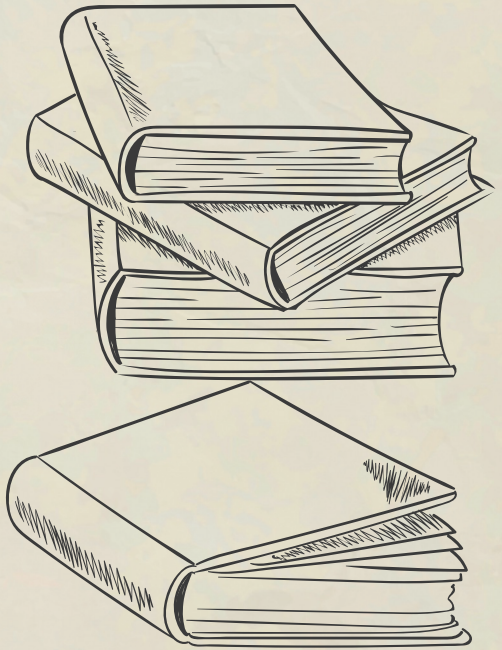
MACHINE LEARNING:

Book Recommendation System

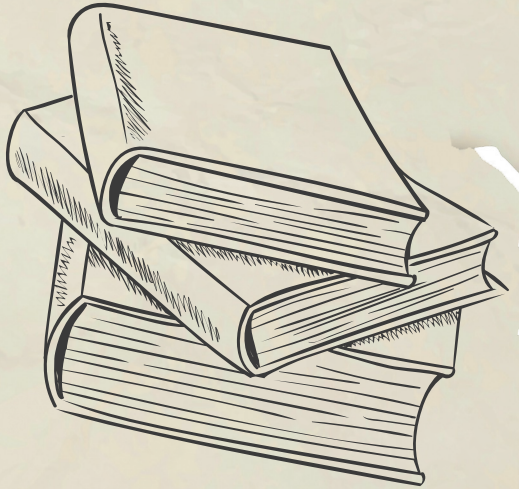


Rosemary Medina-Casanova

Table of Contents

Business Case

Goodreads hired me to create a book recommendation system, using data from an API, to provide their customers with a more personalized experience.



Contents of the Data

This project implements a book recommendation system using collaborative filtering with the Surprise library. The project includes feature engineering, model training, evaluation, and visualizations.

Requirements	Python, pandas, scikit-learn, surprise, matplotlib, seaborn, streamlit
Data Used	Two datasets were collected from Goodreads (API – scraped)
Data Cleaning & Preprocessing	Drop columns & null values, defined chunk size, added genre column, converted data types for features
Machine Learning	Model Training (SVD), Model Evaluation (MAE, R2, RMSE), Model Selection (Surprise, Train/Test Split), Collaborative Filtering
Feature Engineering	Avg Rating, MinMaxScaler

Feature Engineering

These are some of the steps I took in preparing the data for Model Selection & Evaluation

Data Type Conversion Ensured that data was in a format that could be easily processed by the recommendation algorithm, improving the efficiency of the model training process.
Feature Normalization The numerical features, such as ratings, were normalized to ensure that all features contribute equally to the model's predictions.
Sparse Matrix Conversion The user-item matrix was inherently sparse due to many users not rating many books. The Surprise library handles this by converting the user-item matrix into a sparse matrix representation, which is more efficient for handling large datasets with many missing values.

SVD & Collaborative Filtering

SVD was used to perform collaborative filtering by decomposing the user-item interaction matrix.

Collaborative Filtering

Collaborative filtering is a method used to make automatic predictions about the interests of a user by collecting preferences from many users.

Example: If person A has the same opinion as person B on an issue, A is more likely to have B's opinion on a different issue than that of a randomly chosen person.

Singular Value Decomposition (SVD)

SVD is a matrix factorization technique used in collaborative filtering. It decomposes the user-item interaction matrix into three matrices, capturing the underlying structure of the data. This technique helps in reducing the dimensionality of the data and finding latent factors that explain observed interactions.

The basic idea is to represent both users and items in a lower-dimensional space of latent factors, which can be used to predict the missing entries in the user-item matrix.

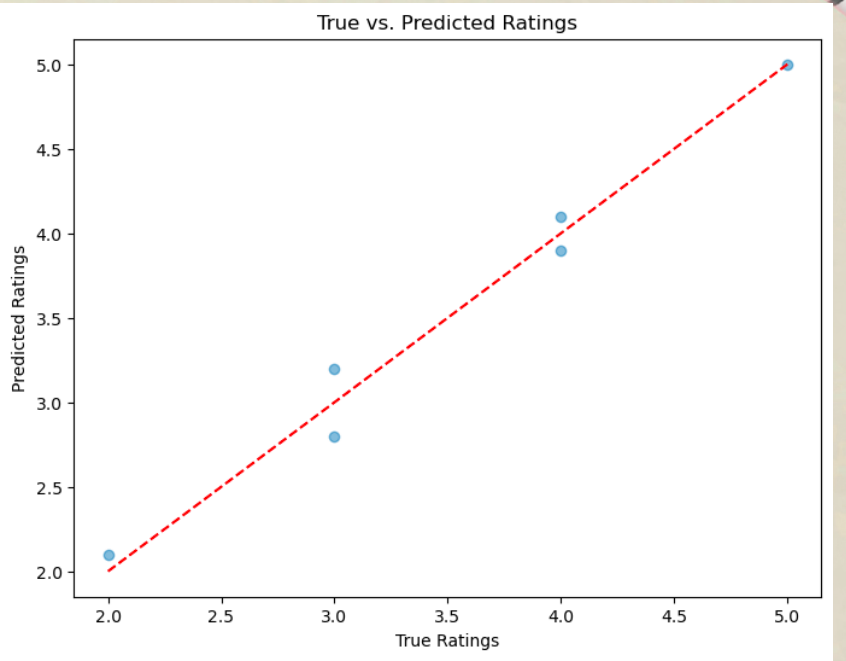
Model Selection

Algorithm Choice (SVD):
Selected for its ability to handle sparse matrices and uncover latent factors.

Data Preparation:
Prepared data using the Surprise library's Reader and Dataset.

Model Training:
Leveraging the surprise library to implement SVD; trained the SVD model on the user-item interaction matrix.

Prediction and Evaluation:
Generated predictions to evaluate model performance.



Model Evaluation

0.6594712259028492

MEAN ABSOLUTE ERROR (MAE)

0.2043545102836752 4

R-SQUARED (R2)

0.8436022074893497

ROOT MEAN SQUARED ERROR (RMSE)

Streamlit Integration Recommendation System

User Input

The system takes a User’s ID as input, representing their unique identity within the dataset

Model Deployment

The trained recommendation model was deployed using Streamlit, enabling users to access and interact with the system directly.

Output

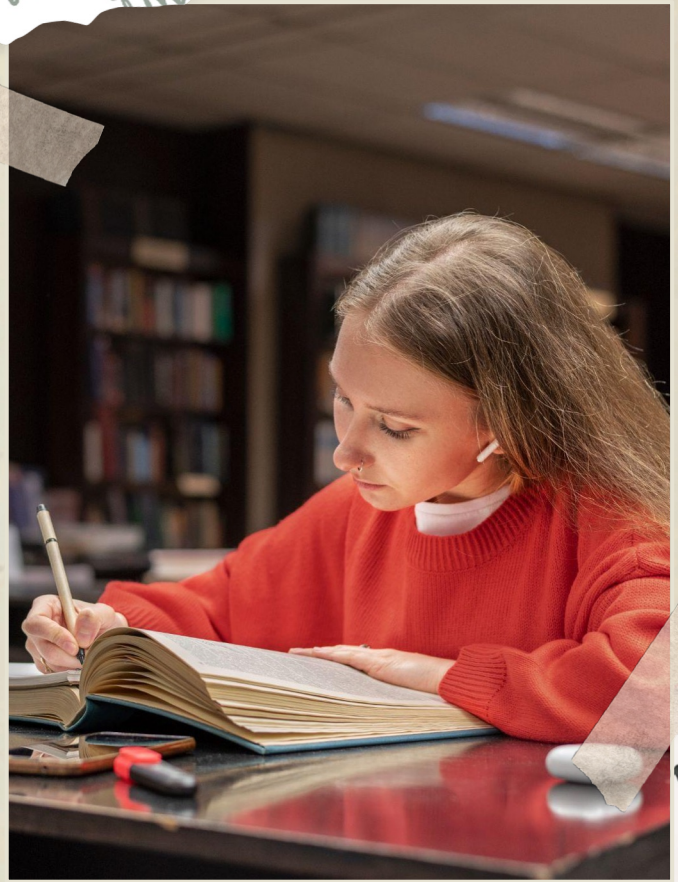
The output lists book recommendations (1-20), including title, author, and average rating, tailored to the User's profile.

Integrating Streamlit with the recommendation system allowed me to create an interactive web application where users can input their preferences and receive real-time book recommendations.

Insights/Improvements

Insights: The book recommendation system creates a more personalized experience, leading to boosted sales, increased engagement, and enhanced user experiences.

Improvements: Collect accurate genre data to improve the accuracy and relevancy of the recommendations.



THANK YOU!

daddy, chill