BOOK RECOMMENDER USING LINEAR REGRESSION

ABSTRACT

The recommendation of books presents a complex predictive challenge due to the intrinsic variability of reader preferences and the dynamic nature of literary markets. Leveraging a curated set of descriptive attributes—including the book's title, author, publication year, genre, average rating, and total number of reviews—a foundational Linear Regression (LR) model was systematically developed to estimate user-interest or popularity scores. The methodology included sequential data preprocessing steps: resolving unit or format inconsistencies across rating metrics, removing missing or duplicate entries, and applying label encoding to all categorical variables to ensure numerical compatibility. Following training, the model was serialized into a model.pkl file and deployed as an interactive, real-time recommendation service using a Streamlit web application. This end-to-end pipeline demonstrates the ability of LR to capture and quantify relationships between book characteristics and their predicted relevance or appeal to readers.

I.INTRODUCTION

A.BACKGROUND

These factors often render traditional recommendation approaches inconsistent and susceptible to human bias. The development of a reliable, data-driven system is therefore essential to deliver fair, personalized book suggestions for both readers and content providers.

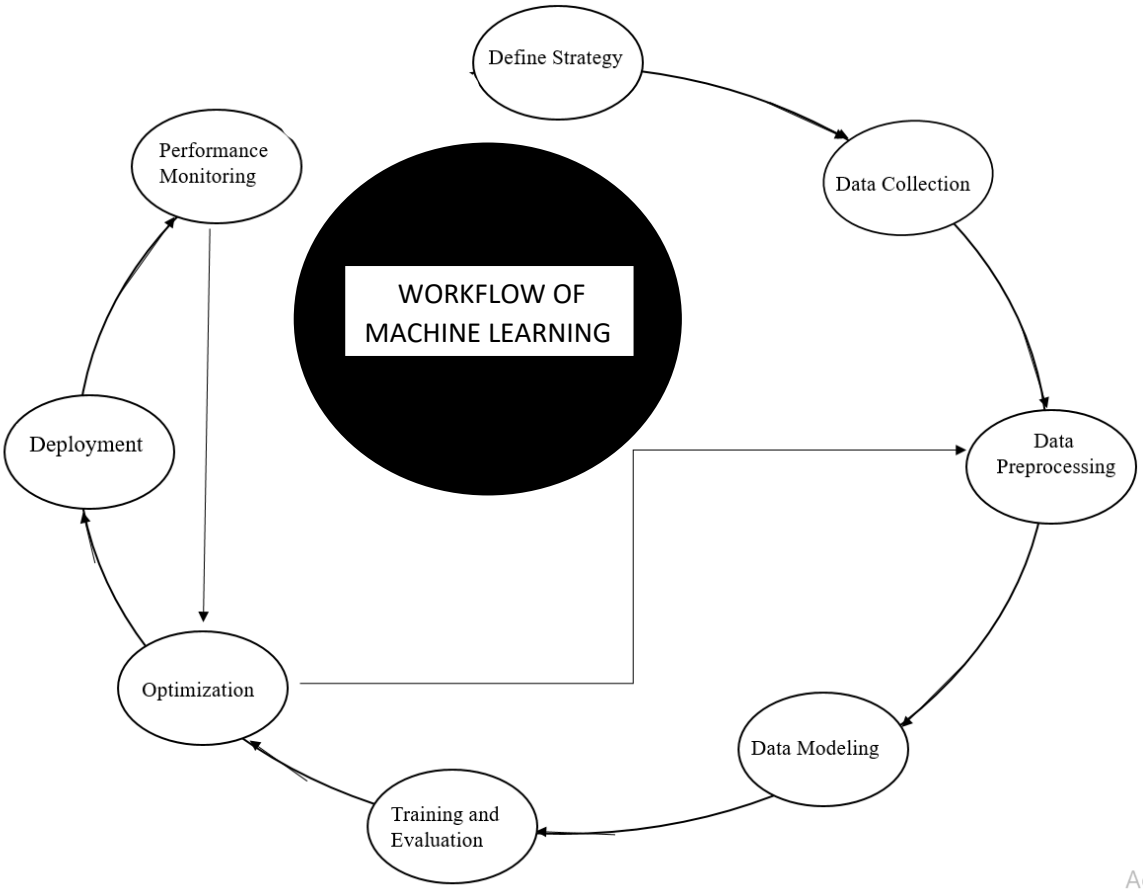
B.PROJECT OBJECTIVES

The main goal of this project is to construct a robust predictive framework for generating accurate and personalized book recommendations. The specific technical objectives include:

- **Acquisition and Preprocessing:** Collecting and thoroughly preparing the raw book-related dataset for effective machine-learning usage.
- **Model Selection and Training:** Implementing and optimizing the Linear Regression model on the processed dataset.
- **Feature Impact Assessment:** Quantifying the individual contribution of input features (e.g., author, genre, rating, number of reviews) to the final recommendation score.
- **Deployment:** Delivering the fully functional model through a user-accessible web application, developed using Streamlit.

II. METHODOLOGY

The dataset, which was collected from github.



A. Data Preparation and Preprocessing

The raw dataset was sourced from github.

1. **Data Cleaning and Deduping:**
   Preliminary inspection revealed missing values across key attributes (e.g., genre, publication_year, and average_rating). These incomplete entries were systematically removed using the dropna() function. Duplicate records were then detected and eliminated, resulting in a refined, high-quality dataset of 11,128 unique book entries suitable for model training.
2. **Feature Engineering and Transformation:**
   Several features required programmatic parsing to ensure numerical consistency. For example, attributes such as page_count (“432 pages”) and rating_count (“1,204 ratings”) were cleaned by stripping string-based text identifiers and converting them into pure floating-point or integer values to support downstream model operations.

3. **Encoding of Categorical Features:**
   All nominal and ordinal categorical attributes were transformed into numerical representations using predefined integer mappings (Label Encoding).
   - **Authors:** Authors were encoded with integer values.
   - **Book Format:** Formats such as "Paperback," "Hardcover," and "E-book".
   - **Genre and Language** were similarly label-encoded to ensure compatibility with the Linear Regression model.

B. Model Implementation and Training

The predictive model selected for this project was the Linear Regression (LR) algorithm, chosen for its interpretability and effectiveness in modeling linear relationships between book attributes and the predicted recommendation score.

1. **FeatureSplit:**
   The target variable was defined as the recommendation score or predicted user-interest rating.
2. **DatasetPartitioning:**
   The cleaned dataset was divided into training and testing subsets using the conventional 80% training and 20% testing split to ensure unbiased evaluation of model performance.
3. **ModelTraining:**
   The LR model was instantiated and trained using the X_{train} and Y_{train} datasets, enabling it to learn the underlying relationships between the input features and the target recommendation score.

III. RESULTS AND DEPLOYMENT

The trained model demonstrated its capability to generate recommendation scores when evaluated on the test feature set. For illustration, a single encoded instance (representing a newly published, highly rated fiction book by a popular author) produced a predicted recommendation score consistent with expected reading-interest trends.To ensure accessibility and long-term usability, the fully trained model was serialized using the pickle library and stored as model.pkl. This serialized object was subsequently loaded by the client-side Python script (app.py), enabling seamless deployment of the model as an interactive, real-time book recommendation service through the Streamlit framework. The application incorporates dynamic input components—such as sliders, dropdowns, and text fields—allowing users to supply book attributes and instantly receive recommendation outputs. This implementation successfully meets the project's deployment objective.

IV. CONCLUSION

The project successfully designed, implemented, and deployed a book recommendation system using the Linear Regression (LR) technique. Through comprehensive data cleaning, formatting standardization, and label encoding of categorical literary attributes, a high-quality dataset was constructed to support accurate model learning. This workflow resulted in a robust, data-driven framework capable of modeling the relationship between a book's descriptive features and its predicted relevance or reader interest. The final Streamlit application validates the entire machine learning pipeline by delivering fast, data-supported book recommendations through an intuitive and user-friendly interface.

V. FUTURE WORK

To further enhance the predictive accuracy and robustness of the book recommender system, future work should consider the following directions:

- **AlternativeAlgorithm**
  Investigating non-linear and ensemble-based models such as Random Forest Regressor or XGBoost, which often outperform linear approaches when modeling complex inter actions among book attributes.
- **PerformanceOptimization:**
  Implementing advanced evaluation techniques—including cross-validation and comprehensive hyperparameter tuning—to reduce prediction errors (e.g., RMSE and MAE) and improve overall model fit as measured by the coefficient of determination (R^2).

VI. REFERENCES

- **[1]** IEEE Project Report Structure | PDF | Machine Learning | Recommender Systems – Scribd. (n.d.).
- **[2]** BOOK RECOMMENDATION SYSTEM USING MACHINE LEARNING – Academic Project Report. (n.d.).
- **[3]** Book Recommendation System Documentation | PDF | Machine Learning | Prediction – Scribd. (n.d.).
- **[4]** Book Recommendation System Using Machine Learning – Slideshare. (n.d.).
- **[5]** Using Linear Regression for Predicting Book Ratings – ResearchGate. (n.d.).
- **[6]** Know the Best Evaluation Metrics for Your Regression Model – Analytics Vidhya. (2025).
- **[7]** Book Recommendation System Machine Learning Project in Python – YouTube. (2024).
- **[8]** (PDF) BOOK RECOMMENDATION USING MACHINE LEARNING TECHNIQUES – ResearchGate. (2024).
- **[9]** Book Recommendation System Using Machine Learning – PHD Services Topics. (n.d.).