

# Website e-commerce sepatu R&R dengan fitur sistem rekomendasi

Mahendra Kirana M.B <sup>1</sup>, Henokh Abhinaya Tjahjadi <sup>2</sup>, Ammar Tyo Pasaribu <sup>3</sup>, Muhammad Aditya Permana <sup>3</sup>, Muh.Yusuf Fikry <sup>4</sup>, Andi Alisha Faiqihah <sup>5</sup>

<sup>1</sup> Universitas Hasanuddin; mahendra22h@student.unhas.ac.id  
<sup>2</sup> Universitas Hasanuddin; Tjahjadiha22h@student.unhas.ac.id  
<sup>3</sup> Universitas Hasanuddin; pasaribuat22h@student.unhas.ac.id  
<sup>4</sup> Universitas Hasanuddin; Permanama22h@student.unhas.ac.id  
<sup>5</sup> Universitas Hasanuddin; Fikrymy22h@student.unhas.ac.id  
<sup>6</sup> Universitas Hasanuddin; Faiqihahaa22h@student.unhas.ac.id  
\* Correspondence: pasaribuat22h@student.unhas.ac.id; Tel.: +62-895-6201-63633 (F.L.)

## 1. Project Charter

## 2. Methods

### 2.1. Data Collection

Pengumpulan data dilakukan melalui penyusunan manual dan pencatatan interaksi pengguna untuk membangun dataset produk sepatu serta dataset interaksi pengguna. Langkah-langkah yang dilakukan adalah sebagai berikut:

#### Identifikasi Sumber Data:

- Dataset sepatu disusun dengan memasukkan produk dari lima kategori utama: formal, sport, heels, boots, dan casual.
- Setiap kategori terdiri dari 10 jenis sepatu berbeda, sehingga total terdapat 50 produk sepatu dalam dataset.

#### Pembuatan Dataset Produk:

- Dataset produk sepatu dibuat secara manual dengan menentukan nama sepatu, kategori, dan informasi dasar lainnya.
- Masing-masing sepatu diberi *id\_sepatu* sebagai identitas unik untuk digunakan dalam pemetaan data interaksi.

#### Pengumpulan Data Interaksi Pengguna:

- Dataset interaksi pengguna (*user\_interaction*) berisi data yang menunjukkan aktivitas pengguna pada setiap produk sepatu di situs.
- Data interaksi ini meliputi *id\_user*, *id\_sepatu*, dan jenis *interaction*, seperti *view*, *cart*, *buy*, dan *wishlist*, yang dicatat berdasarkan aktivitas pengguna.

#### Penyimpanan Data:

- Setiap data interaksi pengguna disimpan dalam format yang terstruktur di dalam dataset *user\_interaction*.
- Data ini diatur agar mudah diakses oleh model rekomendasi untuk melatih sistem berdasarkan aktivitas pengguna yang relevan.

### 2.2. Data Explanation

Pada proyek ini, terdapat dua dataset utama yang digunakan untuk melatih model rekomendasi sepatu, yaitu Dataset Produk dan Dataset Interaksi Pengguna. Berikut adalah penjelasan lebih lanjut tentang kedua dataset ini:

**Citation:** Lastname, F.; Lastname, F.; Lastname, F. Title. *Journal Not Specified*, .

Received:

Accepted:

Published:

**Copyright:** © 2024 by the authors. Submitted to *Journal Not Specified* for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

2.2.1. Dataset Produk

Dataset produk mencakup informasi dasar mengenai sepatu yang tersedia di platform e-commerce. Setiap entri dalam dataset ini mewakili satu produk sepatu dan berisi atribut sebagai berikut:

- **id\_sepatu**: ID unik untuk setiap sepatu. ID ini berfungsi sebagai pengidentifikasi yang akan dipetakan dalam dataset interaksi pengguna.
- **kategori**: Jenis atau kategori sepatu, seperti "formal", "sport", "heels", "boots", atau "casual". Kategori ini membantu mengelompokkan sepatu berdasarkan gaya dan fungsi.
- **nama\_sepatu**: Nama atau model sepatu, yang dapat memberikan konteks bagi pengguna terkait produk yang ditawarkan.

Contoh entri dataset:

id_sepatu	kategori	nama_sepatu
1	formal	Wirken Oxford
2	sport	Ardiles Nfinity Burst
3	heels	Celline Heels
4	boots	Parabellum COBRA
5	casual	Asics Gel Sonoma SE

Dataset produk ini penting untuk memberikan konteks bagi sistem rekomendasi mengenai karakteristik dasar dari setiap produk.

2.2.2. Dataset Interaksi Pengguna (user\_interaction)

Dataset ini berisi data interaksi antara pengguna dan produk di platform, yang mencakup berbagai aktivitas pengguna yang relevan. Atribut-atribut dalam dataset interaksi pengguna meliputi:

- **id\_user**: ID unik pengguna yang melakukan interaksi. Atribut ini membantu dalam pelacakan aktivitas individual pengguna.
- **id\_sepatu**: ID sepatu yang diinteraksikan oleh pengguna, menghubungkan dataset ini dengan dataset produk.

- **user\_interaction** : Jenis interaksi yang dilakukan pengguna pada produk. Nilai ini dapat berupa view (melihat), buy (membeli), wishlist (daftar keinginan), dan cart (keranjang).

Contoh entri dataset:

id_user	id_sepatu	interaction
1	12	view
1	2	buy
2	43	wishlist
3	24	cart
4	15	view

Dataset interaksi pengguna ini penting karena menjadi sumber utama dalam melatih model rekomendasi. Berdasarkan data ini, model dapat belajar pola preferensi pengguna untuk memberikan rekomendasi yang sesuai.

Dengan kedua dataset ini, model rekomendasi dapat dilatih untuk mengenali pola interaksi pengguna dan menawarkan produk yang relevan, meningkatkan pengalaman pengguna serta potensi konversi di platform e-commerce.

2.3. Algorithm or Model

Pada proyek ini, sistem rekomendasi sepatu dibangun menggunakan model *Non-negative Matrix Factorization* (NMF), yang merupakan teknik *Collaborative Filtering*. Berikut adalah langkah-langkah dan detail dari model yang digunakan:

1. Persiapan Data:

- *Dataset Interaksi Pengguna* dan *Dataset Produk* digabungkan untuk membentuk matriks pengguna-produk (*user-item matrix*). Matriks ini berfungsi sebagai dasar untuk pelatihan model.
  - Setiap jenis interaksi pengguna (seperti *view*, *wishlist*, *cart*, dan *buy*) dipetakan ke nilai numerik. Misalnya, *view* diberi nilai 1, *wishlist* diberi nilai 2, *cart* diberi nilai 3, dan *buy* diberi nilai 4. Pemetaan ini membantu dalam mengukur tingkat ketertarikan pengguna terhadap produk.
2. **Pembentukan Matriks:**
- Matriks pengguna-produk dibentuk dengan menggunakan nilai interaksi sebagai elemen matriks. Setiap baris mewakili pengguna, dan setiap kolom mewakili produk. Matriks ini menggambarkan hubungan antara pengguna dan produk berdasarkan interaksi yang terjadi.
  - Nilai kosong dalam matriks diisi dengan nol untuk menunjukkan tidak adanya interaksi. Hal ini penting untuk memastikan bahwa model dapat memproses data dengan benar tanpa kesalahan.
3. **Pelatihan Model NMF:**
- Model NMF dilatih menggunakan matriks pengguna-produk. NMF memfaktorkan matriks ini menjadi dua matriks yang lebih kecil: matriks pengguna-fitur ( $W$ ) dan matriks fitur-produk ( $H$ ). Proses ini membantu dalam mengidentifikasi pola laten dalam data.
  - Matriks  $W$  merepresentasikan preferensi laten pengguna, sedangkan matriks  $H$  merepresentasikan karakteristik laten produk. Dengan kata lain,  $W$  menunjukkan bagaimana setiap pengguna berhubungan dengan fitur-fitur tertentu, dan  $H$  menunjukkan bagaimana setiap produk berhubungan dengan fitur-fitur tersebut.
  - Parameter model seperti jumlah komponen laten ( $n\_components$ ) diatur untuk mengoptimalkan hasil rekomendasi. Pemilihan parameter yang tepat sangat penting untuk meningkatkan akurasi model.
4. **Pemberian Rekomendasi:**
- Untuk pengguna baru, rekomendasi diberikan berdasarkan popularitas produk yang dihitung dari interaksi pengguna lain. Hal ini dilakukan karena tidak ada data interaksi sebelumnya untuk pengguna baru.
  - Untuk pengguna yang sudah ada, skor preferensi dihitung dengan mengalikan matriks  $W$  dan  $H$ , dan produk dengan skor tertinggi direkomendasikan. Proses ini memungkinkan sistem untuk memberikan rekomendasi yang dipersonalisasi berdasarkan preferensi pengguna yang telah dipelajari.
5. **Evaluasi Model:**
- Model dievaluasi menggunakan metrik seperti *Precision@k*, *Recall@k*, dan *Mean Average Precision (MAP)@k* untuk mengukur akurasi rekomendasi. Evaluasi ini penting untuk memastikan bahwa model memberikan rekomendasi yang relevan.
  - Precision mengukur proporsi rekomendasi yang relevan dari semua rekomendasi yang diberikan, sedangkan recall mengukur proporsi item relevan yang berhasil direkomendasikan dari semua item relevan yang tersedia. MAP memberikan gambaran keseluruhan tentang akurasi model dalam memberikan rekomendasi yang relevan.

Dengan pendekatan ini, sistem rekomendasi dapat memberikan saran produk yang lebih relevan kepada pengguna, meningkatkan pengalaman pengguna dan potensi konversi di platform e-commerce.

## 2.4. Testing (Procedures and Metrics)

Pengujian sistem rekomendasi dilakukan untuk memastikan bahwa model memberikan rekomendasi yang akurat dan relevan. Berikut adalah prosedur dan metrik yang digunakan dalam pengujian:

### 1. Prosedur Pengujian:

- *Split Data*: Dataset dibagi menjadi data pelatihan dan data pengujian. Data pelatihan digunakan untuk melatih model, sedangkan data pengujian digunakan untuk mengevaluasi kinerja model.
- *Pelatihan Model*: Model dilatih menggunakan data pelatihan dengan parameter yang telah ditentukan.
- *Prediksi Rekomendasi*: Model menghasilkan rekomendasi untuk pengguna dalam data pengujian.

### 2. Metrik Evaluasi:

- *Precision@k*: Mengukur proporsi rekomendasi yang relevan dari semua rekomendasi yang diberikan hingga posisi ke-k. Precision yang tinggi menunjukkan bahwa model memberikan rekomendasi yang relevan.
- *Recall@k*: Mengukur proporsi item relevan yang berhasil direkomendasikan dari semua item relevan yang tersedia hingga posisi ke-k. Recall yang tinggi menunjukkan bahwa model mampu menangkap sebagian besar item relevan.
- *Mean Average Precision (MAP)@k*: Menghitung rata-rata presisi pada berbagai level cut-off dalam daftar rekomendasi hingga posisi ke-k. MAP memberikan gambaran keseluruhan tentang akurasi model dalam memberikan rekomendasi yang relevan.

### 3. Analisis Hasil:

- Hasil pengujian dianalisis untuk mengidentifikasi kekuatan dan kelemahan model. Analisis ini membantu dalam mengoptimalkan model lebih lanjut.
- Perbandingan dilakukan antara hasil pengujian dan ekspektasi untuk memastikan bahwa model memenuhi tujuan yang diinginkan.

Pengujian ini memastikan bahwa sistem rekomendasi dapat memberikan saran produk yang relevan dan meningkatkan pengalaman pengguna di platform e-commerce.

## 2.5. Evaluation Metrics

### 1. Precision@K:

- Precision pada peringkat  $K$  adalah persentase item yang direkomendasikan dalam  $K$  teratas yang relevan bagi pengguna. Precision berfokus pada relevansi dari rekomendasi dalam daftar  $K$  teratas, bukan seberapa baik rekomendasi mencakup semua item relevan. Dengan kata lain, metrik ini menjawab pertanyaan: Dari semua item yang direkomendasikan dalam  $K$  teratas, berapa banyak yang benar-benar relevan?
- Rumus:

$$\text{Precision@K} = \frac{\text{jumlah item relevan dalam } K \text{ teratas}}{K}$$

- Contoh: Jika kita merekomendasikan 5 item dan 3 di antaranya relevan, maka Precision@5 adalah:

$$\text{Precision@5} = \frac{3}{5} = 0.6$$

### 2. Recall@K:

- Recall pada peringkat  $K$  adalah proporsi item relevan yang berhasil direkomendasikan dalam  $K$  teratas. Metrik ini mengukur seberapa baik model menangkap semua item relevan dalam daftar rekomendasi, tanpa memperhitungkan jumlah item yang direkomendasikan. Recall menjawab pertanyaan: Dari semua item

yang dianggap relevan oleh pengguna, berapa banyak yang muncul dalam  $K$  teratas?

- Rumus:

$$\text{Recall@K} = \frac{\text{jumlah item relevan dalam } K \text{ teratas}}{\text{jumlah total item relevan}}$$

- Contoh: Jika seorang pengguna memiliki 10 item yang relevan tetapi hanya 3 yang muncul dalam  $K$  teratas,  $\text{Recall@K}$  adalah:

$$\text{Recall@10} = \frac{3}{10} = 0.3$$

### 3. Accuracy@K:

- Dalam konteks ini,  $\text{Accuracy@K}$  mengukur persentase item yang direkomendasikan dengan benar (hit) di antara semua item relevan, khususnya dalam  $K$  teratas.  $\text{Accuracy}$  mirip dengan  $\text{Recall}$  tetapi fokus pada perbandingan jumlah item yang direkomendasikan dengan benar relatif terhadap jumlah total item relevan.
- Rumus:

$$\text{Accuracy@K} = \frac{\text{jumlah item relevan yang direkomendasikan dalam } K \text{ teratas}}{\text{jumlah total item relevan}}$$

### 4. Mean Average Precision (MAP@K):

- $\text{MAP@K}$  adalah metrik yang lebih lanjut yang memperhitungkan posisi item relevan dalam daftar rekomendasi.  $\text{MAP@K}$  mengukur precision di setiap posisi dalam peringkat, lalu menghitung rata-ratanya di semua item relevan. Metrik ini berguna untuk kasus di mana urutan rekomendasi berpengaruh karena item relevan yang muncul lebih awal akan mendapatkan skor lebih tinggi.
- Rumus:

$$\text{MAP@K} = \frac{1}{N} \sum_{i=1}^N \text{Average Precision@K}_i$$

- dengan:
  - $N$  adalah jumlah pengguna,
  - $\text{Average Precision@K}_i$  adalah rata-rata precision pada  $K$  untuk setiap pengguna.

### 5. Kemiripan Nilai Antara Precision dan MAP, Recall dan Accuracy:

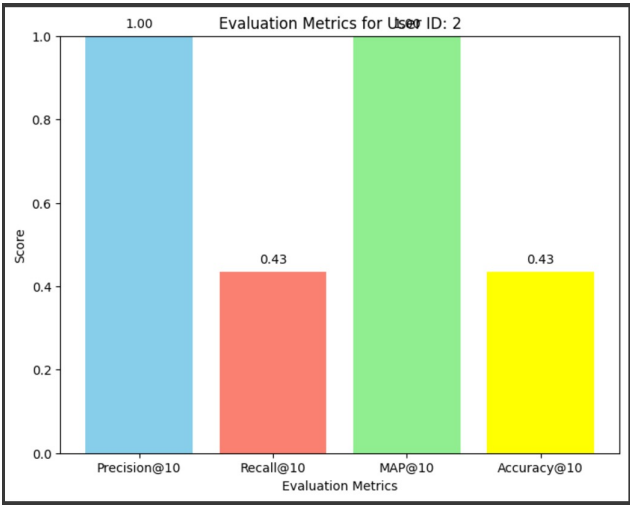
- Berdasarkan dataset yang terdiri dari 500 interaksi pengguna di antara 20 pengguna unik, beberapa pola dapat menjelaskan mengapa precision dan MAP cenderung serupa, dan mengapa nilai recall dan accuracy sering kali mirip.
- *Kemiripan Precision dan MAP*: Baik  $\text{Precision@K}$  maupun  $\text{MAP@K}$  berfokus pada relevansi item dalam  $K$  teratas. Karena MAP memberi bobot lebih tinggi pada item relevan yang muncul di awal daftar rekomendasi, nilai MAP akan cenderung mirip dengan Precision jika sebagian besar item relevan selalu muncul di peringkat atas secara konsisten.
- *Kemiripan Recall dan Accuracy*:  $\text{Recall@K}$  dan  $\text{Accuracy@K}$  keduanya mengukur seberapa baik daftar rekomendasi mencakup item relevan, meskipun dari perspektif yang sedikit berbeda. Recall adalah persentase item relevan yang muncul dalam rekomendasi, sementara akurasi, seperti yang diimplementasikan di sini, juga menghitung hit di antara item relevan.

3. Problem	198
4. Intelligence System	199
4.1. System Architecture	200
Arsitektur sistem e-commerce sepatu ini dirancang untuk mendukung fitur rekomendasi yang efisien dan responsif. Sistem ini terdiri dari beberapa komponen utama yang bekerja secara sinergis untuk memberikan pengalaman pengguna yang optimal.	201-203
1. Frontend:	204
<ul style="list-style-type: none"><li>React.js digunakan untuk membangun antarmuka pengguna yang dinamis dan interaktif. React.js dipilih karena kemampuannya dalam mengelola komponen UI yang kompleks dan memperbarui tampilan secara efisien.</li><li>Komponen frontend bertanggung jawab untuk menampilkan produk, menangani interaksi pengguna seperti pencarian dan filter, serta menampilkan rekomendasi produk secara real-time.</li><li>Antarmuka pengguna dirancang agar responsif dan mudah digunakan, memastikan bahwa pengguna dapat dengan mudah menemukan dan membeli produk yang mereka inginkan.</li></ul>	205-213
2. Backend:	214
<ul style="list-style-type: none"><li>Flask digunakan sebagai framework web untuk mengembangkan API yang menghubungkan frontend dengan database dan model rekomendasi. Flask dipilih karena kesederhanaan dan fleksibilitasnya dalam menangani permintaan HTTP.</li><li>Backend bertanggung jawab untuk memproses permintaan dari frontend, seperti permintaan data produk dan interaksi pengguna, serta mengelola logika bisnis untuk rekomendasi.</li><li>SQLite digunakan sebagai database untuk menyimpan data produk dan interaksi pengguna. SQLite dipilih karena kemudahan penggunaan, integrasi yang mulus dengan Flask, dan kemampuan untuk menangani data dalam skala kecil hingga menengah.</li><li>Database dirancang untuk menyimpan informasi produk, kategori, dan riwayat interaksi pengguna, yang semuanya penting untuk menghasilkan rekomendasi yang akurat.</li></ul>	215-228
3. Model Rekomendasi:	229
<ul style="list-style-type: none"><li>Model rekomendasi dibangun menggunakan <i>Non-negative Matrix Factorization</i> (NMF), yang merupakan teknik collaborative filtering. NMF digunakan untuk mengidentifikasi pola laten dalam data interaksi pengguna.</li><li>Model ini diintegrasikan ke dalam backend untuk memproses data secara efisien dan memberikan rekomendasi yang dipersonalisasi kepada pengguna.</li><li>Proses rekomendasi melibatkan analisis data interaksi pengguna untuk mengidentifikasi preferensi dan memberikan saran produk yang relevan, meningkatkan pengalaman belanja pengguna.</li></ul>	230-237
Arsitektur ini dirancang untuk memastikan bahwa sistem dapat menangani permintaan pengguna dengan cepat dan memberikan pengalaman belanja yang dipersonalisasi dan memuaskan. Dengan memanfaatkan teknologi modern seperti React.js dan Flask, sistem ini mampu memberikan layanan yang responsif dan efisien.	238-241
4.2. System Workflow	242
5. Project Documentation	243
5.1. Implementation	244
5.2. Results and Discussion	245
Bagian ini membahas hasil dari implementasi model rekomendasi dan analisis terhadap kinerja model berdasarkan metrik evaluasi yang telah ditentukan, serta hasil tampilan website.	246-248

1. Hasil Evaluasi Model:

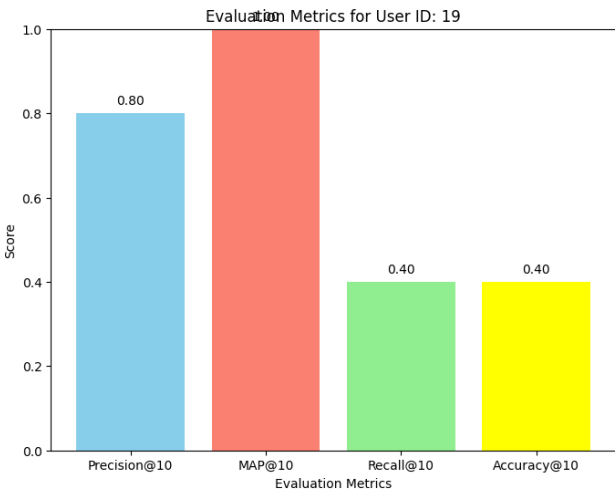
- Gambar Evaluasi Metrik:

– User ID 2:



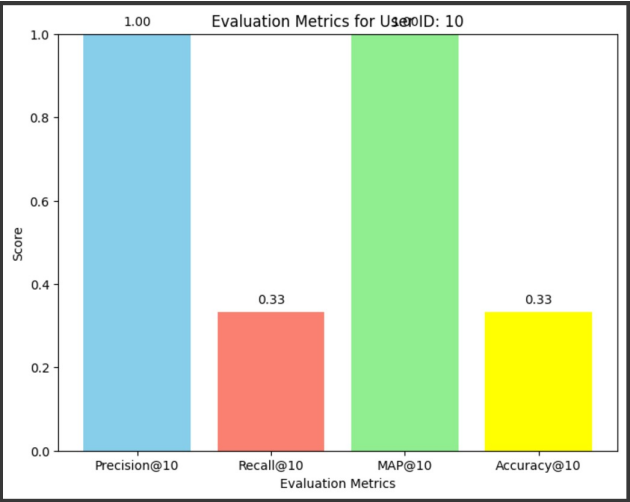
- \* Precision@10: 1.0
- \* Recall@10: 0.43
- \* MAP@10: 1.0
- \* Accuracy@10: 0.43

– User ID 19:



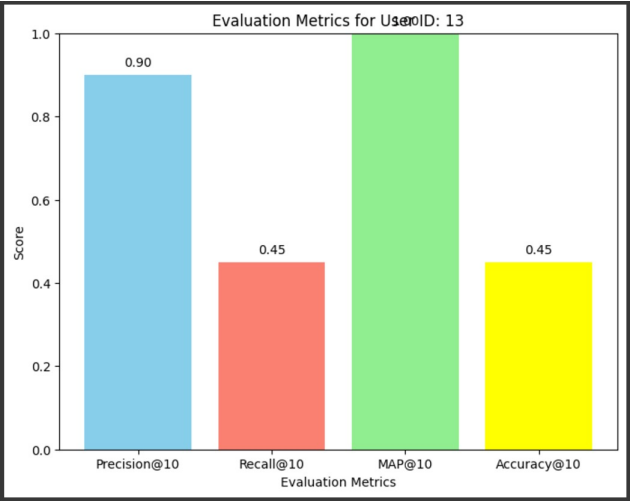
- \* Precision@10: 0.8
- \* Recall@10: 1.0
- \* MAP@10: 0.4
- \* Accuracy@10: 0.4

– User ID 10:



- \* Precision@10: 1.0
- \* Recall@10: 0.33
- \* MAP@10: 1.0
- \* Accuracy@10: 0.33

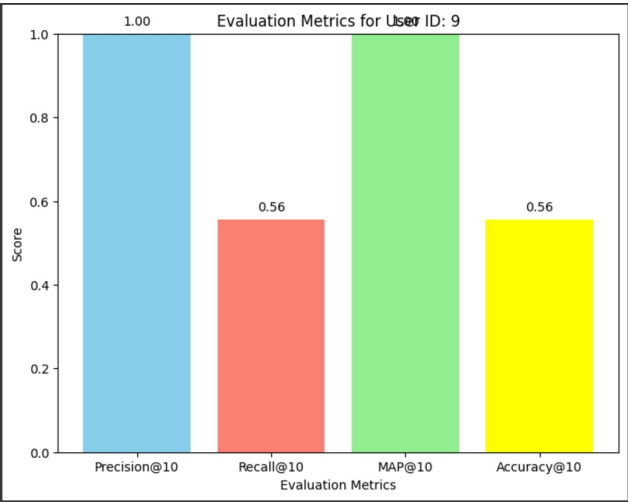
- User ID 13:



- \* Precision@10: 0.90
- \* Recall@10: 0.45
- \* MAP@10: 1.0
- \* Accuracy@10: 0.45

- User ID 9:





- \* Precision@10: 1.0
- \* Recall@10: 0.56
- \* MAP@10: 1.0
- \* Accuracy@10: 0.56

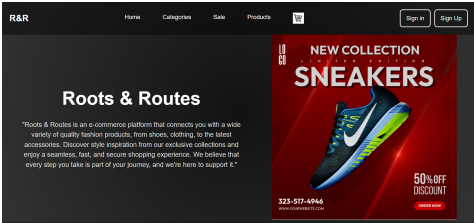
- **Simpulan:**
  - Setiap modul pengguna menunjukkan variasi dalam metrik evaluasi, yang mencerminkan perbedaan dalam pola interaksi dan preferensi pengguna.
  - Pengguna dengan riwayat interaksi yang lebih kaya cenderung memiliki nilai precision dan recall yang lebih tinggi, menunjukkan bahwa model dapat memberikan rekomendasi yang lebih akurat.
  - Variasi dalam MAP dan accuracy menunjukkan bahwa meskipun model memberikan rekomendasi yang relevan, ada ruang untuk peningkatan dalam menangkap semua item relevan.

2. Analisis Kinerja:

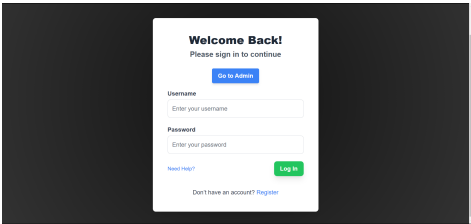
- Model berhasil memberikan rekomendasi yang dipersonalisasi, meningkatkan pengalaman pengguna di platform. Pengguna dengan riwayat interaksi yang lebih banyak mendapatkan rekomendasi yang lebih akurat.
- Untuk pengguna baru, model merekomendasikan produk berdasarkan popularitas, memastikan bahwa mereka tetap mendapatkan saran yang relevan meskipun tanpa riwayat interaksi.

3. Hasil Tampilan Website:

- **Antarmuka Pengguna:**
  - Antarmuka pengguna dibangun dengan *React.js*, menampilkan produk dan rekomendasi dengan responsif dan interaktif. Pengguna dapat dengan mudah menavigasi dan menemukan produk yang diinginkan.

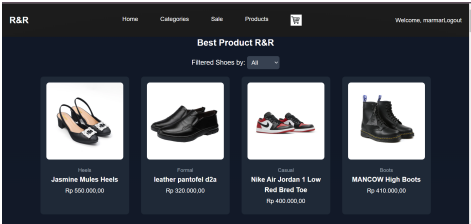


- Halaman home menyajikan berbagai kategori produk dan penawaran terbaru, memberikan pengguna akses cepat ke koleksi sepatu yang tersedia.
- **Proses Masuk (Sign In):**



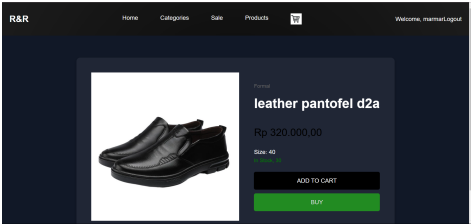
- Halaman sign in dirancang untuk memudahkan pengguna dalam mengakses akun mereka dengan cepat dan aman, menggunakan form yang sederhana dan intuitif.

- **Daftar Sepatu:**



- Halaman daftar sepatu menampilkan koleksi sepatu yang tersedia, dilengkapi dengan filter pencarian untuk memudahkan pengguna menemukan sepatu yang diinginkan.

- **Detail Sepatu:**



- Halaman detail sepatu memberikan informasi lengkap tentang produk, termasuk deskripsi, ukuran yang tersedia, dan harga, serta opsi untuk menambahkannya ke keranjang belanja dan opsi beli.

- **Desain UI:**

- Desain UI yang bersih dan intuitif memastikan bahwa pengguna dapat dengan mudah memahami dan menggunakan fitur rekomendasi. Elemen navigasi yang jelas dan tata letak yang terorganisir membantu pengguna dalam menjelajahi produk dan melakukan pembelian.

4. **Diskusi:**

- *Kekuatan Model:* Model NMF mampu menangkap pola laten dalam data interaksi, memberikan rekomendasi yang relevan dan bervariasi.
- *Keterbatasan Model:* Model mungkin kurang efektif untuk pengguna baru dengan sedikit atau tanpa riwayat interaksi. Solusi potensial termasuk menggabungkan data demografis atau menggunakan pendekatan hybrid.
- *Peluang Peningkatan:* Mengoptimalkan parameter model dan memperkaya dataset dengan lebih banyak fitur dapat meningkatkan akurasi dan relevansi rekomendasi.

Hasil ini menunjukkan bahwa model rekomendasi yang dikembangkan dapat meningkatkan pengalaman pengguna dan potensi konversi di platform e-commerce, meskipun masih ada ruang untuk perbaikan lebih lanjut.

5.3. *Conclusion*

6. **Appendices (if applicable)**