AI-BASED E-COMMERCE PRODUCT RECOMMENDATION SYSTEM

A MINI PROJECT-I REPORT

Submitted by

HARISH VISHNU K	**(621321205016)**
NAVEEN J N	**(621321205030)**
VISHAL S	**(621321205062)**

in partial fulfillment for the award of the degree
of

BACHELOR OF TECHNOLOGY

in

INFORMATION TECHNOLOGY

KONGUNADU COLLEGE OF ENGINEERING AND TECHNOLOGY (AUTONOMOUS)

ANNA UNIVERSITY::CHENNAI 600 025

NOVEMBER 2023

KONGUNADU COLLEGE OF ENGINEERING AND TECHNOLOGY

(AUTONOMOUS)

NAMAKKAL- TRICHY MAIN ROAD, THOTTIAM, TRICHY.

COLLEGE VISION & MISSION STATEMENT

VISION

"To become an Internationally Renowned Institution in Technical Education, Research and Development by Transforming the Students into Competent Professionals with Leadership Skills and Ethical Values."

MISSION

- Providing the Best Resources and Infrastructure.
- Creating Learner-Centric Environment and continuous Learning.
- Promoting Effective Links with Intellectuals and Industries.
- Enriching Employability and Entrepreneurial Skills.
- Adapting to Changes for Sustainable Development.

KONGUNADU COLLEGE OF ENGINEERING AND TECHNOLOGY

(AUTONOMOUS)

NAMAKKAL- TRICHY MAIN ROAD, THOTTIAM, TRICHY.

DEPARTMENT OF INFORMATION TECHNOLOGY

VISION

To produce competent IT professionals, researchers and entrepreneurs with Moral Values in the field of Information Technology.

MISSION

- Enrich the students' programming and computing skills through best teaching, learning processes, laboratory practices and through project-based learning.

- Inculcate real world challenges, emerging technologies and end the students to become entrepreneurs or make them employable.

- Inculcating moral and ethical values to serve the society and focus on students' overall development.

PROGRAM EDUCATIONAL OBJECTIVES (PEOs)

- **PEO I:** Graduates shall become IT professionals with specialization Software Engineering, Networking, Data Mining and Cloud computing.
- **PEO II:** Graduates shall build IT solutions through analysis, design and development of software and firmware solutions for real-world problems and social issues.
- **PEO III:** Graduates shall have professional ethics, team spirit, life-long learning, good oral and written communication skills and adopt corporate culture, core values and leadership skills.

PROGRAM SPECIFIC OUTCOMES (PSOs)

- **PSO1:** Professional skills: Students shall understand, analyses and develop IT applications in the field of Data Mining/Analytics, Cloud Computing, Networking etc., to meet the requirements of industry and society.
- **PSO2:** Competency: Students shall qualify at the State, National and International level competitive examination for employment, higher studies and research.

PROGRAM OUTCOMES (POs)

Engineering Graduates will be able to:

1. **Engineering knowledge**: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis**: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using the firstprinciples of mathematics, natural sciences, and engineering sciences.
3. **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 public health and safety, and cultural, societal, and environmental considerations.
4. **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.
5. **Modern tool usage**: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.
6. **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.
7. **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.
8. **Ethics**: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. **Individual and team work**: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
10. **Communication**: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able tocomprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
11. **Project management and finance**: Demonstrate knowledge and understandingof 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.
12. **Life-long learning**: Recognize the need for, and have the preparation and abilityto engage in independent and life-long learning in the broadest.

KONGUNADU COLLEGE OF ENGINEERING AND TECHNOLOGY (AUTONOMOUS)

ANNA UNIVERSITY::CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this Mini Project-I report titled "**AI-BASED E-COMMERCE PRODUCT RECOMMENDATION SYSTEM"** is a bonafide work of **HARISH VISHNU K (621321205016), NAVEEN J N (621321205030), VISHAL S (621321205062),** who carried out the mini project under my supervision.

SIGNATURE	**SIGNATURE**
Mr.N.PREMKUMAR,M.E.,(Ph.D)	**Mr.N.PREMKUMAR,M.E.,(Ph.D)**
HEAD OF THE DEPARTMENT	HEAD OF THE DEPARTMENT
Associate Professor,	Associate Professor,
Department of Information Technology,	Department of Information Technology,
Kongunadu College of Engineering and Technology (Autonomous).	Kongunadu College of Engineering and Technology (Autonomous).

Submitted for the project work and viva-voce held on ………………

INTERNAL EXAMINER	**EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

We wish to express our sincere gratitude to our beloved chairman **Dr.PSK.R.PERIASWAMY** for providing the facilitation to make us extensively internationalized to meet global competence with almost perfection.

We would like to express our sincere thanks to our Principal **Dr. R. ASOKAN, M.S., M.Tech., Ph.D.,** for the facilities and the encouragement given to the progression and completion of this project.

We proudly render our immense gratitude to **Mr. N. PREMKUMAR, M.E., (Ph.D),** for his effective leadership,encouragement and supportive guidance to this project.

We highly indebted to provide our heart full thanks to our project coordinator **Mr.J.SATHISHKUMAR, M.E., (Ph.D),** Assistant Professor, Department of Information Technology for his valuable ideas, constant encouragement and supportive guidance throughout the project.

We proudly render our thanks to our mini project guide, **Mr.N.PREMKUMAR, M.E., (Ph.D),** Associate Professor and Head of the Department, Department of Information Technology for his valuable suggestion and for continued encouragement during the execution of our project work.

We wish to extend our sincere thanks to all faculty members of Information Technology Department for their valuable suggestions, kind cooperation and encouragement on successful completion of this project.

Finally, we profound gratitude to the almighty for his presence and who gave me the bravery, privilege, power and belief to complete this project.

ABSTRACT

The AI-powered product suggestion system is a tool used in ecommerce platforms to improve user experiences and increase revenues. This system analyzes a massive amount of user data, including browsing history, purchasing behavior, and preferences, using Artificial Intelligence and Machine Learning (AIML) algorithms to deliver customized product recommendations to each individual buyer. The technology effectively suggests relevant band alluring products based on user interests and behavior patterns, boosting customer engagement and loyalty. A useful tool for e-commerce enterprises, the AIML-based product suggestion system boosts consumer satisfaction while boosting revenue development in a cutthroat online market.

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	**VIII**
	LIST OF FIGURES	**XI**
	LIST OF ABBREVIATIONS	**XII**
1	**INTRODUCTION**	**1**
	1.1 OVERVIEW	3
	1.2 PROBLEM STATEMENT	3
2	**LITERATURE SURVEY**	**4**
3	**SYSTEM ANALYSIS**	**7**
	3.1 EXISTING SYSTEM	7
	3.2 PROPOSED SYSTEM	8
4	**SYSTEM DESIGN**	**9**
	4.1 SYSTEM ARCHITECTURE	9
	4.2 SEQUENCE DIAGRAM	10
	4.3 ACTIVITY DIAGRAM	11
5	**SYSTEM SPECIFICATION**	**12**
	5.1 SOFTWARE REQUIREMENTS	12
	5.2 HARDWARE REQUIREMENTS	12

6	**SYSTEM IMPLEMENTATION**	**13**
	6.1 MODULE SPECIFICATIONS	13
	6.2 MODULE DESCRIPTION	13
7	**ALGORITHM DESCRIPTION**	**15**
	7.1 DATA LOADING AND PREPROCESSING	15
	7.2 INTERACTION MATRIX	15
	7.3 SPARSE REPRESENTATION	15
	7.4 SVD DECOMPOSITION	16
	7.5 PREDICTED RATINGS	16
	7.6 USER-BASED RECOMMENDATIONS	16
	7.7 GUI IMPLEMENTATION	16
8	**TESTING**	**17**
	8.1 SYSTEM TESTING	17
	8.2 TYPES OF TESTING	17
9	**APPENDICES**	**20**
	9.1 SAMPLE PROGRAM	20
	9.2 OUTPUTS	26
10	**CONCLUSION AND FUTURE ENHANCEMENT**	**28**
	10.1 CONCLUSION	28
	10.2 FUTURE ENHANCEMENT	28
11	**REFERENCES**	**29**

LIST OF FIGURES

FIG NO	NAME OF THE FIGURE	PAGE NO
4.1	System Architecture	9
4.2	Sequence Diagram	10
4.3	Activity Diagram	11
9.2.1	Home Page	26
9.2.2	Personalized Recommendation	26
9.2.3	Common Recommendation	27

LIST OF ABBREVIATIONS

AIML	**A**RTIFICIAL **I**NTELLIGENCE AND **M**ACHINE **L**EARNING
SVD	**S**INGULAR **V**ALUE **D**ECOMPSITION
GUI	**G**RAPHICAL **U**SER **I**NTERACE
ROM	**R**EAD **O**NLY **M**EMORY
RAM	**R**ANDOM **A**CCESS **M**EMORY
IDLE	**I**NTEGRATED **D**EVELOPMENT AND **LEARNING** **E**NVIRONMENT

CHAPTER – 1
INTRODUCTION

The Product Recommendation System using AIML report delves into the innovative integration of Artificial Intelligence Markup Language (AIML) in the domain of personalized product recommendations. In an era marked by information overload and abundant choices, this system aims to enhance user experiences by intelligently suggesting products that align with individual preferences.

Leveraging AIML's natural language processing capabilities, the system engages users in dynamic conversations, understanding their preferences and needs. Through a sophisticated combination of user profiling, contextual understanding, and data-driven recommendation algorithms, this system seeks to bridge the gap between users and products, ultimately leading to more informed and satisfying purchasing decisions.

In the ever-expanding realm of e-commerce, where countless products vie for the attention of online shoppers, the role of recommendation systems has become paramount. These systems, driven by Artificial Intelligence and Machine Learning (AIML), serve as the virtual shop assistants of the digital age, striving to guide users through the vast aisles of e-commerce platforms with personalized and relevant product recommendations. While they have indeed revolutionized the way consumers discover and engage with products, the current landscape of e-commerce recommendation systems is not without its challenges.

The prevailing systems, often reliant on rule-based mechanisms, predefined patterns, and rudimentary machine learning techniques, exhibit notable limitations in their ability to provide highly accurate and deeply personalized product suggestions. Users are presented with recommendations that may not truly align with their preferences, leading to reduced user satisfaction and, in some cases, missed sales opportunities.

In light of these challenges, there is an imperative to develop and implement an advanced AIML-based product recommendation system that not only mitigates the shortcomings of existing systems but also leverages state-of-the-art machine learning techniques to provide precise and relevant product recommendations. This system, by virtue of its deep learning capabilities, aims to adapt to changing user preferences and market trends. However, the incorporation of advanced techniques comes with its own set of complexities, such as the need for substantial data and computational resources. Furthermore, these systems often grapple with the fast-paced, ever-changing dynamics of the e-commerce market, where user tastes evolve, and new trends emerge rapidly.

This paper embarks on a journey to explore the pressing issues that plague current e-commerce recommendation systems, delving into their limitations and the consequences they pose for both users and e-commerce platforms. Furthermore, it seeks to identify and understand the challenges associated with the development and implementation of a highly sophisticated AIML-based recommendation system. By addressing these challenges, the proposed system aspires to enhance the overall online shopping experience, ensuring not only highly accurate product recommendations but also preserving user privacy, fairness, and transparency. In doing so, it endeavors to bridge the gap between user expectations and the capabilities of existing systems, ushering in a new era of highly personalized and engaging online shopping.

This report explores the conceptual foundation, technical architecture, and practical implications of employing AIML to craft a cutting-edge product recommendation solution.

.

1.1 OVERVIEW

An AI-based e-commerce product recommendation system leverages advanced algorithms and data analytics to enhance the shopping experience. It operates by collecting and analyzing user data, including browsing history, purchase history, and user preferences. This data is processed to generate personalized product recommendations. The system can employ various recommendation techniques, such as collaborative filtering, content-based filtering, and hybrid methods. Machine learning models are often used to predict user preferences and suggest products accordingly. Real-time data updates ensure that recommendations stay relevant. Additionally, the system can incorporate factors like product popularity and user ratings to improve recommendations. AI-driven e-commerce recommendation systems have become a crucial tool for boosting sales and enhancing user satisfaction in online shopping platforms

1.2 PROBLEM STATEMENT

Current e-commerce recommendations fall short in providing accurate and personalized suggestions, impacting user satisfaction and sales. Rule-based systems hinder adaptability to evolving user preferences and market trends. The challenge is to build an AIML-based system that overcomes data and resource limitations for enhanced accuracy. The goal is to create a recommendation system that not only offers precise product recommendations but also maintains user privacy and fairness, ultimately enhancing the overall online shopping experience.

CHAPTER 2

LITERATURE SURVEY

"Deep Neural Networks for YouTube Recommendations" by Paul Covington, Jay Adams, and Emre Sargin. Discusses the architecture and methods used by YouTube's recommendation system. Highlights the importance of embedding neural networks in improving recommendation quality. Addresses the challenges of training deep learning models on large-scale datasets. Emphasizes the role of reinforcement learning in enhancing recommendations. Provides insights into how deep learning powers real-time video suggestions on YouTube.

"A Survey of Recommender Systems in E-Commerce" by Ricardo Baeza-Yates and Paolo Boldi. Provides an overview of various types of recommender systems, including collaborative filtering, content-based, and hybrid approaches .Discusses the challenges and limitations of e-commerce recommendation systems. Highlights the importance of personalization and diversity in recommendations. Addresses the role of context in improving e-commerce recommendations. Offers valuable insights into the research landscape of recommender systems in e-commerce.

"Context-Aware Event Recommendation in Event-based Social Networks" by Mirza Shukurov, et al. Focuses on context-aware event recommendations in the context of social networks. Discusses the importance of considering contextual information such as user preferences and social connections. Presents methods for improving the relevance and quality of event suggestions. Highlights the application of machine learning techniques in event recommendation. Provides insights into enhancing user engagement and event discovery.

"Recommender Systems" by Francesco Ricci, Lior Rokach, and Bracha Shapira. Offers a comprehensive introduction to the principles of recommender systems. Discusses various recommendation algorithms, including collaborative filtering and content-based methods. Highlights the importance of evaluation metrics for assessing recommendation system performance. Addresses the challenges of cold start problems and scalability in recommendations. Serves as a valuable resource for understanding the fundamental concepts of recommendation systems.

"Deep Learning for Recommender Systems" by Alexandros Karatzoglou, et al. Focuses on the application of deep learning in enhancing recommendation accuracy. Discusses the use of neural networks and embedding techniques for personalized recommendations. Highlights the role of deep content-based models in capturing user preferences. Presents case studies of deep learning applications in recommender systems. Emphasizes the potential of deep learning to improve recommendation quality in various domains, including e-commerce.

"Survey of Collaborative Filtering Techniques" by A. M. Sarwar, et al.Provides an overview of various collaborative filtering techniques, including user-based and item-based methods. Discusses the importance of addressing data sparsity and the cold start problem in recommendations. Highlights the challenges and advantages of matrix factorization techniques. Emphasizes the need for effective similarity measures in collaborative filtering. Offers insights into the scalability and efficiency of collaborative filtering approaches.

"A Survey of Recommender Systems in Online Social Networks" by U. S. Ratha and Dr. S. Anjanadevi. Explores the role of recommender systems in online social networks, focusing on personalized content recommendations. Discusses the impact of social connections and user behaviors on recommendation quality. Highlights the challenges of privacy and trust in social network recommendations.Addresses the importance of real-time recommendations in social media platforms.Offers insights into the diversity and serendipity of recommendations in social networks.

"Learning to Rank for Personalized Fashion Recommender Systems" by Alexandros Karatzoglou, et al. Presents a recommendation system tailored for fashion products using learning-to-rank techniques. Highlights the significance of feature engineering in fashion recommendation. Discusses the incorporation of implicit user feedback for improved ranking. Emphasizes the role of personalized fashion recommendations in enhancing user satisfaction. Addresses the need for user-centric and interpretable fashion recommendations.

"A Survey of Recommender Systems for Different Applications" by Adel M. Almaleh. Offers a comprehensive survey of recommender systems applied to various domains. Discusses the common challenges and solutions in recommendation across different application areas. Highlights the significance of domain-specific features in recommendation algorithms. Addresses the importance of understanding user preferences and behavior for personalized recommendations. Provides insights into the diversity of applications, including e-commerce, news, and entertainment.

CHAPTER-3

SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

The existing product recommendation systems employing Artificial Intelligence and Machine Learning (AIML) heavily depend on rule-based mechanisms and predefined patterns to generate recommendations. While these systems have demonstrated their effectiveness in providing personalized product suggestions to users, they have notable limitations. One of the primary shortcomings is their inability to fully grasp complex and evolving user preferences, as they operate within the confines of predetermined rules. This can result in the system struggling to keep up with rapidly changing trends and failing to deliver highly accurate recommendations for users with diverse and evolving tastes. Some of these systems have attempted to incorporate basic machine learning techniques to analyze user interactions and improve recommendation accuracy. However, these efforts may fall short in achieving the depth and sophistication necessary for making precise predictions, as they often lack the capacity to adapt to the intricate nuances of individual user behaviors and preferences, which are essential for delivering truly tailored and effective recommendations.

DISADVANTAGES

- Reliance on rule-based systems and predefined patterns restricts adaptability to evolving user preferences and trends.
- Struggles to provide precise recommendations for users with diverse or evolving tastes.
- Incorporation of basic machine learning may lack the depth and sophistication required for accurate predictions.

3.2 PROPOSED SYSTEM

The proposed system for a product recommendation system, leveraging Artificial Intelligence and Machine Learning (AIML), would undoubtedly benefit from the incorporation of cutting-edge techniques such as machine learning algorithms, collaborative filtering, and deep learning. These advanced methodologies have the potential to significantly enhance recommendation accuracy by enabling the system to learn and adapt from user interactions. By analyzing user behavior and preferences, the system can offer highly personalized product suggestions, thereby improving user satisfaction and engagement. However, it's worth noting that the implementation of these sophisticated techniques could pose challenges. To effectively leverage machine learning, collaborative filtering, and deep learning, the system would necessitate substantial amounts of data to train its models and the computational resources required for processing and analyzing this data. Therefore, a comprehensive approach that balances the advantages of these advanced techniques with the practical constraints of data and resources will be essential for the successful development and deployment of such an advanced AIML-based recommendation system.

ADVANTAGES

- Offers highly personalized product suggestions based on individual user preferences and behaviors.
- Utilizes machine learning, collaborative filtering, and deep learning to enhance recommendation accuracy.
- Increases user satisfaction and engagement by providing relevant and tailored recommendations.
- Drives sales by promoting products that align with user interests and needs.

CHAPTER 4

SYSTEM DESIGN

4.1 SYSTEM ARCHITECTURE

AI-based e-commerce product recommendation system involves a multi-layered approach designed to efficiently process, analyze, and deliver personalized recommendations. At the foundation, a data layer captures and stores extensive user data, including browsing history, purchase patterns, and feedback. The processing layer incorporates machine learning algorithms, employing techniques like collaborative filtering and deep learning to extract meaningful insights from the data. This layer continuously learns and adapts to evolving user preferences. The recommendation engine, a pivotal component, resides in the application layer, orchestrating the generation and delivery of tailored product suggestions based on the insights gained.

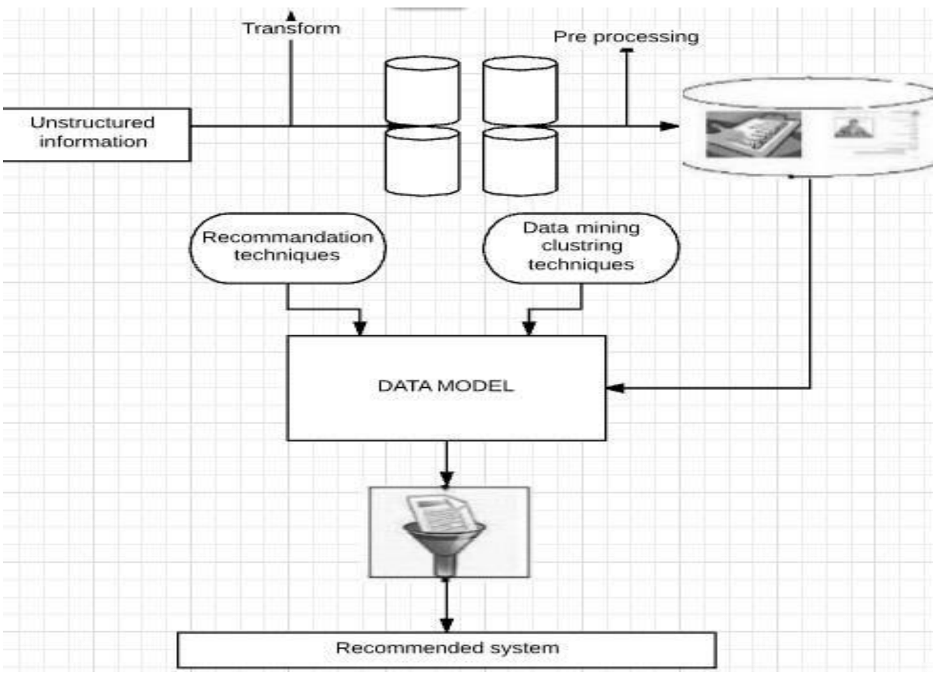

Fig 4.1: System Architecture

4.2 SEQUENCE DIAGRAM

The sequence diagram outlines the interactions between various components to deliver personalized product suggestions to users. The process typically begins with the user interacting with the e-commerce platform, triggering a request for recommendations. The front-end sends this request to the recommendation engine, which utilizes machine learning algorithms to analyze user preferences, purchase history, and browsing behavior.

Upon receiving the request, the recommendation engine communicates with the user database and product database to gather relevant information. Subsequently, it processes this data using AI algorithms, such as collaborative filtering or content-based filtering, to generate tailored product recommendations.

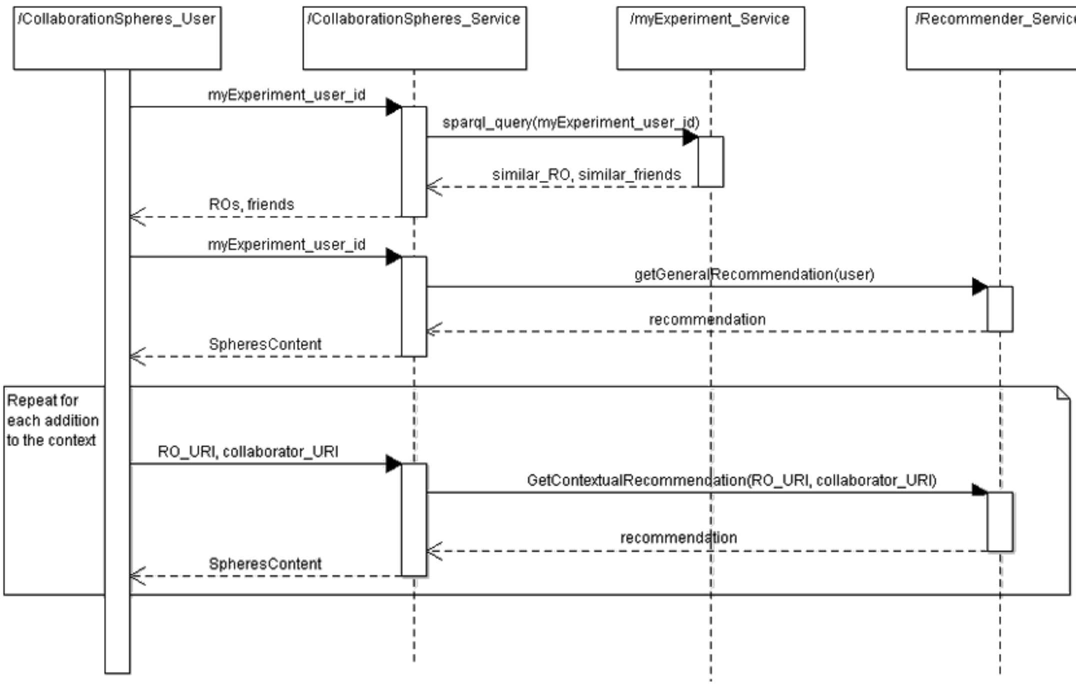

Fig 4.2: Sequence Diagram

4.3 ACTIVITY DIAGRAM

An activity diagram for an AI-based e-commerce product recommendation system is a visual representation that illustrates the dynamic flow of actions and processes within the system. This diagram is designed to capture the sequential steps and interactions involved in generating personalized product recommendations for users.

The activity diagram provides a holistic view of the system's functionalities, emphasizing the steps where computational intelligence, driven by AI algorithms , analyzes user data to dynamically tailor product recommendations.

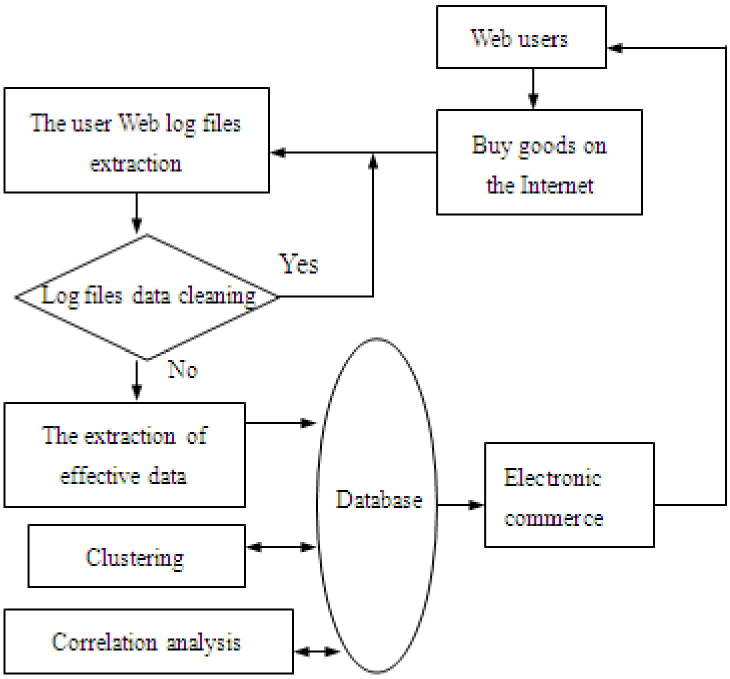

Fig 4.3: Activity Diagram

CHAPTER 5

SYSTEM SPECIFICATION

5.1 SOFTWARE REQUIREMENTS

Operating System	:	Windows 7
Front End	:	HTML, CSS
Back End	:	JavaScript, Python
Language	:	Python
Tool	:	Python IDLE

5.2 HARDWARE REQUIREMENTS

Processor	:	Intel Core i3
Hard Disk	:	256 GB
RAM	:	4 GB

CHAPTER 6

SYSTEM IMPLEMENTATION

Data collection and preprocessing are crucial. Gather user behavior data such as browsing history, purchase history, and demographic information. Clean and structure this data for analysis . Next, select an appropriate recommendation algorithm, like collaborative filtering or content-based filtering. Customize the algorithm to suit your e-commerce platform's needs

6.1 MODULE SPECIFICATIONS

- Importing the dataset.
- Preprocessing the data.
- Recommendation Algorithm.
- Real-Time Data Processing.
- Generating Recommendation.

6.2 MODULES DESCRIPTION

6.2.1 IMPORTING THE DATASET

Importing the dataset module is the initial step in building a recommendation system. It involves retrieving data from various sources, such as user interactions, product details, and historical records. This data serves as the foundation for the recommendation system, allowing it to learn and make suggestions based on user behavior and preferences.

6.2.2 PREPROCESSING THE DATA

Data preprocessing is a critical module in recommendation systems. It involves cleaning, transforming, and structuring the raw data to make it suitable for analysis. This module handles tasks like handling missing values, removing duplicates, and standardizing data formats. Clean and well-structured data is essential for accurate recommendations.

6.2.3 RECOMMENDATION ALGORITHM

Data preprocessing is a critical module in recommendation systems. It involves cleaning, transforming, and structuring the raw data to make it suitable for analysis. This module handles tasks like handling missing values, removing duplicates, and standardizing data formats. Clean and well-structured data is essential for accurate recommendations.

6.2.4 REAL-TIME DATA PROCESSING

The recommendation generation module combines the results of the recommendation algorithm with user-specific data to produce a list of product suggestions. It takes into account the user's historical behavior, real-time interactions, and other contextual information to tailor the recommendations to the individual user. This module is responsible for delivering the final personalized recommendations to users.

6.2.5 GENERATING RECOMMENDATIONS

The recommendation generation module combines the results of the recommendation algorithm with user-specific data to produce a list of product suggestions. It takes into account the user's historical behavior, real-time interactions, and other contextual information to tailor the recommendations to the individual user. This module is responsible for delivering the final personalized recommendations to users.

In summary, these modules work together to import and preprocess data, apply recommendation algorithms to generate suggestions, and process real-time data to ensure that the recommendations remain relevant and up-to-date. The ultimate goal is to provide users with accurate, personalized, and engaging product recommendations in the dynamic e-commerce environment.

CHAPTER 7

ALGORITHM DESCRIPTION

SINGULAR VALUE DECOMPOSITION (SVD)

Singular Value Decomposition (SVD) is used in product recommendation systems as a collaborative filtering technique to make personalized product recommendations. Collaborative filtering methods, including SVD, are based on the idea that users who have interacted with or shown interest in similar products will have similar preferences and can be used to make recommendations.

7.1 DATA LOADING AND PREPROCESSING

- Load the user-item interaction data, typically represented as a user-item matrix where rows represent users, columns represent items, and the values represent user-item interactions (e.g., ratings or purchase history).
- Preprocess the data by handling missing values, normalizing ratings, or performing other necessary data cleaning tasks.

7.2 INTERACTION MATRIX

An interaction matrix is a fundamental data structure in recommendation systems and collaborative filtering. It represents user-item interactions, where rows correspond to users and columns correspond to items. Create the user-item interaction matrix based on the preprocessed data. This matrix represents the interactions between users and items. It may be a dense or sparse matrix, depending on the data and its sparsity.

7.3 SPARSE REPRESENTATION

In real-world recommendation systems, the user-item interaction matrix is often sparse because most users do not interact with all items. To handle this sparsity, you can create a sparse representation of the matrix, which only stores non-zero values and their indices efficiently.

7.4 SVD DECOMPOSITION

- Perform the Singular Value Decomposition on the user-item interaction matrix. SVD decomposes the matrix into three matrices: U, Σ, and V^T, where U represents the user space, Σ contains the singular values, and V^T represents the item space.
- Typically, you would approximate the original matrix by retaining only the top-k singular values and their associated columns in U and V^T, reducing the dimensionality of the problem.

7.5 PREDICTED RATINGS

Using the reduced SVD representation, you can calculate predicted ratings for all user-item pairs. The predicted rating for a user and an item is given by the dot product of the user's vector (from U) and the item's vector (from V^T).

7.6 USER-BASED RECOMMENDATIONS

Once you have the predicted ratings, you can recommend items to users based on these predictions. For a specific user, you can recommend items with the highest predicted ratings that the user has not interacted with before. This can be done by sorting the items by their predicted ratings.

7.7 GUI IMPLEMENTATION

GUI implementation involves creating the graphical user interface for a software application. It begins with careful design, determining the layout, and selecting the programming language and framework best suited for the project, whether it's for desktop, web, or mobile

CHAPTER – 8

TESTING

8.1 SYSTEM TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies, and/or a finished product. It is the process of exercising software with the intent of ensuring that the software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement.

8.2 TYPES OF TESTING

8.2.1 UNIT TESTING

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application. It is done after the completion of an individual unit before integration. This is structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at the component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results. By conducting unit tests, developers can quickly detect and rectify errors, track the performance of different project components, and maintain the reliability and quality of their machine learning models, leading to more robust and maintainable ML projects.

8.2.2 INTEGRATION TESTING

Integration testing is the second level of the software testing process that comes after unit testing. In this testing, units or individual components of the software are tested in a group. The focus of the integration testing level is to expose defects at the time of interaction between integrated components or units. Integration tests are one program. Testing is event-driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although they were individually satisfactory, as shown by successfully unit testing, components the combination of components is correct and consistent.

8.2.3 FUNCTIONAL TESTING

Functional tests provide systematic demonstrations that functions tested are Available as specified by the business and technical requirements, system documentation, and user manuals. Functional testing is centered on the following items:

Valid Input : Identified classes of valid input must be accepted.

Invalid Input : Identified classes of invalid input must be rejected.

Functions : Identified functions must be exercised.

Output : Identified classes of application outputs must be exercised.

Systems/Pro : Interfacing systems or procedures must be invoked.

The organization and preparation of functional tests are focused on requirements, key functions, and cases. In addition, systematic coverage pertaining to identifying Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

8.2.4 SYSTEM TESTING

System testing ensures that the entire integrated software system meets requirements. It is a level of testing that validates the complete and fully integrated software product. The purpose of a system test is to evaluate the end-to-end system specifications. Usually, the software is only one element of a larger computer-based a series of tests different whose sole purpose is to exercise the full computer-based system. There are three ways to test a program:

- For Correctness
- For Implementation efficiency
- For Computational Complexity

Tests for implementation efficiency attempt to find ways to make a correct program faster or use less storage. It is a code-refining process, which reexamines the implementation phase of algorithm development. Tests for computational complexity amount to an experimental analysis of the complexity of an algorithm or an experimental comparison of two or more algorithms, which solve the same problem.

A quality team deputed by the management verified all the necessary documents and tested the Software while entering the data at all levels. The development process involves various types of testing. Each test type addresses a specific testing requirement.

CHAPTER 9

APPENDICES

9.1 SAMPLE PROGRAM

<u>Epr.py:</u>

```
import tkinter as tk
from tkinter import ttk, scrolledtext
from tkinter.constants import END
from PIL import Image, ImageTk
import numpy as np
import pandas as pd
from scipy.sparse.linalg import svds
from scipy.sparse import csr_matrix
# Load and preprocess data (You need to have 'ratings_Electronics.csv' in the
same directory)
df = pd.read_csv('ratings_Electronics.csv', header=None)
df.columns = ['user_id', 'prod_id', 'rating', 'timestamp']
df = df.drop('timestamp', axis=1)
# Create the interaction matrix
df_copy = df.copy(deep=True)
counts = df['user_id'].value_counts()
df_final = df[df['user_id'].isin(counts[counts >= 50].index)]
final_ratings_matrix = df_final.pivot(index='user_id', columns='prod_id',
values='rating').fillna(0)
final_ratings_sparse = csr_matrix(final_ratings_matrix.values)
# Singular Value Decomposition
```

```
U, s, Vt = svds(final_ratings_sparse, k=50)
sigma = np.diag(s)
all_user_predicted_ratings = np.dot(np.dot(U, sigma), Vt)
preds_df = pd.DataFrame(abs(all_user_predicted_ratings),
columns=final_ratings_matrix.columns)
preds_matrix = csr_matrix(preds_df.values)
def recommend_items():
    user_index = int(user_entry.get())
    num_recommendations = int(recommendation_entry.get())
    user_ratings = final_ratings_sparse[user_index, :].toarray().reshape(-1)
    user_predictions = preds_matrix[user_index, :].toarray().reshape(-1)
    temp = pd.DataFrame({'user_ratings': user_ratings, 'user_predictions':
user_predictions})
    temp['Recommended Products'] = final_ratings_matrix.columns
    temp = temp.set_index('Recommended Products')
    temp = temp.loc[temp.user_ratings == 0]
    temp = temp.sort_values('user_predictions', ascending=False)
    recommendations = temp.head(num_recommendations).index
    recommended_products.config(state="normal")
    recommended_products.delete(1.0, END)
    recommended_products.insert("insert", '\n'.join(recommendations))
 recommended_products.config(state="disabled")
```

Index.html:

```
<!DOCTYPE html>
<html lang="en">
<head>
```

```
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>My E-Commerce Store</title>
<link rel="stylesheet" type="text/css" href="{{ url_for('static',
filename='styles.css') }}">
</head>
<body>
<header>
<h1>Welcome to My E-Commerce Store</h1>
</header>
<nav>
<a href="/home">Home</a>
<a href="/products">Products</a>
<a href="/recommendations">Recommendations</a>
</nav>
<h1> Personalized Product Recommendation System</h1>
<form action="/recommend" method="post">
<label for="user-entry">Enter User Index:</label>
<input type="text" name="user_index" placeholder="User Index"><br>
<label for="recommendation-entry">Number of Recommendations:</label>
<input type="text" name="num_recommendations" placeholder="Number of
Recommendations"><br>
<input type="submit" value="Recommend">
</form><br>
<label for="recommended-products">Recommended Products for the Celebrity
user:</label><br>
```

```
<textarea id="recommended-products" rows="10" readonly></textarea>
<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
<script>
$(document).ready(function () {
$('form[action="/recommend"]').submit(function (e) {
e.preventDefault();
const userIndex = $('input[name="user_index"]').val();
const numRecommendations = $('input[name="num_recommendations"]').val();
$.post('/recommend', { user_index: userIndex, num_recommendations:
numRecommendations }, function (data) {
$('#recommended-products').val(data.recommendations.join('\n'));
});});
<div class="footer">
<div class="footer-content">
<!-- Footer Section 1 - Company Info -->
<div class="footer-section">
<h4>Company</h4>
<ul>
<li>About Us</a></li>
<li>Contact Us</a></li>
<li>Privacy Policy</a></li>
<li>Terms of Service</a></li>
</ul>
</div>
<!-- Footer Section 2 - Customer Service -->
<div class="footer-section">
<h4>Customer Service</h4>
```

```
<ul>
<li>FAQs</a></li>
<li>Shipping</a></li>
<li>Returns</a></li>
<li>Track Your Order</a></li>
</ul>
</div>
<!-- Footer Section 3 - Explore -->
<div class="footer-section">
<h4>Explore</h4>
<ul>
<li>Products</a></li>
<li>Deals</a></li>
<li>Blog</a></li>
<li>Affiliate Program</a></li>
</ul>
</div>
<!-- Footer Contact Info -->
<div class="footer-contact">
<h4>Contact Us</h4>
<p>
ABC E-Commerce Platform<br>
123 Main Street<br>
Trichy,Tamil Nadu<br><br>
Email: info@abdecommerce.com<br>
Phone: (123) 456-7890
</p>
```

```
</div>
</div>
</div>
</body></html>
```

Server.js:

```
 const express = require('express');
const { spawn } = require('child_process');
const app = express();
const port = process.env.PORT || 5000;
app.use(express.static('static'));
app.use(express.urlencoded({ extended: true }));
app.get('/', (req, res) => {
    res.sendFile(__dirname + '/templates/index.html');
});
app.post('/recommend', (req, res) => {
 const user_index = req.body.user_index;
 const num_recommendations = req.body.num_recommendations;

const pythonProcess = spawn('python', ['app.py', user_index,
num_recommendations]);
pythonProcess.stdout.on('data', (data) => {
const recommendations = data.toString().split('\n').filter(Boolean);
res.send({ recommendations });
});
pythonProcess.stderr.on('data', (data) => {
console.error(data.toString());
```

```
res.status(500).send('Error generating recommendations');
});
});
app.post('/common_recommendations', (req, res) => {
const num_common_recommendations =
req.body.num_common_recommendations;
const pythonProcess = spawn('python', ['app.py',
num_common_recommendations]);
pythonProcess.stdout.on('data', (data) => {
const common_recommendations = data.toString().split('\n').filter(Boolean);
res.send({ common_recommendations });
});
pythonProcess.stderr.on('data', (data) => {
console.error(data.toString());
res.status(500).send('Error generating common recommendations');
});
});

app.post('/recommend', (req, res) => {
const user_index = req.body.user_index;
const num_recommendations = req.body.num_recommendations;
const pythonProcess = spawn('python', ['app.py', user_index,
num_recommendations]);
pythonProcess.stdout.on('data', (data) => {
const recommendations = data.toString().split('\n').filter(Boolean);
res.send({ recommendations });
});
```

```
pythonProcess.stderr.on('data', (data) => {
console.error(data.toString());
res.status(500).send('Error generating recommendations');
});
app.listen(port, () => {
console.log(`Server is running on port ${port}`);
});
```

9.2 OUTPUTS

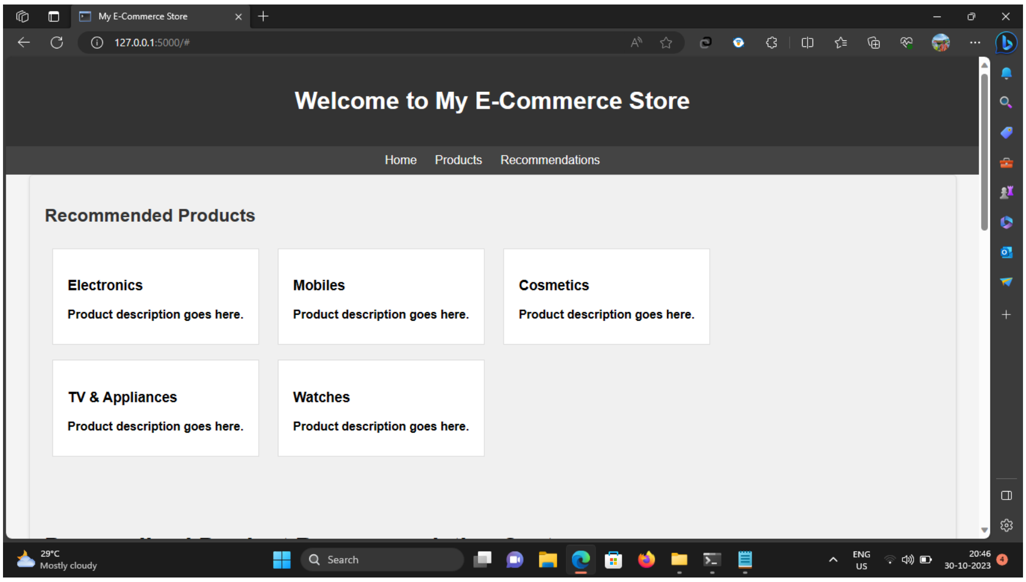

Fig 9.2.1: Home Page : It is the home page of our website which gives personalized and common recommendations to users. Users can get their recommendations easily.

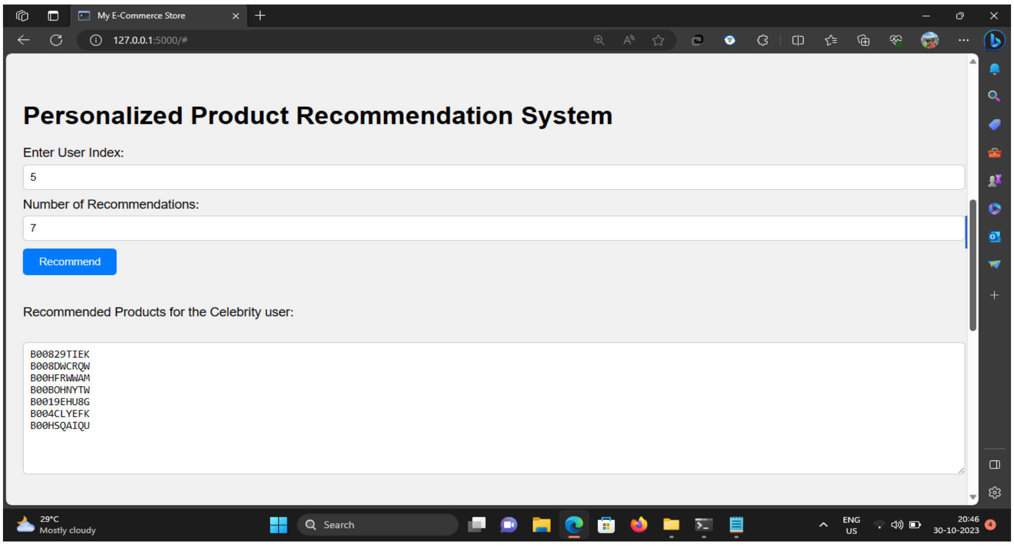

Fig 9.2.2: It demonstrates the page where the user can get their personalized recommendation by entering their user-id and number of recommendations. Our system will provide the recommendations after entering the user-id.

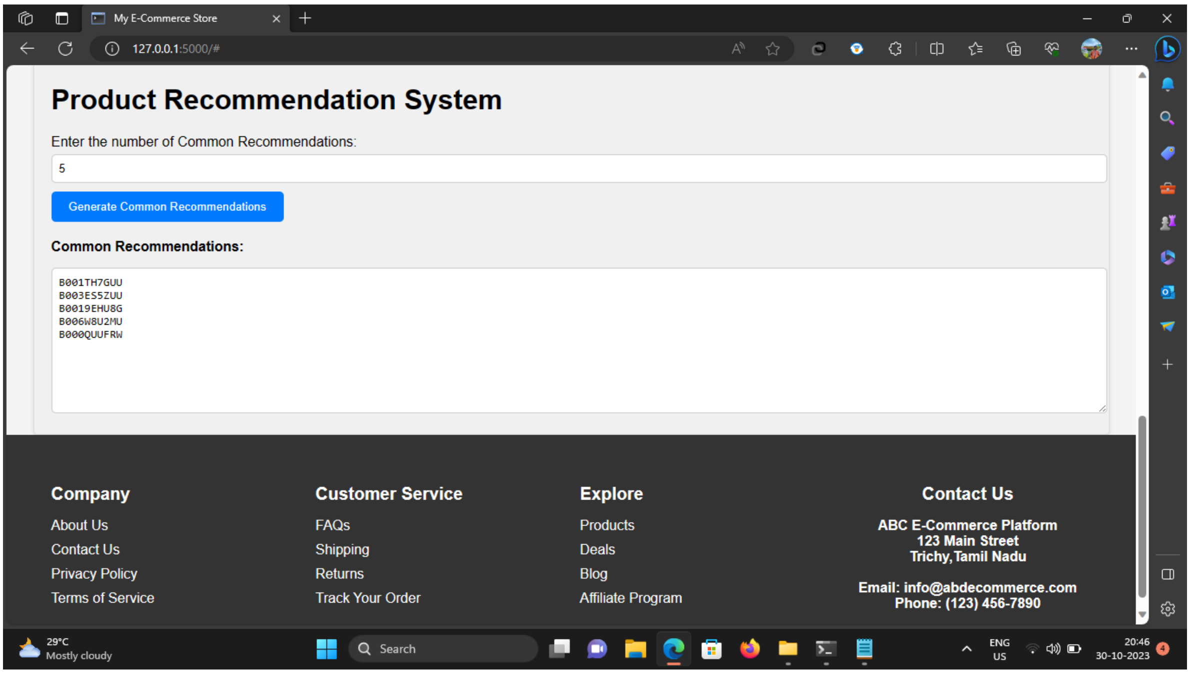

Fig 9.2.3: It demonstrates the page where the user can get their common recommendation by entering the number of recommendations. Our system will provide the recommendations based on the highest rating products.

CHAPTER 10

CONCLUSION AND FUTURE ENHANCEMENT

10.1 CONCLUSION

In conclusion, e-commerce product recommendation systems are indispensable tools in the modern online shopping landscape. These systems have evolved significantly, incorporating a wide array of techniques ranging from collaborative filtering and content-based recommendations to cutting-edge deep learning models. They not only enhance user experience by providing personalized suggestions but also boost sales and customer engagement. Moreover, ongoing research in the field, such as incorporating fairness and addressing issues of bias, continues to improve the effectiveness and ethical implications of recommendation systems. As e-commerce platforms strive to tailor their services to individual preferences, the development and refinement of recommendation systems will remain a crucial area of focus, ensuring that consumers receive the most relevant and satisfying shopping experiences.

10.2 FUTURE ENHANCEMENT

The future work of the project includes improving the efficiency of the system. And it should also be able to give appropriate recommendations to the users who don't have any previous purchase history or to the new users. In future we can try to use recurrent neural networks and deep learning. With the help of deep learning techniques we can overcome some of the drawbacks of the matrix factorization technique. Deep learning uses recurrent neural networks to accommodate time in the recommender system which is not possible in the matrix factorization method.

CHAPTER-11

REFERENCES

[1] [Adomavicius and Tuzhilin, 2005] Adomavicius, G. and Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE Trans. on Knowl. and Data Eng., 17(6):734–749.Z.

[2] Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2001). Item-based Collaborative Filtering Recommendation Algorithms. In Proceedings of the 10th International Conference on World Wide Web.

[3] D. Burke, R. (2002). Hybrid Recommender Systems: Survey and Experiments. User Modeling and User-Adapted Interaction, 12(4), 331-370.

[4] He, X., Liao, L., Zhang, H., Nie, L., Hu, X., & Chua, T. S. (2017). Neural Collaborative Filtering. arXiv preprint arXiv:1708.05031.

[5] Cheng, H. T., Koc, L., Harmsen, J., Shaked, T., Chandra, T., Aradhye, H., ... & Tov, J. Y. (2016). Wide & Deep Learning for Recommender Systems. In Proceedings of the 1st Workshop on Deep Learning for Recommender Systems.

[6] Aggarwal, C. C. (2016). Evaluating Recommendation Systems. In Recommender Systems (pp. 9-32). Springer.

[7] Covington, P., Adams, J., & Sargin, E. (2016). Deep Neural Networks for YouTube Recommendations. In Proceedings of the 10th ACM Conference on Recommender Systems.

[8] Covington, P., Adams, J., & Sargin, E. (2016). Deep Neural Networks for YouTube Recommendations. In Proceedings of the 10th ACM Conference on Recommender Systems.

[9] Covington, P., Adams, J., & Sargin, E. (2016). Deep Neural Networks for YouTube Recommendations. In Proceedings of the 10th ACM Conference on Recommender Systems.

[10] Li, L., Chu, W., Langford, J., & Schapire, R. E. (2010). A Contextual-Bandit Approach to Personalized News Article Recommendation. In Proceedings of the 19th International Conference on World Wide Web.

[11] Tintarev, N., & Masthoff, J. (2007). A Survey of Explanations in Recommender Systems. In User Modeling and User-Adapted Interaction, 17(4), 259-291.

[12] Ekstrand, M. D., Riedl, J. T., & Konstan, J. A. (2018). Losing Diversity in Recommender Systems. In Proceedings of the 13th ACM Conference on Recommender Systems.