RECOMMENDER SYSTEM FOR ONLINE COURSES

A PROJECT REPORT

Submitted by

RANJANI V (2303811714822037)

in partial fulfilment of requirements for the award of the course

AGB1102 – MACHINE LEARNING TECHNIQUES

in

ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

K. RAMAKRISHNAN COLLEGE OF TECHNOLOGY

(An Autonomous Institution, affiliated to Anna University Chennai and Approved by AICTE, New Delhi)

SAMAYAPURAM – 621 112

DECEMBER - 2024

K. RAMAKRISHNAN COLLEGE OF TECHNOLOGY

(AUTONOMOUS)

SAMAYAPURAM – 621 112

BONAFIDE CERTIFICATE

Certified that this project report titled **"RECOMMENDER SYSTEM FOR ONLINE COURSES"** is the bonafide work of **Ranjani V (2303811714822037),** who carried out the project work under my supervision. Certified further, that to the best of my knowledge the work reported here in does not form part of any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE

Dr. T. AVUDAIAPPAN M.E., Ph.D.,
HEAD OF THE DEPARTMENT
ASSOCIATE PROFESSOR
Department of Artificial Intelligence
K. Ramakrishnan College of Technology
(Autonomous)
Samayapuram–621112.

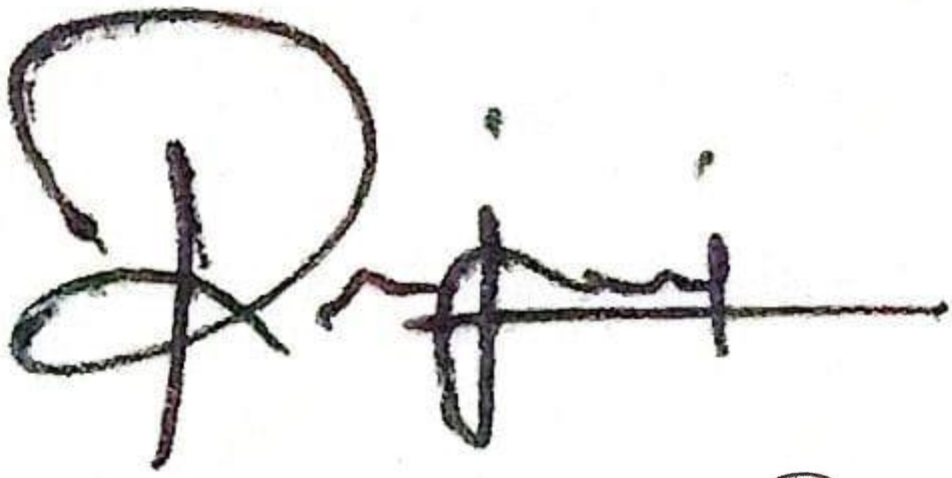
SIGNATURE

Mr. R. ROSHAN JOSHUA. M.E.,
SUPERVISOR
ASSISTANT PROFESSOR
Department of Artificial Intelligence
K. Ramakrishnan College of Technology (Autonomous)
Samayapuram–621112.

Submitted for the viva-voce examination held on 07.12.2024

INTERNAL EXAMINER

EXTERNAL EXAMINER



Signature

(RANJANI V)

Place: Samayapuram

Date:06/12/2024

ACKNOWLEDGEMENT

It is with great pride that I express our gratitude and indebtedness to our institution, "**K. Ramakrishnan College of Technology (Autonomous)**", for providing us with the opportunity to do this project.

I extend our sincere acknowledgment and appreciation to the esteemed and honourable Chairman, **Dr. K. RAMAKRISHNAN**, **B.E.,** for having provided the facilities during the course of our study in college.

I would like to express our sincere thanks to our beloved Executive Director, **Dr. S. KUPPUSAMY, MBA, Ph.D.,** for forwarding our project and offering an adequate duration to complete it.

I would like to thank **Dr. N. VASUDEVAN, M.TECH., Ph.D.,** Principal, who gave the opportunity to frame the project to full satisfaction.

I thank **Dr. T. AVUDAIAPPAN M.E., Ph.D.,** Head of the Department of **ARTIFICIAL INTELLIGENCE**, for providing her encouragement in pursuing this project.

I wish to convey our profound and heartfelt gratitude to our esteemed project guide **Mr.R.ROSHAN JOSHUA., M.E.,** Department of **ARTIFICIAL INTELLIGENCE,** for her incalculable suggestions, creativity, assistance and patience, which motivated us to carry out this project.

I render our sincere thanks to the Course Coordinator and other staff members for providing valuable information during the course.

I wish to express our special thanks to the officials and Lab Technicians of our departments who rendered their help during the period of the work progress.

VISION OF THE INSTITUTION

To emerge as a leader among the top institutions in the field of technical education.

MISSION OF THE INSTITUTION

- Produce smart technocrats with empirical knowledge who can surmount the global challenges.
- Create a diverse, fully-engaged, learner-centric campus environment to provide quality education to the students.
- Maintain mutually beneficial partnerships with our alumni, industry, and Professional associations.

VISION OF DEPARTMENT

To become a renowned hub for AIML technologies to producing highly talented globally recognizable technocrats to meet industrial needs and societal expectation.

MISSION OF DEPARTMENT

Mission 1: To impart advanced education in AI and Machine Learning, built upon a foundation in Computer Science and Engineering.

Mission 2: To foster Experiential learning equips students with engineering skills to tackle real-world problems.

Mission 3: To promote collaborative innovation in AI, machine learning, and related research and development with industries.

Mission 4: To provide an enjoyable environment for pursuing excellence while upholding strong personal and professional values and ethics.

PROGRAM EDUCATIONAL OBJECTIVES

Graduates will be able to:

1. PEO1: Excel in technical abilities to build intelligent systems in the fields of AI & ML in order to find new opportunities

2. PEO2: Embrace new technology to solve real-world problems, whether alone or as a team, while prioritizing ethics and societal benefits.

3. PEO3: Accept lifelong learning to expand future opportunities in research and product development.

PROGRAM SPECIFIC OUTCOMES (PSOs)

PSO 1: Domain Knowledge

To analyse, design and develop computing solutions by applying foundational concepts of Computer Science and Engineering.

PSO 2: Quality Software

To apply software engineering principles and practices for developing quality software for scientific and business applications.

PSO 3: Innovation Ideas

To adapt to emerging Information and Communication Technologies (ICT) to innovate ideas and solutions to existing/novel problems

PROGRAM OUTCOMES (POs)

Engineering students will be able to:

Engineering knowledge:

Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

Problem analysis:

Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences

Design/development of solutions:

Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations

Conduct investigations of complex problems:

Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions

Modern tool usage:

Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complete engineering activities with an understanding of the limitations.

The engineer and society:

Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice

Environment and sustainability:

Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development

Ethics:

Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

Individual and team work:

Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

Communication:

Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

Project management and finance:

Demonstrate knowledge and understanding of the engineering and management principles and apply these to one’s own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

Life-long learning:

Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

ABSTRACT

A recommender system is a subclass of information filtering systems designed to suggest items, services, or content to users based on their preferences, behaviors, or interactions. These systems leverage advanced machine learning techniques, data analysis, and algorithms to provide personalized recommendations, enhancing user experience and engagement. This study explores various approaches to building recommender systems, including collaborative filtering, content-based filtering, and hybrid methods. Furthermore, it discusses recent advancements such as deep learning-based recommendations, contextual modelling, and explainable recommendations to address limitations of traditional methods. The development and deployment of a recommender system are presented with a focus on scalability, accuracy, and user satisfaction, emphasizing its applications in e-commerce, entertainment, education, and other domains.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

ABBREVIATIONS

- PMF – Probabilistic Matrix Factorization
- MF-ALS – Matrix Factorization with Alternating Least Squares
- PCC – Pearson Correlation Coefficient
- RMSE – Root Mean Square Error
- MAE – Mean Absolute Error
- MSE – Mean Squared Error
- MAP – Mean Average Precision
- NDCG – Normalized Discounted Cumulative Gain

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION TO PROJECT

Recommender systems are specialized information filtering tools designed to predict and suggest items of interest to users based on their preferences, behaviour, or contextual data. They are a cornerstone of personalization, enhancing user experience by helping individuals navigate vast amounts of data efficiently. These systems are ubiquitous, powering recommendations in domains such as e-commerce, streaming platforms, online education, social media, and more.

1.2 PURPOSE AND IMPORTANCE OF THE PROJECT

1. **Personalization**: Tailor content, products, or services to match user preferences, enhancing engagement and satisfaction.
2. **Decision Support**: Assist users in discovering relevant items among a vast array of options, reducing decision fatigue and effort.
3. **Business Growth**: Increase sales, user retention, and customer loyalty by offering items that align with user interests.
4. **User Experience Optimization**: Foster a seamless and engaging experience by anticipating and fulfilling user needs.
5. **Revenue Maximization**: Drive monetization through targeted recommendations, cross-selling, and upselling.

- ✓ **Addressing Information Overload**: With the exponential growth of digital content and products, recommender systems filter out irrelevant options, making navigation more efficient.
- ✓ **Improving User Satisfaction**: Personalized recommendations lead to higher engagement and a sense of connection with the platform.

- ✓ **Boosting Business Metrics**:
  - **E-commerce**: Increase conversion rates and average order values.
  - **Media Streaming**: Enhance content consumption and reduce churn rates.
  - **Education Platforms**: Facilitate adaptive learning paths.
  - **Enabling Discovery**: Help users uncover new and niche items they might not have found otherwise, fostering diversity in choices.

1.3 OBJECTIVES

- ✓ Personalization
- ✓ User Experience Improvement
- ✓ Information Discovery
- ✓ Business Growth and Revenue Maximization
- ✓ Efficiency and Accuracy

1.4 PROJECT SUMMARIZATION

To design and implement a recommender system that provides personalized and relevant suggestions to users, enhancing user experience and supporting business goals like increased engagement, sales, and retention.

The project aims to create a recommendation engine tailored for a specific domain (e.g., e-commerce, media streaming, or education). It will leverage user behavior data, item features, and contextual information to deliver accurate and diverse recommendations.

User Personalization:

Recommendations will be customized based on individual user preferences, interaction history, and demographic data.

Recommendation Techniques:

- **Content-Based Filtering**: Suggest items based on user and item attributes.
- **Collaborative Filtering**: Leverage user or item similarity using interaction data.

- Hybrid Approach: Combine multiple techniques for improved performance.

Real-Time Recommendations:

Provide instant suggestions based on user activity and current context.

CHAPTER 2
PROJECT METHODOLOGY

2.1 INTRODUCTION TO SYSTEM ARCHITECTURE

The system architecture of a recommender system is a blueprint that outlines the structure and flow of data and processes required to generate personalized recommendations. It serves as a foundation for integrating various components and technologies that work together to deliver efficient and accurate suggestions.

2.1.1 High-Level System Architecture

The high-level system architecture for the recommender system application typically consists of several key components:

- ✓ Data Collection Layer
- ✓ Data Storage Layer

2.1.2 Components of the System Architecture

a. **User Data**

- **User Data** Definition: Information about users, including demographics, preferences, and interaction history.
- Examples:
    - Demographics: Age, gender, location.
    - Interaction Data: Clicks, purchases, ratings, time spent on an item.

b. Item Data

- Definition: Characteristics and metadata about the items being recommended.
- Examples:
    - E-commerce: Product descriptions, categories, price.
    - Media Streaming: Genre, cast, release year.

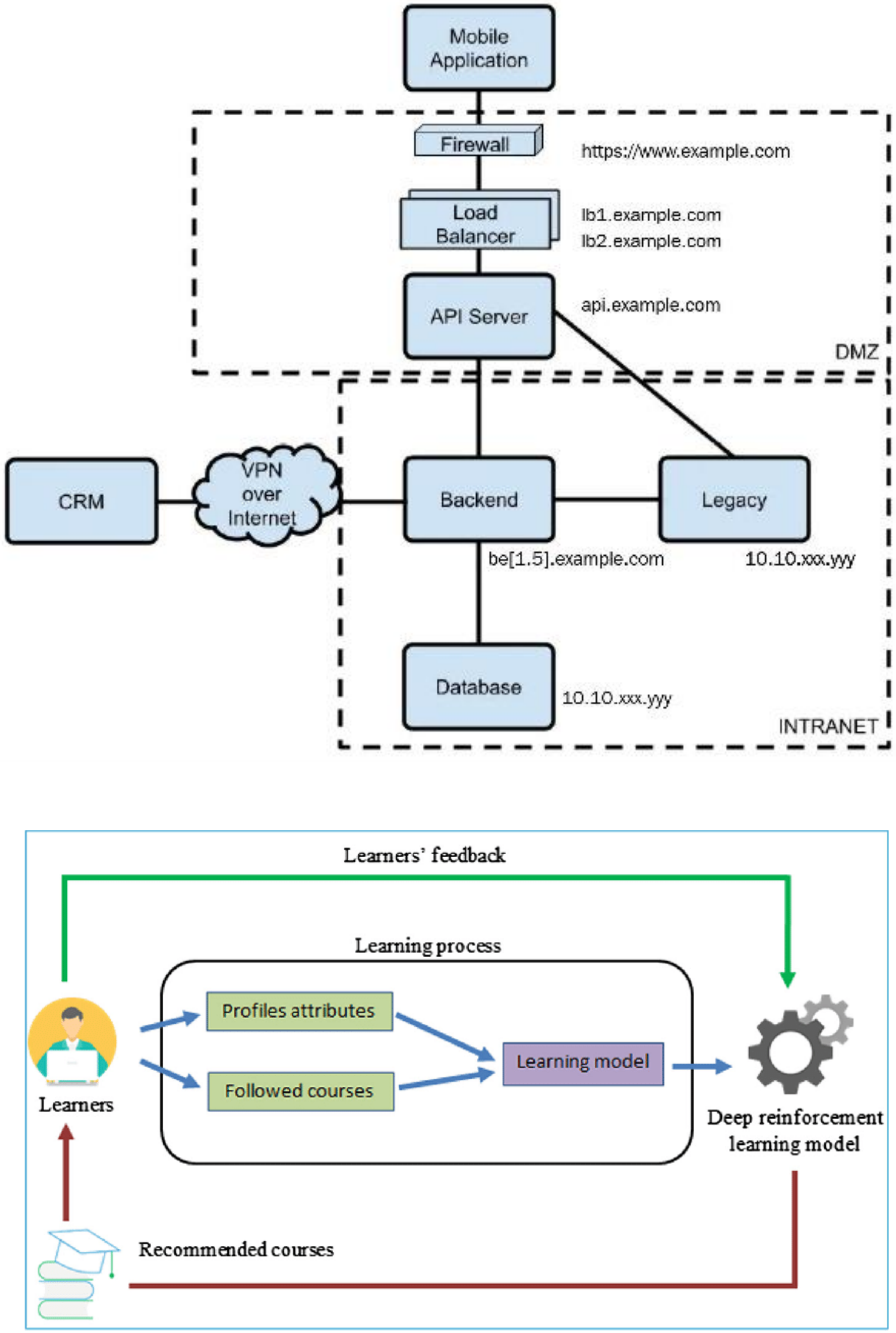


Fig 2.1 : Architecture Diagram (Sample)

CHAPTER 3

MACHINE LEARNING PREFERANCE

3.1 EXPLANATION OF WHY A RECOMMENDER SYSTEM WAS CHOSEN

A recommender system is chosen for a variety of reasons, primarily to enhance user experience, increase engagement, and improve business outcomes.

3.1.1 Personalized User Experience

Relevance of Content: Recommender systems help deliver content that is personalized to each user’s preferences and behavior, increasing the chances of user engagement. For example, movie recommendation platforms like Netflix or YouTube suggest videos based on users' viewing history, ensuring the content is relevant to their tastes.

3.1.2 Increase in User Engagement

Higher Engagement and Retention: By personalizing the user experience, recommender systems can increase user satisfaction, leading to higher retention rates and longer interaction times. For instance, e-commerce platforms use recommendation engines to suggest products, which may lead to increased purchases or clicks.

3.1.3 Handling Large Scale Data

Efficient Filtering: As the volume of data (e.g., products, users, items) increases, manual filtering or browsing becomes impractical. Recommender systems can process vast amounts of data and efficiently suggest items to users based on complex algorithms, such as collaborative filtering or content-based filtering.

3.1.4 Better Decision Making

Data-Driven Insights: Recommender systems provide valuable insights into customer preferences, behaviours, and trends. Businesses can leverage this data to make informed decisions about inventory, content strategy, pricing, or promotions.

3.1.5 Improving Customer Satisfaction and Loyalty

User-Cantered Design: Recommender systems are fundamentally designed to prioritize the user's needs, improving the overall satisfaction and the likelihood of users returning to the platform.

3.2 COMPARISON WITH OTHER ML

Recommender systems focus on predicting user preferences based on past interactions, such as ratings or clicks, using historical data. In contrast, supervised learning involves predicting an outcome (like class labels or continuous values) based on input features and requires labelled data for training. Recommender systems often work with sparse data where many interactions are missing, while supervised learning requires large, labelled datasets to be effective. Supervised models, such as decision trees or regression models, generally perform well on structured data, while recommender systems excel in scenarios where personalization is crucial, such as recommending products or movies.

3.3 ADVANTAGES AND DISADVANTAGES OF USING A ML

3.3.1 Advantages:

Personalized Learning Experience: Recommender systems tailor course suggestions based on individual preferences, learning history, and performance. This customization helps learners find relevant courses more easily, enhancing engagement and satisfaction.

Improved Course Discovery: With a wide range of online courses available, recommender systems help users discover courses they might not have found otherwise, encouraging continuous learning and exploration.

3.3.2 Disadvantages:

Bias and Lack of Diversity: Recommender systems often suggest courses based on past behaviours or preferences, which can lead to a narrow view and limited exposure to diverse topics. This can reinforce existing biases and prevent learners from exploring new or different subject areas.

Overfitting: If the system is too heavily based on a learner's past behaviour, it may lead to overfitting, where the recommendations are too similar to what the learner has already engaged with. This can stifle learning opportunities and personal growth by not presenting more challenging or unfamiliar content.

Data Privacy Concerns: Recommender systems rely on tracking user activity, such as course enrolment, ratings, and interactions. This raises privacy concerns, especially in the context of sensitive educational data.

CHAPTER -4

MACHINE LEARNING METHODOLOGY

4.1 Collaborative Filtering

- **User-based Collaborative Filtering**: Recommends items by finding similar users and suggesting what similar users liked.
  - ✓ Example: If User A likes items 1, 2, 3, and User B likes items 2, 3, 4, then item 4 may be recommended to User A.
- **Item-based Collaborative Filtering**: Focuses on the relationship between items. If an item is similar to items a user has liked, it will recommend those similar items.
  - ✓ Example: If a user liked item 1 and item 1 is similar to item 2, then item 2 is recommended.

4.1.1 Key features

1.Personalization

- User Profiling: The system tailors recommendations based on the preferences, behaviours, or interactions of individual users.
- Custom Recommendations: Each user receives unique recommendations that reflect their past behaviours (e.g., movie ratings, product purchases).

2. Context-Awareness

- Time-Based Recommendations: Some recommender systems adapt to the time of day, season, or other contextual factors (e.g., recommending warm clothes in winter).

- Location-Based Recommendations: Items can be suggested based on a user's geographical location, such as local restaurants or events.
- Device and Platform Adaptation: The system can provide recommendations tailored to the device the user is interacting with, such as mobile app recommendations vs. web browser recommendations.

4.2 **Content-Based Filtering**

- **Feature Extraction**: Extracts relevant features (e.g., genre, director, keywords) from the items.
- **User Profile Creation**: Builds a profile for the user based on their past preferences, which can be used to recommend similar items.
- **Similarity Measurement**: Uses methods like cosine similarity or Euclidean distance to compare items' feature vectors and recommend the most similar ones.

.

4.3. Hybrid Methods

1. **Weighted Hybrid**: Assigns a weight to each method and combines their results (e.g., 70% collaborative filtering, 30% content-based filtering).
2. **Switching Hybrid**: Switches between methods based on the context (e.g., using collaborative filtering when user-item interaction data is abundant, and content-based filtering when data is sparse).
3. **Mixed Hybrid**: Recommends items by using both methods in parallel, providing a diverse set of suggestions.

CHAPTER-5

MODULES

5.1 Introduction to Recommender Systems

5.1.1 Overview of Recommender Systems

- ✓ Definition and types (Collaborative Filtering, Content-Based Filtering, Hybrid Approaches)
- ✓ Applications in e-commerce, online courses, social media, etc.
- ✓ Item-based vs. user-based collaborative filtering
- ✓ Content-based filtering and its use cases
- ✓ Hybrid approaches combining both methods

5.2 Collaborative Filtering Techniques

5.2.1 User-Based Collaborative Filtering and Item-Based Collaborative Filtering

- Understanding the algorithm
- Nearest neighbor techniques
- Building a user-based recommender system
- Similarity measures: Cosine similarity, Pearson correlation
- Building an item-based recommender system

5.3 Content-Based Filtering and Hybrid Methods

5.3.1 Content-Based AND Hybrid Techniques

- ✓ Feature extraction (using text, images, etc.)
- ✓ TF-IDF, Bag-of-Words, Word2Vec for text-based recommendation
- ✓ Combining collaborative and content-based filtering
- ✓ Weighted hybrid, switching, and mixed approaches

CHAPTER 6
CONCLUSION & FUTURE SCOPE

6.1 CONCLUSION

In conclusion, recommender systems are a powerful tool for personalizing user experiences by analyzing past behaviors, preferences, and interactions. They play a crucial role in various domains such as e-commerce, entertainment, and social media, helping users discover relevant content or products. By employing techniques like collaborative filtering, content-based filtering, and hybrid approaches, recommender systems can significantly enhance user engagement and satisfaction. However, challenges such as data sparsity, scalability, and ensuring diversity in recommendations must be addressed for optimal performance. As technology continues to advance, the evolution of recommender systems promises even more precise, context-aware, and ethical recommendations.

6.2 FUTURE SCOPE

The future scope for recommender systems is vast and continually expanding as businesses and industries strive to provide more personalized and efficient experiences for users. With advancements in machine learning and artificial intelligence, recommender systems are expected to become increasingly sophisticated, offering hyper-personalized suggestions by analyzing not only user preferences but also contextual data such as location, time, and social influences. The integration of multimodal data, such as text, images, and videos, will enhance the ability to make recommendations across diverse platforms and applications.

In addition, the rise of deep learning and neural networks offers opportunities for more accurate predictions and complex recommendation algorithms. Furthermore, privacy and ethical considerations will drive the development of more transparent and privacy-preserving recommendation techniques, ensuring that users' data is protected while still delivering relevant and effective recommendations. With these advancements, recommender systems are set to play an even more integral role in fields like e-commerce, entertainment, healthcare, and education.

APPENDICES

APPENDIX A-SOURCE CODE

```
import pandas as pd
import numpy as np
from sklearn.neighbors import NearestNeighbors
from sklearn.preprocessing import LabelEncoder

# Step 1: Data Collection
# Sample data of User, Course, Rating
data = {
    'User': ['Alice', 'Bob', 'Charlie', 'David', 'Eve', 'Alice', 'Bob', 'Charlie', 'David', 'Eve'],
    'Course': ['Python Basics', 'Python Basics', 'Python Basics', 'Python Basics', 'Python
Basics',
               'Machine Learning', 'Machine Learning', 'Machine Learning', 'Machine
Learning', 'Machine Learning'],
    'Rating': [4, 5, 5, 3, 4, 3, 4, 5, 2, 4]
}

# Convert to DataFrame
df = pd.DataFrame(data)

# Step 2: Data Preprocessing
# Create a matrix where rows represent users and columns represent courses
user_course_matrix = df.pivot_table(index='User', columns='Course', values='Rating')

# Fill NaN values with 0 (since a missing rating indicates no interaction)
user_course_matrix = user_course_matrix.fillna(0)

# Step 3: Label Encoding for Users and Courses (if needed for scaling)
```

```
user_encoder = LabelEncoder()
course_encoder = LabelEncoder()

# Transform users and courses into numeric values
user_course_matrix_encoded = user_course_matrix.copy()
user_course_matrix_encoded.index =
user_encoder.fit_transform(user_course_matrix.index)
user_course_matrix_encoded.columns =
course_encoder.fit_transform(user_course_matrix.columns)

# Step 4: Model - KNN for Collaborative Filtering (User-Based)
# Using NearestNeighbors for finding similar users based on ratings
knn = NearestNeighbors(metric='cosine', algorithm='brute', n_neighbors=3)
knn.fit(user_course_matrix_encoded)

# Step 5: Make Recommendations (Example: for Alice)
# Get Alice's encoded index
alice_index = user_encoder.transform(['Alice'])[0]

# Find similar users
distances, indices = knn.kneighbors(user_course_matrix_encoded.iloc[alice_index,
:].values.reshape(1, -1))

# Step 6: Display Recommendations
# Get the similar users' indices and map back to user names
similar_users = user_encoder.inverse_transform(indices.flatten())

print(f"Users similar to Alice: {similar_users}")
print("\nRecommended courses based on similar users:")

# Find courses that Alice has not rated yet and recommend those
```

```
recommended_courses = set()
for similar_user in similar_users:
    if similar_user != 'Alice':
        similar_user_courses = user_course_matrix.loc[similar_user]
        # Recommend courses that have a high rating (greater than 3) and Alice hasn't rated yet
        for course, rating in similar_user_courses.items():
            if rating > 3 and user_course_matrix.loc['Alice', course] == 0:
                recommended_courses.add(course)

# Display the recommended courses
print(recommended_courses)
```

APPENDIX B – SCREENSHOTS

OUTPUT