A
MACHINE LEARNING PROJECT REPORT
ON
BOOK RECOMMENDED SYSTEM

In fulfillment of

B.Tech 3rd yr (Computer Science & Engg.)

Submitted to :

UMESH AGARWAL
ASSISTANCE PROFESSOR

Submitted by :

ABHISHEK MALAV
22EJICS010

Acknowledgment

I would like to express my gratitude to **Mr. Umesh Agarwal** for providing me with the opportunity to work on this project and for their continuous support and guidance. I would also like to thank **Ms. Mamta Garg**, Head of the Computer Science and Engineering Department at the Jodhpur Institute of Engineering and Technology, for their encouragement and moral support. Additionally, I am grateful to all the faculty members of the department, my parents, and God for their consistent blessings.

Table of Contents

1. Introduction

The aim of this project is to build a **Book Recommendation System** using machine learning techniques. Recommendation systems are widely used in various industries to predict the preferences of users and suggest relevant items such as books, movies, and products. Our system provides book recommendations by analyzing user-book interactions through ratings.

The main features include:

- Embedding books and users into a lower-dimensional space.
- Predicting user ratings for books based on past data.
- Providing book recommendations for specific users.

2. Technology Used

- **Programming Languages**: Python
- **Libraries**:
  - **Numpy**: For numerical operations. ○ **Pandas**: For data manipulation. ○ **Matplotlib & Seaborn**: For data visualization. ○ **TensorFlow & Keras**: For building and training deep learning models.
  - **Scikit-learn**: For data splitting and dimensionality reduction.
- **Tools**: VS Code

3. Project Details

3.1 Flow Chart

Start

|

Load Data (User-Book Ratings)

|

Split Data into Train and Test Sets

|

Create Embeddings for Users and Books

|

Train Model using Training Data

|

Evaluate Model on Test Data

|

Generate Book Recommendations for User

|

End

3.2 Functions/Modules

- **Data Loading and Preprocessing**: Load and preprocess the book ratings dataset using Pandas.
- **Model Architecture**:
    - User and book embeddings using TensorFlow/Keras. ○ A simple dot product-based model for rating prediction.
    - An advanced model with additional dense layers for improved performance.
- **Training and Evaluation**: Model training with mean squared error loss function and evaluation on the test set.
- **Dimensionality Reduction**: Use PCA and t-SNE for visualizing book embeddings.

- **Recommendation Generation**: Predict ratings for unseen books and recommend top books.

3.3 Project Code

The core code for the project includes the following steps:

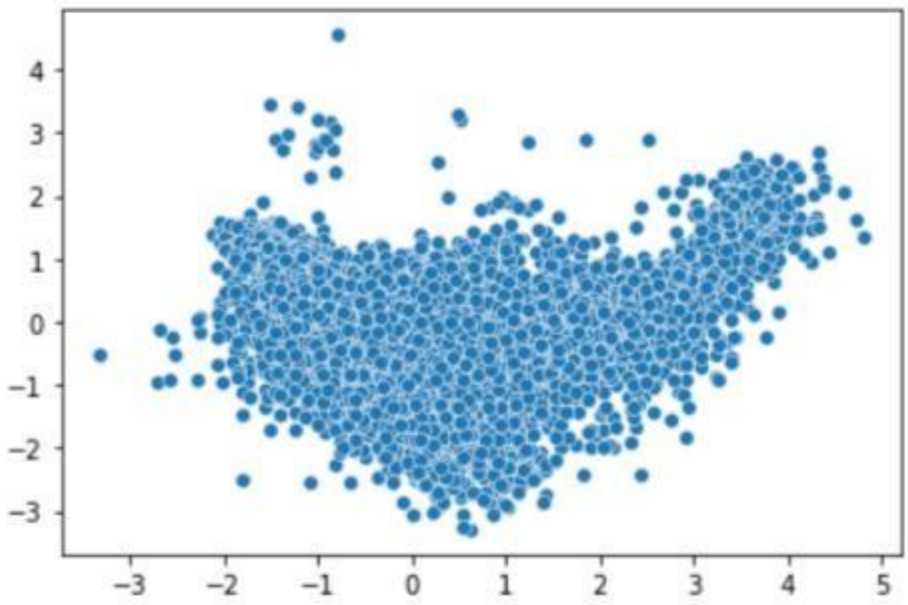
```
import numpy as np import
pandas as pd import
matplotlib.pyplot as plt import
os import warnings
from tensorflow.keras.layers import Input, Embedding, Flatten, Dot, Dense,
Concatenate
from tensorflow.keras.models import Model from
sklearn.model_selection import train_test_split
from keras.models import load_model from
sklearn.decomposition import PCA from
sklearn.manifold import TSNE import
seaborn as sns

# Ignore warnings
warnings.filterwarnings('ignore')

# Load dataset
dataset = pd.read_csv('ML_Project/ratings.csv')

# Split the dataset into training and testing sets
train, test = train_test_split(dataset, test_size=0.2, random_state=42)

# Get the number of unique users and books n_users
= len(dataset.user_id.unique())
n_books = len(dataset.book_id.unique())

# Create book and user embeddings
book_input = Input(shape=[1], name="Book-Input")
```

```
book_embedding = Embedding(n_books + 1, 5,
name="BookEmbedding")(book_input)
book_vec = Flatten(name="Flatten-Books")(book_embedding)

user_input = Input(shape=[1], name="User-Input") user_embedding
= Embedding(n_users + 1, 5, name="UserEmbedding")(user_input)
user_vec = Flatten(name="Flatten-Users")(user_embedding)

# Dot product for rating prediction
prod = Dot(name="Dot-Product", axes=1)([book_vec, user_vec])

# Compile and fit the model
model = Model([user_input, book_input], prod)
model.compile(optimizer='adam', loss='mean_squared_error')

# Train and evaluate the model
history = model.fit([train.user_id, train.book_id], train.rating, epochs=5,
verbose=1)
model.evaluate([test.user_id, test.book_id], test.rating)

# PCA and t-SNE for visualization pca
= PCA(n_components=2)
pca_result = pca.fit_transform(model.get_layer('Book-
Embedding').get_weights()[0])
```

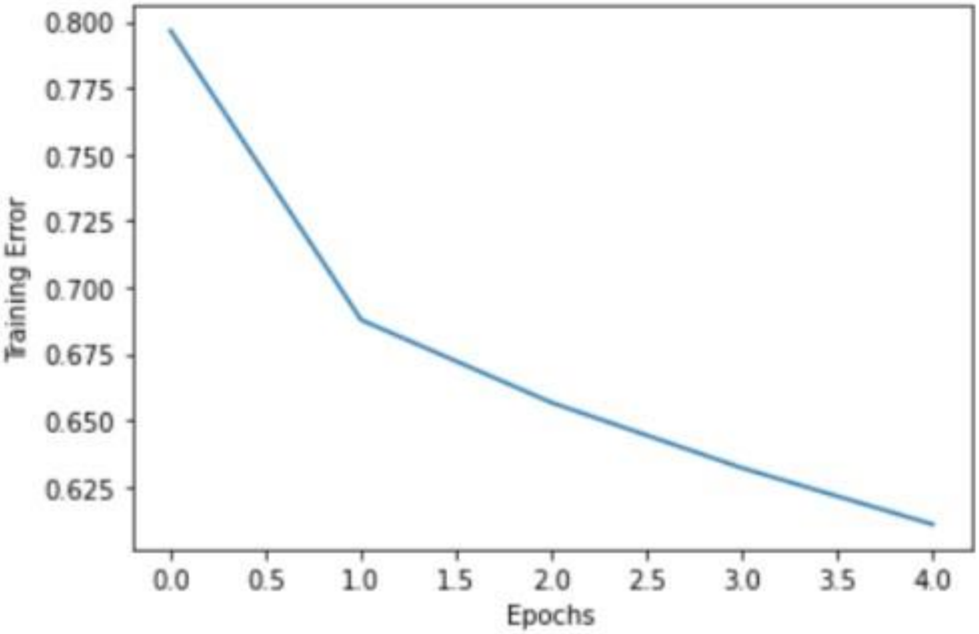
3.4 Project Screenshots

Include screenshots of the following:

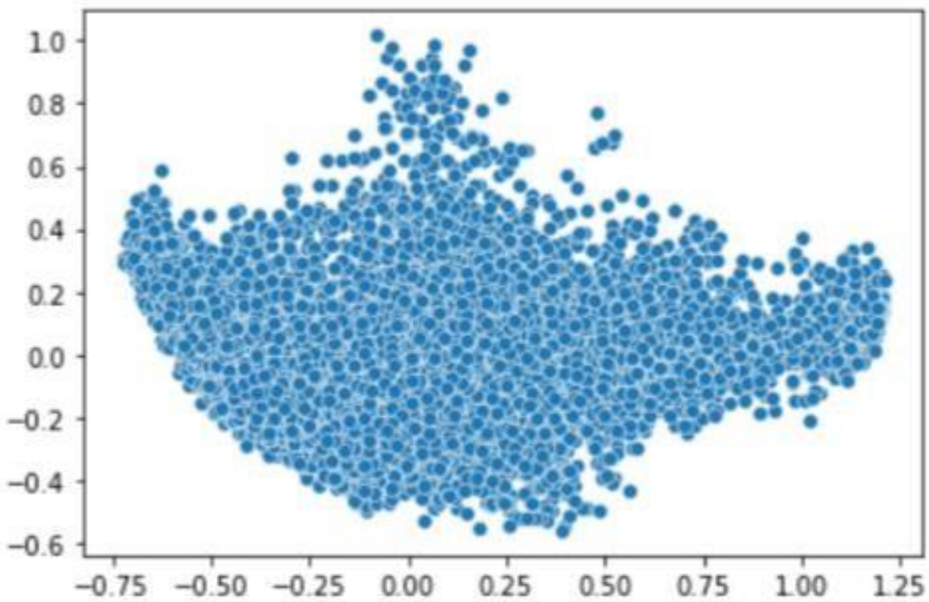
1. Dataset head and shape



2. Loss curve during training



3. PCA and t-SNE scatter plots for book embeddings


1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00
1.25

4. Applications

The Book Recommendation System can be used in:

- **Online Bookstores** to provide personalized book recommendations.
- **Libraries** to suggest books to readers based on their borrowing history.
- **E-learning Platforms** to recommend course materials or textbooks.

5. Conclusion and Future Work

This project successfully demonstrates how machine learning models can be used to build a book recommendation system. The basic dot product model and the advanced model with dense layers both provide meaningful predictions. Future improvements can be made by incorporating additional features such as book metadata, user demographics, and using more advanced architectures like neural collaborative filtering.

6. References

- **Pandas Documentation**: https://pandas.pydata.org/
- **TensorFlow/Keras Documentation**: https://www.tensorflow.org/
- **Scikit-learn Documentation**: https://scikit-learn.org/
- **GitHub link:** GitHub Link