

# BAB I PENDAHULUAN

## 1.1 Latar Belakang

*E-commerce* saat ini menjadi salah satu sektor bisnis yang paling ramai pengguna dan berkembang dengan sangat pesat di seluruh dunia[1]. Penerapan teknologi pembelajaran mesin (*machine learning*) pada bisnis dapat meningkatkan penjualan dengan memberikan rekomendasi yang tepat ke pengguna[2]. Sistem rekomendasi dikembangkan agar sistem lebih mengerti kebutuhan pengguna. Pengembangan sistem rekomendasi yang telah dilakukan merupakan hasil *project capstone* pada program Studi Independen Merdeka Belajar – Kampus Merdeka (MBKM). Selanjutnya dilakukan penelitian dan pengembangan lebih lanjut guna meningkatkan efisiensi kepada pengguna *e-commerce* ketika pengguna melakukan pemilihan satu produk, tetapi tetap ingin mendapatkan beberapa pilihan variasi produk yang mirip. Sistem rekomendasi menjadi solusi agar pengguna mendapatkan produk yang sesuai dengan cepat.

Dalam penelitian ini terdapat beberapa batasan pada *project* implementasi sistem rekomendasi ke *website* yang dilakukan. Berikut batasan yang telah ditetapkan:

1. Dataset yang digunakan tidak memiliki data gambar, sehingga untuk membuat tampilan *website* agar tetap menarik dan terlihat seperti *website e-commerce* dengan menggunakan gambar, diambil beberapa sampel gambar saja.
2. Total dataset yang digunakan dalam model algoritma setelah dilakukan *preprocessing* yaitu 11.597 dan hasil rekomendasi atau *similar items* hanya menggunakan top-N 4 item.
3. Data gambar yang digunakan hasil dari *input* secara manual ke database. Sehingga, dapat ditampilkan pada *website* dengan menghubungkan database dan model algoritma.
4. Tidak memungkinkan melakukan *scraping* untuk data gambar karena dataset yang digunakan dari kaggle hasil *scraping* tahun 2021.

5. Tahap pengujian (*testing*) yang diimplementasikan pada *website* hanya 8 produk atau id produk.
6. Tahap pengujian (*testing*) menggunakan 8 data teratas

Tujuan dari pengembangan produk *website e-commerce* sistem rekomendasi guna untuk mempermudah pengguna dalam mendapatkan produk yang sesuai dengan target yang ingin dicari. Selain itu, sistem rekomendasi memberikan efisiensi dengan lebih cepat mendapatkan produk yang mirip tanpa melakukan pencarian ulang produk yang lain. Di sisi lain, penelitian ini bertujuan untuk mengembangkan ilmu pengetahuan pada bidang *data science*.

Penerapan model *machine learning* dalam penelitian ini secara teknis dapat digunakan atau diimplementasikan pemilik bisnis untuk memiliki sebuah sistem rekomendasi pada aplikasinya secara sederhana. Kepada peneliti selanjutnya, dapat menggunakan model dengan tingkat kompleksitas yang tinggi seperti *deep learning* agar mendapatkan rekomendasi yang *real-time*. Pada aplikasi *website e-commerce*, sistem rekomendasi ini berharap dapat memberi kepuasan kepada pengguna karena kemudahan dan efisiensinya.

## 1.2 Profil

BISA AI Academy (PT Bisa Artifisial Indonesia) merupakan startup digital berbasis *education technology (edutech)* yang fokus pada pembelajaran terkait kecerdasan artifisial (AI), ilmu data (*data science*) dan subset lainnya.

Dalam program studi independen BISA AI Academy pada *track AI - Hacker* memiliki target-target pembelajaran yang tercantum pada rincian kompetensi sebagai berikut:

1. Mahasiswa mampu menguasai Teknologi Kecerdasan Artifisial (*AI - Hacker*)
2. Mahasiswa memiliki dasar pengetahuan dari pilihan yang diambil pada posisi *AI - Hacker*, seperti *Data Science, Full Stack Programming, Machine Learning, dan Big Data*.
3. Mahasiswa mampu menghasilkan portofolio bidang AI-Hacker

- Mahasiswa mampu meningkatkan wawasan dan skill pada pembelajaran mandiri

Kegiatan pembelajaran dilakukan secara daring melalui *platform* tampil.id. Durasi program ini dilaksanakan selama kurang lebih 5 bulan dengan detail waktu 15 Agustus - 31 Desember 2022. Berikut adalah jadwal kegiatan pembelajaran terjadwal yang dilaksanakan selama program berlangsung:

Tabel 1. 1 Jadwal Kegiatan Studi Independen

No.	Kelas	Deskripsi Kelas	Pelaksanaan
1	Matematika dan Statistika	Belajar dasar Matematika dan Statistika untuk <i>Artificial Intelligence</i> .	Senin (08.00 – 10.00 WIB)
2	Full Stack Programming	Pembelajaran fokus pada pengembangan <i>website</i> .	Senin (10.00 – 12.00 WIB)
3	Data Science	Implementasi ilmu data dengan Bahasa pemrograman Python untuk solusi studi kasus.	Selasa (08.00 – 10.00 WIB)
4	Machine Learning	Belajar dasar beberapa model dan evaluasi dalam <i>machine learning</i> .	Selasa (13.00 – 15.00 WIB)
5	Big Data	Dasar pengelolaan data, analisis data, <i>data engineer</i> , dan implementasi data pada model <i>machine learning</i> .	Selasa (19.00 – 21.00 WIB)

No.	Kelas	Deskripsi Kelas	Pelaksanaan
6	Project Capstone AI - Hacker	Sebagai <i>project</i> akhir pada program studi independen dengan berbagai pilihan bidang.	Kamis (15.30 – 17.30 WIB)

Di luar dari jadwal di atas, terdapat pembelajaran mandiri yang dapat dikerjakan kapanpun pada *platform e-learning* BISA AI Academy dan pembelajaran tamu pada setiap hari Senin - Jumat dengan berbagai jadwal melalui TAMPIL ID untuk diikuti.

### 1.3 Landasan Teori

#### 1.3.1 Sistem Rekomendasi

Sistem rekomendasi merupakan sebuah metode yang dapat menerima suatu inputan dan menghasilkan suatu *output* berupa rekomendasi dari hasil inputan. Penerapan sistem rekomendasi dapat dijumpai pada sebuah aplikasi *e-commerce* yang mana dapat membantu pengguna dalam mendapatkan item yang menarik[3].

Metode yang dapat digunakan untuk sistem rekomendasi, seperti *collaborative-filtering*, *item-based filtering*, *knowledge-based*, dan *hybrid filtering*[4]. *Collaborative-filtering* dapat memberikan rekomendasi berdasarkan historis item pengguna, baik dalam bentuk ulasan (*feedback*) atau riwayat pencarian. *Item-based filtering* melihat kemiripan atribut atau beberapa atribut lalu dilakukan perbandingan antar item untuk dijadikan rekomendasi. *Knowledge-based* berdasarkan kondisi nilai atribut yang sudah ditentukan pengguna. Terakhir, *hybrid filtering* merupakan gabungan dari *item-based filtering* dan *collaborative-filtering*[5].

### 1.3.2 Cosine Similarity

*Cosine similarity* adalah salah satu teknik untuk mengukur atau menghitung nilai kemiripan antara 2 item[3]. Dalam pengertian lain, *cosine similarity* menghitung antara 2 vektor yang mana menghitung antara sudut *cosinus* di antara keduanya[5].

Kesamaan *cosinus* digunakan untuk melihat tingkat kemiripan dokumen dengan menerapkan konsep derajat *cosinus*. Batas nilai yang dimilikinya antara nilai 0 dan 1. Jika hasil *cosine similarity* 0 berarti tidak ada kemiripan dan jika hasil *cosine similarity* 1 maka tingkat kemiripannya sangat kuat atau serupa[6].

Rumus perhitungan *cosine similarity*:

$$\text{similarity}(x, y) = \frac{\sum x_i \cdot y_i}{\sqrt{\sum x_i^2} \cdot \sqrt{\sum y_i^2}}$$

Keterangan :

- $\sum x_i \cdot y_i$  = vektor (dot) produk dari vektor x dan y
- $\sqrt{\sum x_i^2} \cdot \sqrt{\sum y_i^2}$  = panjang produk antara vektor x dan y

### 1.3.3 Item-Based Filtering

Pendekatan *item-based filtering* salah satu metode untuk sistem rekomendasi. *Item-based filtering* mengukur kemiripan item berdasarkan karakteristik item-itemnya. Metode ini dengan mengambil atribut item yang bisa diukur atau dihitung kesamaannya. Semua item akan dihitung dan diukur seberapa mirip item dengan item target. Sejumlah  $k$  item dapat ditentukan sebagai batasan jumlah item yang paling mirip[7].

#### 1.3.3.1 Item Profile

Metode *item-based filtering* dengan memberikan rekomendasi berdasarkan kemiripan *item profile*. *Item profile*

merupakan representasi data yang mengacu pada suatu *item* atau konten. Istilah tersebut digunakan untuk menjelaskan karakteristik atau atribut yang mendeskripsikan *item*. Karakteristik tersebut bisa dari satu atribut atau gabungan dari atribut lainnya[8].

### **1.3.4 Text Preprocessing**

*Text preprocessing* merupakan tahapan proses awal untuk memproses data teks agar menjadi data teks yang dapat diolah selanjutnya. Beberapa tahapan *text preprocessing* sebagai berikut[9]

#### **1.3.4.1 Cleaning**

*Cleaning* merupakan proses membersihkan dari kata yang tidak diperlukan untuk mengurangi proses *noise* pada proses pembuatan model.

#### **1.3.4.2 Case Folding**

*Case folding* merupakan proses mengubah semua huruf menjadi huruf kecil. Dalam tahapan ini membantu menghindari perbedaan yang tidak perlu karena perbedaan huruf besar dan kecil saat melakukan *modeling*.

#### **1.3.4.3 Tokenizing**

*Tokenizing* merupakan proses pemisahan atau memecah teks menjadi bagian-bagian kata.

#### **1.3.4.4 Filtering atau Stopword**

*Filtering* merupakan proses mengabaikan kata-kata umum yang tidak memiliki makna khusus.

### 1.3.5 Bahasa Pemrograman

Bahasa pemrograman yang digunakan dalam pengerjaan penelitian dan *project* ini adalah Python versi 3.11.1. Penggunaan bahasa pemrograman ini digunakan dalam proses pembangunan model algoritma dan pengembangan *website* sebagai aplikasi *deployment* model.

Aplikasi *website* yang dikembangkan didukung dengan Flask yang mana *framework* bahasa pemrograman Python. Pengembangan *website* dengan Flask dibuat pada sebuah *virtual environment* untuk dijalkannya.

Data gambar yang tampil pada sebuah *website* diakses dari sebuah *database* dengan bahasa pemrograman SQL. SQL termasuk bahasa *query* untuk mengakses data dalam basis data relasional[10].

#### 1.3.5.1 Framework Flask

Flask merupakan sebuah *framework* bahasa pemrograman Python yang bertujuan untuk membantu pengembang membangun sebuah web terstruktur dan dapat diatur dengan mudah[11].

Flask termasuk jenis *microframework*, penggunaan dalam membangun aplikasi *website* yang ringan, fleksibel, dan berkinerja tinggi. Sebagian besar fungsi dan komponen umum seperti validasi form, *database*, dan sebagainya tidak terpasang secara *default* di Flask. Fungsi dan komponen-komponen tersebut sudah disediakan pihak ketiga dan Flask dapat menggunakannya seakan diimplementasikan oleh Flask sendiri[12].

#### 1.3.5.2 Structured Query Language (SQL)

*Structured Query Language* (SQL) adalah suatu Bahasa pemrograman untuk menjalankan serta membuat perintah untuk manajemen data pada *Relational Database Management System* (RDBMS)[12]. Perintah yang digunakan SQL untuk berkomunikasi dengan *database* disebut *query*. Dengan menggunakan *query* dapat

melakukan membuat, edit, tambah, dan hapus data yang ada pada *database*.

### 1.3.6 Precision

Precision adalah salah satu metrik evaluasi yang bertujuan untuk mengukur performa model algoritma. Precision akan membandingkan prediksi benar positif dengan keseluruhan hasil prediksi positif. Precision memberikan informasi tentang seberapa sering model algoritma membuat prediksi positif yang benar[13].

Secara matematika, untuk menghitung *precision* bisa menggunakan rumus berikut:

$$\text{Precision} = \frac{\text{jumlah prediksi benar (positif)}}{\text{prediksi benar (positif)} + \text{prediksi salah (positif)}}$$

### 1.3.7 Prototype

*Prototype* merupakan proses yang ditujukan untuk menjadikan berbagai sifat abstrak dari sebuah ide menjadi lebih berwujud atau terlihat menyerupai hasil sebenarnya. Dalam pengembangan produk, hal ini sangat penting untuk memastikan bahwa aplikasi tersebut bisa berfungsi dengan baik dan memenuhi kebutuhan pengguna[14].

Hal ini memiliki peran yang penting dalam mempercepat waktu pengembangan produk dan memastikan bahwa produk yang dihasilkan sesuai dengan kebutuhan dan keinginan pengguna. Selain itu, dapat meminimalisir kesalahan dan risiko lainnya dalam pengembangan produk[15].

### 1.3.8 Black Box Testing

Pengujian *black-box* (juga dikenal sebagai pengujian berbasis data atau pengujian berbasis input/output) adalah metode pengujian untuk



mengamati hasil *input* dan *output* dari perangkat lunak tanpa mengetahui struktur kode dari perangkat lunak. Metode ini seperti kotak hitam dengan tidak peduli pada programnya. Namun, pengujian untuk menemukan keadaan di mana program tidak berperilaku sesuai spesifikasinya[16].

