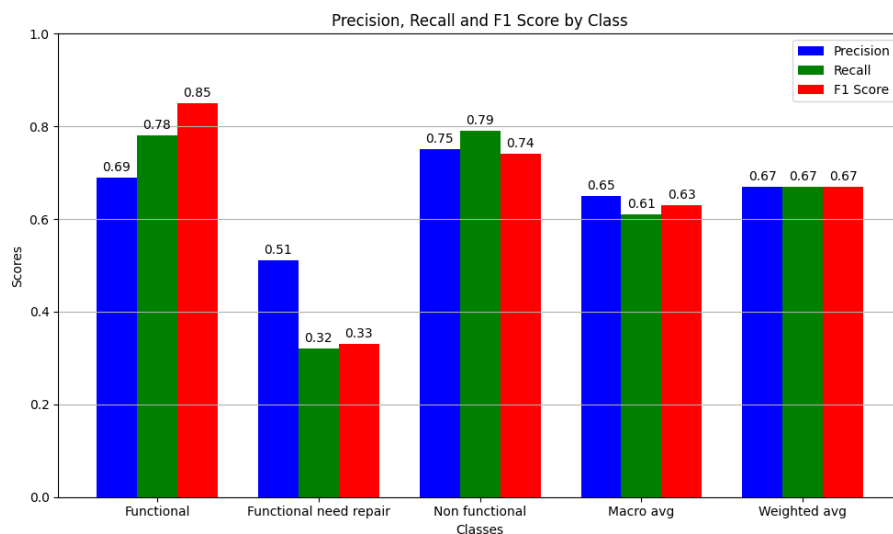
Gambar 7. Hasil Ujicoba 3

Pada Tabel 5 dan Gambar 7 menunjukkan hasil akurasi adalah 74%. Hasil ini menggunakan *data training* sebesar 70% dan 30% untuk *data testing*. Dalam tabel maupun gambar grafik dari semua kategori nilai tertinggi adalah 83% untuk recall functional dan terendah adalah 15% untuk precision bagian functional need repair.

**Ujicoba persentase 60% *data training* dan 40% *data testing***

Tabel 6. Hasil Ujicoba 4

	<i>Precision</i>	<i>Recall</i>	<i>f1-score</i>
<i>Functional</i>	0.69	0.78	0.85
<i>Functional need repair</i>	0.51	0.32	0.33
<i>Non functional</i>	0.75	0.79	0.74
<i>Accuracy</i>			0.67
<i>Macro avg</i>	0.65	0.61	0.63
<i>Weighted avg</i>	0.67	0.67	0.67



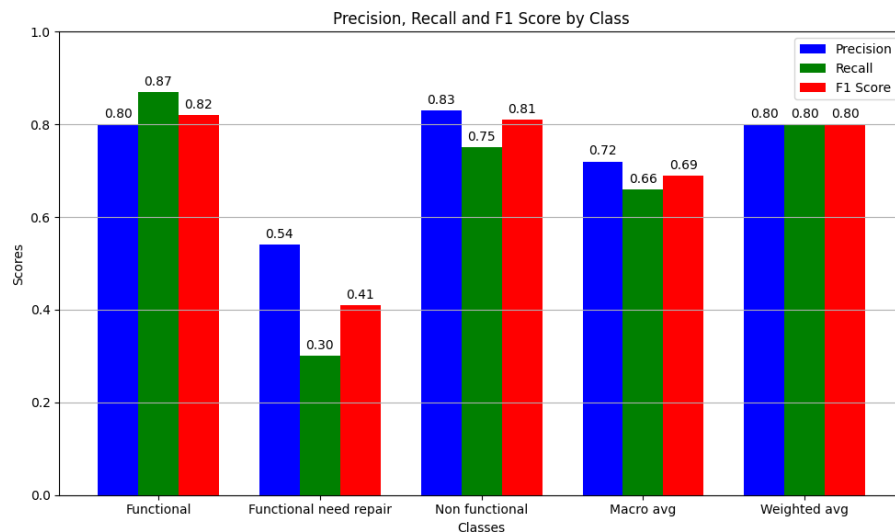
Gambar 8. Hasil Ujicoba 4

Pada Tabel 6 dan Gambar 8 menunjukkan hasil akurasi adalah 67%. Hasil ini menggunakan *data training* sebesar 60% dan 40% untuk *data testing*. Dalam tabel maupun gambar grafik dari semua kategori nilai tertinggi adalah 85% untuk f1-score functional dan terendah adalah 32% untuk recall bagian *functional need repair*.

### Ujicoba persentase 85% *data training* dan 15% *data testing*

Tabel 7. Hasil Ujicoba 5

	<i>Precision</i>	<i>Recall</i>	<i>f1-score</i>
<i>Functional</i>	0.80	0.87	0.82
<i>Functional need repair</i>	0.54	0.30	0.41
<i>Non functional</i>	0.83	0.75	0.81
<i>Accuracy</i>			0.80
<i>Macro avg</i>	0.72	0.66	0.69
<i>Weighted avg</i>	0.80	0.80	0.80



Gambar 9. Hasil Ujicoba 5

Pada Tabel 7 dan Gambar 9 menunjukkan hasil akurasi adalah 80%. Hasil ini menggunakan *data training* sebesar 85% dan 15% untuk *data testing*. Dalam tabel maupun gambar grafik dari semua kategori nilai tertinggi adalah 87% untuk *recall functional* dan terendah adalah 30% untuk recall bagian *functional need repair*.

### Pembahasan

Tabel 8. Hasil perbandingan akurasi

Dataset	Skenario Ujicoba		Hasil Akurasi
	<i>Data training</i>	<i>Data testing</i>	
227 data	80 %	20 %	81 %
	90 %	10 %	78 %
	70 %	30 %	74 %
	60 %	40 %	67 %
	85 %	15 %	80 %

Berdasarkan Tabel 8, didapatkan hasil nilai akurasi sebesar 81% dengan menggunakan data uji coba persentase data 80% dan 20%. Nilai akurasi tertinggi kedua adalah menggunakan persentase 85% dan 15% yang menghasilkan akurasi sebesar 80%. Dan data

dengan nilai terendah yang menghasilkan akurasi sebesar 67% adalah data dengan uji coba 60% dan 40%. Nilai tertinggi akurasi 81% ini bisa dikategorikan baik dan memiliki error yang cukup rendah. Dengan nilai ini diharapkan membantu wisatawan dalam memilih rekomendasi tempat di Kota Batu lebih akurat.

## KESIMPULAN

Penelitian ini berhasil mengembangkan sistem rekomendasi destinasi wisata menggunakan metode *Neural Collaborative Filtering* (NCF) untuk membantu wisatawan dalam memilih destinasi yang sesuai dengan preferensi mereka. Berdasarkan pengujian dengan berbagai skenario pembagian data, sistem menunjukkan performa akurasi yang cukup baik. Dengan pembagian *data training* sebesar 80% dan *data testing* sebesar 20%, akurasi terbaik sebesar 81% berhasil dicapai. Skenario lainnya, seperti pembagian data 85:15, juga menghasilkan akurasi tinggi sebesar 80%, menunjukkan bahwa metode ini memiliki kemampuan generalisasi yang baik. Namun, penurunan akurasi terlihat pada skenario dengan proporsi *data training* yang lebih kecil, seperti 60:40 (67%) dan 70:30 (74%), mengindikasikan bahwa metode NCF membutuhkan dataset training yang lebih besar untuk menghasilkan prediksi yang lebih akurat. Hal ini mencerminkan pentingnya data yang representatif dalam melatih model. Secara keseluruhan, sistem rekomendasi ini menunjukkan potensi besar dalam memberikan rekomendasi destinasi wisata yang personal dan akurat. Selain membantu wisatawan, sistem ini diharapkan dapat mendukung promosi pariwisata, khususnya di Kota Batu, sekaligus berkontribusi pada pertumbuhan industri pariwisata di Indonesia.

## REFERENSI

- [1] Aisha Kusuma Putri and Suliadi, "Rekomendasi Destinasi Wisata di Indonesia Menggunakan Metode Item2Vec," *J. Ris. Stat.*, pp. 11–18, 2023, doi: 10.29313/jrs.v3i1.1770.
- [2] Y. S. Pasaribu *et al.*, "Rekomendasi Destinasi Wisata Kota Bandung Menggunakan Algoritma Collaborative Filtering Timotius Selar Sitompul," *J. Penelit. dan Karya Ilm.*, vol. 1, no. 6, 2023, [Online]. Available: <https://doi.org/10.59059/mutiara.v1i6.736>
- [3] A. Adil, R. A. Dwiputri, and B. K. Triwijoyo, "Aplikasi Spasial Rekomendasi Wisata Terdekat dengan Metode Haversine Berbasis Mobile," *J. Bumigora Inf. Technol.*, vol. 4, no. 1, pp. 95–106, 2022, doi: 10.30812/bite.v4i1.1948.
- [4] M. S. Negara and A. Z. Mardiansyah, "Implementasi Machine Learning dengan Metode Collaborative Filtering dan Content-Based Filtering pada Aplikasi Mobile Travel (Bangkit Academy)," *J. Begawe Teknol. Inf.*, vol. 5, no. 1, pp. 126–136, 2024, doi: 10.29303/jbegati.v5i1.1193.
- [5] N. K. Ayyiyah, R. Kusumaningrum, and R. Rismiyati, "Film Recommender System Menggunakan Metode *Neural Collaborative Filtering*," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 10, no. 3, pp. 699–708, 2023, doi: 10.25126/jtiik.20231036616.
- [6] D. A. Nugroho, C. Lubis, and N. J. Perdana, "Sistem Rekomendasi Film Menggunakan Metode *Neural Collaborative Filtering* Movie Recommendation System Using *Neural Collaborative Filtering*," *J. Inf. Technol. Comput. Sci.*, vol. 7, no. 3, pp. 6765–6775, 2024.
- [7] J. Y. Kim and C. K. Lim, "Feature Extracted Deep *Neural Collaborative Filtering* for E-Book Service Recommendations," *Appl. Sci.*, vol. 13, no. 11, 2023, doi: 10.3390/app13116833.
- [8] A. Faizin and I. Surjandari, "Product recommender system using *Neural Collaborative Filtering* for marketplace in indonesia," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 909, no. 1, 2020, doi: 10.1088/1757-899X/909/1/012072.
- [9] A. Kwofie, M. Hamada, and R. Prasad, "*Neural Collaborative Filtering* and Autoencoder-Enabled Deep Learning Models for Recommender Systems," *AIP Conf. Proc.*, vol. 2909, no. 1, 2023, doi: 10.1063/5.0182170.

- 
- [10] J. Bobadilla, A. Gutiérrez, S. Alonso, and Á. González-Prieto, “Neural Collaborative Filtering Classification Model to Obtain Prediction Reliabilities,” *Int. J. Interact. Multimed. Artif. Intell.*, vol. 7, no. 4, pp. 18–26, 2022, doi: 10.9781/ijimai.2021.08.010.
- [11] B. Juarto and A. S. Girsang, “Neural Collaborative with Sentence BERT for News Recommender System,” *Int. J. Informatics Vis.*, vol. 5, no. 4, pp. 448–455, 2021, doi: 10.30630/JOIV.5.4.678.
-