RV Educational Institutions ®
RV College of Engineering ®
Autonomous Institution Affiliated to Visvesvaraya Technological University, Belagavi
Approved by AICTE, New Delhi

Go, change the world

Major Project
18MCA61

on

A Web Based Recommendation for Learners using Natural Language Processing

Submitted
by
Chaitra B V
1RV19MCA21

Under the Guidance
of

Dr. S Anupama Kumar
Associate Professor
Department of MCA
RV College of Engineering®
Bengaluru-59

Submitted in partial fulfillment of the requirements for the award of degree Of

MASTER OF COMPUTER APPLICATIONS

2022

RV COLLEGE OF ENGINEERING®

(Autonomous Institution Affiliated to Visvesvaraya Technological University, Belagavi)

DEPARTMENT OF
MASTER OF COMPUTER APPLICATIONS

Bengaluru– 560059

CERTIFICATE

Certified that the project work titled **"A Web Based Recommendation for Learners using Natural Language Processing"** carried out by **Chaitra B V, USN: 1RV19MCA21** , a bonafide student of **RV College of Engineering®, Bengaluru** submitted in partial fulfilment for the award of **Master of Computer Applications** of **RV College of Engineering®, Bengaluru affiliated to Visvesvaraya Technological University, Belagavi** during the year **2021-22**. It is certified that all corrections/suggestions indicated for internal assessment have been incorporated in the report deposited in the departmental library. The project report has been approved as it satisfies the academic requirement in respect of project work prescribed for the said degree.

Dr. S Anupama Kumar	**Dr. Andhe Dharani**	**Dr. K. N. Subramanya**
Associate Professor	Professor and Director	Principal
Department of MCA	Department of MCA	RVCE, Bengaluru–59
RVCE, Bengaluru –59	RVCE, Bengaluru–59	PRINCIPAL RV COLLEGE OF ENGINEERING BENGALURU - 560 059

RV COLLEGE OF ENGINEERING®

(Autonomous Institution Affiliated to Visvesvaraya Technological University, Belagavi)

DEPARTMENT OF
MASTER OF COMPUTER APPLICATIONS

Bengaluru– 560059

DECLARATION

I, **Chaitra B V** student of sixth semester MCA in **Department of Master of Computer Applications**, RV College of Engineering®, Bengaluru declare that the project titled **"A Web Based Recommendation for Learners using Natural Language Processing"** has been carried out by me. It has been submitted in partial fulfilment of the course requirements for the award of degree in **Master of Computer Applications** of RV College of Engineering®, Bengaluru affiliated to Visvesvaraya Technological University, Belagavi during the academic year **2021-22**. The matter embodied in this report has not been submitted to any other university or institution for the award of any other degree or diploma

Chaitra B V

Date of Submission: **Signature of the Student**

Student Name: Chaitra B V

USN: 1RV19MCA21

Department of MCA

RV College of Engineering®

Bengaluru-560059.

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the success of any work would be incomplete unless I mention the name of the people, who made it possible, whose constant guidance and encouragement served a beacon light and served our effort with success.

I express my wholehearted gratitude to Dr. **Subramanya K N**, Principal, RV College of Engineering® for providing me an opportunity.

I express my special thanks to **Dr. Andhe Dharani**, Professor, and Director, Department of MCA, RV College of Engineering®, Bengaluru for her constant support and guidance.

I express my sincere thanks and wholehearted credit to my Internal guide **Dr. S Anupama Kumar**, Associate Professor, Department of MCA, RV College of Engineering®, Bengaluru for her constant encouragement, support, and guidance during the project work.

I am also thankful to lab in-charge staff and all faculty of the department for their help and support during the project.

On a moral personal note, my deepest appreciation and gratitude to my beloved family, who have been a fountain of inspiration and have provided unrelenting encouragement and support.

Chaitra B V
Department of MCA
RV College of Engineering®
Bengaluru-59

ABSTRACT

Natural language processing (NLP) refers to the branch of computer science and more specifically, the branch of artificial intelligence or AI concerned with giving computers the ability to understand text and spoken words in much the same way human beings can. NLP combines computational linguistics rule-based modelling of human language with statistical, machine learning, and deep learning models. Together, these technologies enable computers to process human language in the form of text or voice data and to 'understand' its full meaning, complete with the speaker or writer's intent and sentiment.

A recommendation system is a subclass of Information filtering Systems that seeks to predict the rating or the preference a user might give to an item. In simple words, it is an algorithm that suggests relevant items to users by taking data from the user's previous activities and running a similarity analysis. Recommendation techniques are extensively studied and mainly classified into three categories: content-based, collaborative filtering (CF)-based, and knowledge-based. The three categories all have their own characteristics and suitable application scenarios. Content-based recommendation methods profile users with their past behaviours and recommend items that are similar to user profiles. CF-based recommendation methods assume that similar users will prefer similar items and generate recommendations with the collaboration of users. Knowledge-based recommendation uses explicit knowledge of users/items and business rules to profile user interest and provide recommendations. The hybrid of the above three techniques can be used to build Learner's recommendation system.

The outcome of this project would be to build a web-based recommendation system that suggests leaners' relevant items to the users based on users' preference with high accuracy. A Web based recommendation system for learners using NLP is web application that focus particularly learners and suggest items based on learner's context. The items can be learning styles, learning materials, learning courses, learning routes, books, Journals, learning videos etc.

Table of Contents

List of Tables

List of Figures

CHAPTER 1: INTRODUCTION

This chapter gives brief introduction to a web based recommendation for learners using NLP

1.1 Project Description:

A recommendation system is the application of Natural Language Processing (NLP) which categorises or ranks products or users. Machine learning is a branch of research that aims to better understand and develop "learning" methods, i.e. how data would be utilized to improve task performance. Recommender systems have evolved into a significant technology for supporting people to make educated choices especially in the era of substantial data in which learners have to make choices from a huge number of results. Quite a lot of Recommender system models and techniques have already been proposed and nearly all of them have achieved great success. Amongst them, the content-based and collaborative Recommender algorithms are two representative ones. Their efficacy has been demonstrated by both research and industry communities. People often get confused with search engines and recommender systems. Users will try a search engine if they have specific needs whereas users will try a recommender system if they do not know what they want now or they can not use keywords to describe needs.

Recommendation systems (RS) are a new and rapidly growing topic of study. Significant advancements in internet technology and e-commerce have generated growing interest in this study issue. There are several advantages of RS over e-commerce. Help novice clients with online buying, sell materials together, and build customer loyalty are three methods to improve users RS e-commerce system. The introduction of Amazon's shared filtering (CF) approach, which was successfully deployed in the late 1990s, was a highlight of RS research. As Amazon's success grew in prominence, more online businesses began to include RS into their websites. The CF technique has been registered as a US patent by Amazon. Because the major purpose of the RS field is to filter information, it may be thought of as a subset of filtering information.

A recommendation system for learners has received major attention in recent years as a means to cope with information explosion in learners' recommendation systems

as well as provide relevant recommendations to learners. The main problem in this system is to develop the model , particularly for learners, and to suggest learners' relevant materials to the users based on users' preference with high accuracy.Another problem includes Lack of Data, Changing Data, Changing User Preferences, Complexity. The project aims to implement a web based recommendation system that fulfils the following objectives.

1. Build a web-based recommendation system that suggests learners' relevant materials to the users based on users' preference with high accuracy.
    a. Suggest Books using content based filtering.
    b. Suggest Journals using collaborative filtering.
2. Analyse the data for recommendation
3. Create a Learner's information search to restrict learners for learning content

Web based recommendation system for learners using Natural Language Processing is a web application that focuses particularly on learners and suggests materials based on learners' context. The materials can include recommendations of books, Journals, learning videos, etc. This project aims to build a recommendation system that comprises the content based filtering algorithm, collaborative filtering algorithm and also the programmable search engine.[1]

This project is divided into four modules that is book recommendation module which recommends user relevant books in context to similar authors, similar interests and top rated books, Journal recommendation Module using collaborative filtering which recommends user relevant Journals in context to Similar Users and Similar Interests, Visualization Module which Analysis of the dataset in the form of Graphs and figures, Learner's Search which returns web pages related to learning context, Returns Images related to learning context.

A new area of research that is flourishing and in high demand is recommender systems (RS). Major developments in Internet technology and e-commerce have also contributed to the increased interest in this study issue. In e-commerce, RS provides a lot of benefits [2]. There are three ways in RS which may enhance the ecommerce marketplace. i.e., by assisting customers who have never shopped online, cross-selling products, and enhancing client loyalty. When Amazon succeeded in

boosting sales by implementing collaborative filtering (CF) techniques in the late 1990s, RS research experienced an enormous surge. As word spread about Amazon's popularity, other internet companies started incorporating RS on their websites. The CF technique is protected by a US patent owned by Amazon. Since finding priority information and removing material that people dislike is the primary goal of RS, it may be said that RS is a subset of information filtering. Following the investigation of customer preferences using historical data, machine learning algorithms are used to rate the recommended products that consumers like.[3]

Since the invention of computers, the concept of employing computers to present people with the finest content has been around. In 1979, the RS concept was first used in a system named Grundy, which is the computer library that lets users read books. This trend continued when the first commercial RS Tapestry was introduced in the early 1990s. The GroupLens Lab at the University of Minnesota, USA, released another version of RS in the early 1990s that helps individuals locate what they enjoy. It was termed the GroupLens system. The ideology of the system is said to be comparable to that of the Tapestry, Ringo, BellCore, and Jester systems. The introduction of one of the most popular RS features in the late 1990s was a further improvement of the system. The advent of one of the most prominent RS technologies, Amazon Collaborative Filtering, was a later evolution of RS in the late 1990s. RS based on Joint Filtering has grown highly widespread since this time and has been applied in a variety of e-commerce and online platforms. RS tool boxes have also been created in large numbers. Amazon's success has spawned a slew of RS algorithms that incorporate many strategies, dubbed hybrid approaches.[4]

The industry spent extensively on RS research after a successful spell in the late 1990s. Netflix, an Internet streaming media service, conducted the most popular RS contest. They began the 2006 Netflix Awards by presenting a prize of $1 million to the winner of a competition for Best RS Film. The winning squad was declared in 2009. YouTube first made RS available on its platform in 2010.[5]

Recommendation system is a process of suggesting contents to the user. This comes under an information filtering system which analyzes how a user would rate or prefer

a material. To address this in simple words, it is a kind of algorithm that uses data from the user's previous behaviors and does a similarity analysis to propose relevant products to them. Recommendation Systems are used to forecast user preferences on commercial and non-profit websites.

Analysis of user data and extraction of important information for further predictions are the fundamental functions of recommendation systems. Learners can make use of the recommendation to choose materials from a universe of options or to interact with other users who share their interests, as the recommendation system also detects others who share their interests.

Recommender systems are designed to recommend circumstances to the user on the basis of a variety of factors. The recommendation systems predict the probable product that users are interested in or most likely to buy.

Typically this system relates to a sizable volume of data in the form of information present by filtering information structured based on the information/data provided by the user. It also includes other factors that give importance to the user's preference and interest. It finds out the pattern between the user and his interest and computes the similarities between them for recommendation.

There are many things that could be recommended. This may include content such as movies, books, reports, articles, jobs, and so on. Netflix uses the recommender system that is capable of recommending movies in addition to web series to all people. Similarly YouTube uses recommendation algorithms that suggest different videos which match the interests of the user. In the same way there are a wide variety of examples and uses of recommender systems that are widely used in this modern era.[6]

1.2 Dissertation Organization:

The entire report is divided into nine fragments and the Chapter l contains the Description of the project and its Introduction , Chapter 2 consists of literature Survey and Tools and technologies used in the project , Chapter 3 contains Software requirement specifications, Chapter 4 contains System Designs ,Chapter 5 includes Detailed designing of the project then the Chapter 6 consists of the Implementation of the project, Chapter 7 has Testing of the project, Chapter 8 comprises of Conclusion lastly in Chapter 9 The Future Enhancements of the undertaking is referenced.

Chapter 2 - Literature Review

The purpose of this chapter is to understand the existing research paper and gain knowledge about the recommendation module

2.1 Literature Survey:

"Shun-Feng Su, Xi-Jun Chen presents - Recommender Systems in E-learning" [7] Targets on some specific attributes which helps in making e-learning recommendation system different from other recommendation systems he also discuss the complexity of recommendation system and focuses on challenges and future direction on e-Learning recommendation system.

"Mojtaba Salehi, Isa Nakhai Kmalabadi presents - A Hybrid Attribute-based Recommender System for E-learning Material Recommendation"[8] Concentrates on specific algorithms which can take over the traditional recommendation systems by considering precision, recall and FI scores and also discusses which attributes or more suited for e-Learning platform based on this consider the algorithm which can satisfy learners real learning choice accurately and with updated data in it.

"Salam Fraihat, Qusai Shambour presents - A Framework of Semantic Recommender System for e-Learning" [9] speaks about the uses and advantage of semantic recommender which helps in area of e-learning recommendation for individuals that is in providing important advertises to assist the learners helping them to find the relevant Los in the field where learner is interested in. The author also presents the formula for semantic recommendation which utilizes intra and extra semantic attributes between LOs.

"Zhendong Niu, Shanshan Wan presents - A Hybrid E-learning Recommendation Approach Based on Learners' Influence Propagation" [10] Concentrates on an influence dependent filtering model that is independent of rating the data as information. IFL is applied to the recommendation model to make them optimize which will help in presenting a more accurate and adoptive learner model. The author presents the model which will cluster the similar learners as one group and display the recommendation based on the similarity.

"Karim Dahdouh, Ahmed Dakkak, Lahcen Oughdir, and Abdelali Ibriz presents Large-scale e-learning recommender system based on Spark and Hadoop" [11] Targets creating an allocated recommender system of learning courses for helping students in taking more optimal learning resources. This also recommends a list of desirable classes for college students according to their interest attributes to improve learner satisfaction and performance by showing them individualized and adapted advice and academic resources.

"Pradnya V. Kulkarni, Dr. Sunil Rai, and Dr.Rohini Kale presents - Recommender System in eLearning"[12] Offers a study on an overview of recommendation systems and the specific field of eLearning Suggestion System. Emphasis was given to the prominent approaches applied in this area until now.

"Hayder Murad, Linda Yang - Personalized E-Learning Recommender System using Multimedia Data" [13] recommend a framework that supports the enhancement of an individualized eLearning recommender system incorporating the techniques regarding student profiling, expertise estimation, assessment, and even feedback as a way to increase students' learning method by recommending the web video learning elements based on scholar profile and expertise level

"Reema Sikka, Amita Dhankhar, Chaavi Rana - A Survey Paper on E-Learning Recommender System" [14] recommend an methodology that helps in building a software bot that takes into consideration data mining associated with it. He also proposes a model which helps in understanding users behavior. These kinds of recommendation models can help learners to find their way to the web materials by finding relevant resources faster.

"Oana Maria Teodorescu presented - Building Test Recommender Systems for e-Learning Systems" [15] Gives a personal set of questions depending on both their current status and the status of these colleagues who have previously consumed tests. Visual stats of the trial and error results in words of knowledge coverage of the strategy map show appealing initial results.

"John Taurus, Zhendong Niu, Bhakti Khadidja presents - E-Learning Recommender System Based on Collaborative Filtering and Ontology" [16] offer a recommendation approach combining CF and ontology for suggesting individualized learning materials to online students taking into bank account the learner characteristics. In our method, ontology can be used to include learner characteristics in the recommendation process. Fresh results show that our proposed ontology-based suggestion approach outperforms the VOIR algorithm on its own.

Itinerary recommendation systems Madhu & Manjula[17] proposed an automatic location recommendation system, enhanced in this work with the addition of itinerary recommendation. Choudhury et al. (2010) discussed Using a POI chart made from a user's picture feed, automatically designing an itinerary. However, it only offers a rough answer, coupled with a longer execution time and requiring more user intervention

Dunstall et al.[18] proposed a technique for automatically scheduling vacation travel which recommends the travel package for customers based on his previous trips . This approach is purely commercial and calls for more user involvement and processing time.

Activity/event recommendation systems Zheng et al. [19] proposed a recommendation system for recommending locations and activities. The system uses a CF-based technique for recommendation and maintains three matrices: location-activity, location feature and activity-activity matrix. When a user logs in for an activity, data maintained in the location-activity matrix provides information to the effect that the user has just been associated with an activity. The location-activity matrix displays the association between locations and categories of points of interest. Similar locations will have similar, possible activities.

Zheng et al. [20] proposed a user recommendation system which identifies expert users by deploying the HITS algorithm. The algorithm is applied over a hierarchical graph, built using users historical trajectories. Friends can also be recommended to users, based on this method, by 20 following the links. The node or person

connected with the most number of links will be the expert or celebrity, depending on the context

User recommendation systems Ying etal.[21] proposed a friend recommendation system by following a systematic approach. Users' travel routes are converted to a sequence of locations and a mining algorithm used to discover patterns in the routes. Similarities between patterns are identified and friends recommended, based on the similarities identified.

Sandholm & Ung proposed[22] a social media recommendation system for online web content. It is built on a CF-based method which considers geographical influences on ratings. Social media recommendation systems A social media recommendation system recommends media in social networks or the internet like online news, Twitter pages, online videos etc. to users.

By reviewing the above journals following observations were made:

- Lack of Data is the main issue in recommendation system
- Cold beginnings and data analysis are issues for recommendation systems.
- Contextual information in real time can accurately match students' real-time learning demands.
- The semantic representation algorithm is based on search term extension. Modeling student interactions to group optimal learning suggestions has been done using self-organization theory.
- In the learning referral platform, the primary principle of the referral mechanism is to assess the conduct and behaviour of prior students.

2.2 Existing and Proposed System:

The internet has evolved into a tangible aspect of the globalized world, serving as a key tool for breaking down barriers across physical differences and global boundaries. The incorporation of the internet into our daily lives has become almost unavoidable in this modern era, and a large number of people use it for a variety of purposes, including posting their opinions and reading others' opinions while also commenting on them, e-learning, e-banking, e-library, and e-commerce, among others. The amount of materials available on the internet is sufficient to meet the different educational and research needs of the public on a regular basis.[23]

The path to online learning merely requires understanding the best-way to explore and grow rather than reinventing the wheel for previously studied aspects of knowledge. These two phenomena - Big Data and The Internet of Things - are intricately intertwined in a big and complex ecosystem in the future of the internet, whose contours are progressively becoming obvious. Data is collected in this system based on the web pages people visit, our social media activity, our smartphones, and the many sensors in the actual environment.[24]

A Web based recommendation system for learners using NLP is a web application that focuses particularly on learners and suggests materials based on learner's context. The materials can be books, Journals, learning videos and so on. The main problem in this system to develop system particularly for learners and to suggests learners' relevant materials to the users based on users' preference with high accuracy and other problem includes Lack of Data, Changing Data, Changing User Preferences, Complexity

Recommender Systems[25] are software tools and strategies that recommend content that a user might find useful. It assists the learner's in making a decision
There are three types of recommender systems in general:

- A collaborative recommender system is one that results in recommendations which are based on the ratings of users with comparable tastes in the past.
- A content based filtering recommender system is one that produces results based on the content similarity of documents or things.

- A knowledge-based recommender system generates its results using supplementary and means–end knowledge.

The details of the proposed system are given below. The project is built using the following four modules.

1. Book Recommendation System(Using content based filtering):

A book recommendation system that gives readers a list of books to read depending on their preferences. The Book Recommendation system uses content based filtering which recommends books based on content like similar authors, similar publications and similar interests.

The objective of this module is to recommend books according to the user's interest. To achieve the objectives following steps are performed:

#1. Data Cleaning and Preprocessing: The dataset consists of three tables; Books, Users, and Ratings.

For Books Table:

- Drop all three Image URL features.
- Check for the number of null values in each column. There are only 3 null values in the table. Replace these three empty cells with 'Other'.
- Check for the unique years of publications. Two values in the year column are publishers. Also, for three tuples the name of the author of the book was merged with the title of the book. Manually set the values for these three above obtained tuples for each of their features using the ISBN of the book.
- Convert the type of the years of publications feature to the integer.
- By keeping the range of valid years as less than 2022 and not 0, replace all invalid years with the mode of the publications that is 2002.
- Upper-casing all the alphabets present in the ISBN column and removal of duplicate rows from the table.

For Users Table:

- Check for null values in the table. The Age column has more than 1 lakh null values.
- Check for unique values present in the Age column. There are many invalid ages present like 0 or 244.
- By keeping the valid age range of readers as 10 to 80, replace null values and invalid ages in the Age column with the mean of valid ages.
- The location column has 3 values: city, state, and country. These are split into 3 different columns named; City, State, and Country respectively. In the case of null value, 'other' has been assigned as the entity value.
- Removal of duplicate entries from the table.

For Ratings Table:

- Check for null values in the table.
- Check for the Rating column and User-ID column to be an integer.
- Removal of punctuation from ISBN column values and if that resulting ISBN is available in the book dataset only then consider dropping that entity.
- Upper-casing all the alphabets present in the ISBN column.
- Removal of duplicate entries from the table.

#2. Algorithms Implemented:

Popularity Based Recommendation : This is further subdivided into popular in whole collection, popular at a given place, Similar author and publications

Popular in the Whole Collection

The dataset is sorted according to the total ratings each of the books have received in non-increasing order and then recommended top n books.

Popular at a Given Place

The dataset was filtered according to a given place (city, state, or country) and then sorted according to total ratings they have received by the users in decreasing order of that place and recommended top n books.

Books By the Same Author, Publisher of Given Book Name
For this model, models have sorted the books by rating for the same author and same publisher of the given book and recommended top n books.

Popular Books Yearly
This is the most basic model in which models have grouped all the books published in the same year and recommended the top-rated book yearly.

2.2 Content Based Recommendation: By comparing similarities in book titles, this system suggests books to users. For this, TF-IDF feature vectors for unigrams and bigrams of book titles were made; only those books' data that had at least 80 ratings were taken into consideration.

2. Journal Recommendation System(Using collaborative filtering):
A Journal recommendation system gives learner’s a list of articles to read depending on their preferences. The journal recommendation uses collaborative filtering which takes into account similar users. This algorithm clusters the similar users and groups them in one unit and suggests the journals if a user belongs to that cluster.

1. Rank Based Recommendations: Find the most popular articles simply based on the most interactions. Since there are no ratings for any of the articles, it is easy to assume the articles with the most interactions are the most popular. These are then the articles models might recommend to new users (or anyone depending on what users know about them).

2. User-User Based Collaborative Filtering: Look at users that are similar in terms of the materials they have interacted with. These materials could then be recommended to similar users and it's a step towards more personal recommendations.

3. Visualization Module:
This module helps in analysing the dataset. This model enables developers to more easily and quickly comprehend vast amounts of information. It enables a more in-depth understanding of the information's impact on the recommendation system

for learners and visually conveys that understanding to both internal and external audiences.

4. Learner's Search:

This module helps in restriction based search for learning context. After adding restrictions to the module, when the Learner gives the input to learners search results are displayed only from learning contents when the learner searches for any other content other than educational content no Results will be displayed because input will not belong to learning content.

2.3 Tools and Technologies used

The project involved the utilization of several open source tools for the successful development of the application. Software once written requires a framework or platform inorder to run properly. Some of the platforms, technologies and the tools that were used in the development of the project as follows:

Jupyter Notebook:

The most recent web-based interactive development environment for code, data, and notebooks is JupyterLab. Users can configure and arrange workflows in data science, scientific computing, computational journalism, and machine learning using the interface's flexibility. A modular structure encourages expansions to increase and improve functionality. The first web application for producing and sharing computational documents was called a Jupyter Notebook. It provides a straightforward, efficient, document-focused experience. To "create open-source software, open standards, and services for interactive computing across dozens of programming languages," Project Jupyter is a project and community. Fernando Pérez and Brian Granger separated it from IPython in 2014. Project Jupiter is a tribute to Galileo's notebook, which documents the discovery of Jupyter's satellites, and it refers to the three primary programming languages Jupyter supports: Julia, Python, and R. Products like Jupyter Notebook, JupyterHub, and JupyterLab are developed and supported by Project Jupyter. NumFOCUS provides financial support for Jupyter.[17]

Programming in several languages is possible using Jupyter Notebook's connections to numerous kernels. The Jupyter kernel is a software application that responds to various requests (code execution, code completion, and inspection). The kernel can be located on the same machine or on a different one because it communicates with other Jupyter components via ZeroMQ. The Jupyter kernel can connect to multiple clients at once and is unaware that it is tied to a specific document, unlike many other notebook-like interfaces. With a few exceptions, the kernel typically can only support one language. [Reference required] The IPython kernel is preinstalled on the Jupyter Notebook by default. There are 49 Jupyter compatible kernels for numerous programming languages, including Python, as of version 2.3 (October 2014).[18]

The nbconvert library [19] or "jupyter" allows users to "name and download" Jupyter notebooks in a web interface in a variety of open output formats, including HTML, presentation slides, LaTeX, PDF, and ReStructuredText. users may export to Python and Markdown. The shell's "command line interface" for nbconvert. The nbconvert library [20] is offered as a service via NbViewer [21] to make it easier to visualise Jupyter notebook documents online. can display and transform HTML.

Version 0.12 [18] (December 2011) saw the addition of the Notebook interface to IPython, which in 2015 changed its name to Jupyter Notebook (IPython 4.0 is Jupyter 1.0). The notebook interface of other programmes like Maple, Mathematica, and SageMath is comparable to that of Jupyter Notebook. A computer interface design known as SageMath first appeared in Mathematica in the 1980s. [17] In the beginning of 2018, Jupyter was more popular than the Mathematica notebook interface. [18]

Project Jupyter now has a new user interface called JupyterLab. It offers the elements of a conventional Jupyter notebook with a configurable user interface (notebook, terminal, text editor, file browser, rich output, etc.). On February 20, 2018, the first stable release was revealed. [19]

In this project the Jupyter notebook is being used to run python3 code which is used to build the project.

Anvil Software:

Anvil is a Python-only development and hosting environment for full-stack web applications. Drag and drop the user interface, then create the frontend and backend Python code to make everything function. It's never been simpler to construct a website (or faster). Platform as a Service includes the technological stack tool Anvil. [20]

Anvil software is made up of:

- The user interface was created by the user using a drag and drop designer for the Anvil programme.
- Python client-side code that is executed in a web browser

- Server-side Running Python code on Anvil's server
- data table or embedded database for storing data
- users computer's Python programme is running and can communicate with users app. [21]

In this project Anvil software is used to create the interface so that learners can interact with the recommendation module.

Programmable Search Engine:

Users may design search engines for individual websites, blogs, or groups of websites using programmable search engines. Search engines may be set up such that both web pages and photos are searched. Users may alter search results' appearances, add their own promotions, and improve ranks. By linking their search engine to their Google AdSense account, users may earn money from their searches.

Programmable search engines may be used in two different scenarios. Users can build a search engine that simply looks for material on one particular website (a site search) or one that looks for information on a certain subject across several websites. Programmable search engines can be instructed by users to search, prioritise, or ignore specific websites based on their subject-matter knowledge. Users may customize their search engine to their preferences because they are familiar with their users. [21]

user can perform following:

- Create a search engine that may be programmed to explore a certain set of websites and pages
- Enable website image searches
- Customize users search results' appearance and feel, and enable autocomplete as users write.
- Improve search results using promotions
- Use structured data from users website to personalise the search results users receive.
- So that users may profit from ad clicks on the search results page, connect users search engine to users Google AdSense account.[23]

In this project Programmable search engine is used to create restriction based learners search.

Python 3:

Python is a sophisticated, widely used interpreted programming language. Python's design ethos places a high value on code readability, as seen by the extensive usage of whitespace. Its language structure and object-oriented methodology are designed to aid programmers in creating logical and clear code for both large-scale and small-scale applications. Python has garbage collection and dynamic typing. One of the supported programming paradigms is functional programming, along with structured (particularly procedural) and object-oriented programming. Due to its broad standards, Python is sometimes referred to as a "battery-containing" language. Library.[24]

In this project Python3 is used to create a machine learning recommendation module.

Libraries Used:

NumPy 1.19.2: One of the most often used Python apps for scientific computing is Numpy. Along with versions made up of mask and matrices, it also provides a multidimensional array object that may be utilised for a variety of mathematical functions.[25]

Pandas 1.1.5: The most often used open source Python library for data science, data analysis, and machine learning activities is called Pandas. It is constructed on top of Numpy, a different package that supports multi-dimensional arrays.[26]

Matplotlib 3.3.2: For Python and its numerical extension NumPy, Matplotlib is a cross-platform data visualisation and graphical charting package. As a result, it presents a strong open source substitute for MATLAB. The APIs (Application Programming Interfaces) for Matplotlib allow programmers to include charts into GUI applications.[27]

Sklearn 0.14.5: A crucial package for the Python programming language that is frequently used in machine learning applications is called Scikit-learn. The main focus of Scikit-learn is on machine learning tools, such as the general-purpose, statistical, and mathematical algorithms that serve as the foundation for many machine learning technologies.[28]

Seaborn: An open-source Python library based on matplotlib is called Seaborn. It is utilised for data exploration and data visualisation. With dataframes and the Pandas library, Seaborn functions with ease. The generated graphs are also easily customizable.[29]

Scipy: It's an open-source Python library which is employed to address issues in science and mathematics. It is based on the NumPy extension and offers a large variety of high-level functions for manipulating and visualising data.[30]

2.4 Hardware and Software Requirements

The hardware and software requirements for developing the system are outlined below -

Table 2.4(a) Hardware Requirements

Hardware Requirements		
Particulars	**Minimum**	**Recommended**
Processor	Dual-core 1.6GHz	Octa-core 2.01GHz
Storage	2GB	8 GB
RAM size	2 GB	8 GB
Installed Memory	4 GB	8 GB

Table 2.4(b) Software Requirements

Software Requirements	
Compatible Operating system	Ubuntu 16.08 or higher, Windows 7 or higher, Mac OS
Language	Python3 or higher, Html5, CSS3
Text Editor and Framework	Jupyter Notebook 6.1.6
UI Tool	Anvil software
Libraries	NumPy 1.19.2, Pandas 1.1.5, Matplotlib 3.3.2, sklearn, seaborn, matplotlib, scipy.

DATASET REQUIREMENTS:

The dataset for book recommendation is taken from IBM book repository. It has 3 tables Books, Users, Ratings

Users Table: It consists of data of the users. It consists of UserId, Location, Age. UserId is Integer. Location is a string and Age is integer. These fields contain Integer, String or NULL-values.[31]

Books Table:

By using the appropriate ISBN, books are recognised. The dataset has previously been cleared of any invalid ISBNs. Additionally, certain content-based data from IBM is provided, including "Book-Title," "Book-Author," "Year of Publication," and "Publisher." Keep in mind that just the first author is shown when there are many writers. Additionally, three alternative flavours of URLs ('Image-URL-S', 'Image-URL-M', and 'Image-URL-L'), which stand for small, medium, and large, are provided that connect to cover pictures.[32]

Ratings Table:

Contains the details of the book's rating. Ratings (or "Book-Ratings") can be represented explicitly by using a scale of 1 to 10 (with higher numbers signifying better acclaim), or implicitly by using the number 0.[33]

Chapter 3: Software Requirement Specification

This chapter outlines the set of conditions necessary for the project's overall operation.

3.1 Introduction

The Software Requirement Specifications shows the purpose and background for the proposed work. The following sections define what the work serves to achieve and how it is executed. This software requirements document specification gives users all the details users need to know about the A Web Based Recommendation for Learner's Using NLP. This application was created to give assistance to learners from recommendation mechanics. This part looks into the problem's definition, as well as the goal and scope of this application.[34]

Table 3.1 Acronyms and Abbreviations

Acronym	Abbreviation
API	Application Programming Interface
NLP	Natural Language Programming
UI	User Interface
HTTP	HyperText Transfer Protocol
XML	Extensible Markup Language
JDBC	Java Database Connectivity
IDEs	Integrated Development Environment
HTML	HyperText Markup Language

CSS	Cascading Style Sheets
URL	Uniform Resource Locator.
TF-IDF	term frequency-inverse document frequency
ISBN	International Standard Book Number
OOD	object-oriented design

3.2 General description:

This section gives a brief overview of the project, which includes the product description, product functions, system user characteristics, general constraints, and assumptions.

3.2.1 Product Description:

These days everyone is constantly being recommended products that either want to grab our attention and retain our attention or to purchase that particular product. Anything that sort of magically pops up in our feed is being fed by a recommendation system. So, whether it be users tube videos or instagram, post facebook, posts, news, articles,amazon, prime deals, netflix videos and so many other products that exist out there. Everything is being fed into this recommendation system to attract our attention. So why would users even care about recommendation algorithms? Well, for one thing, if users have the most powerful recommendation algorithm out there then that particular algorithm is worth potentially billions of dollars, because users can overtake companies whose entire revenue stream or a large chunk of the revenue stream comes from recommendations and attraction of the viewer base. users can take a look at tick tock, for instance, their main driver behind all of their success is their recommendation. Algorithms and users all know that's incredibly powerful and then made them into a multi dollar corporation. So Recommendations is a booming technique in every sector and when it comes to the educational domain the recommendation helps in personalized results for learners.

3.2.2 Learner's Recommendation Functions:

- Web based recommendation for learners results in returning top recommendations for learner
- Get recommendation based on the learners Content
- Get personalized results
- Get web results based only from educational sites
- Get recommendations based on UserId
- It results in giving personalized results for users using userId
- BookRec uses content based filtering to return top recommendation
- JournalRec uses collaborative filtering to return rop recommendations
- Anvil is used for easy creation of user interface

learners search engine results in displaying web pages for learners which uses NLP thus recommendation for learners can be used by learners for quick and easy personalized searching

User Characteristics :
The system will be used by learners who want to save time in searching for learning based information. The user must be familiar with the internet and browsers. The learners can easily use the system if he or she is familiar with Python programming and its environment, as well as Jupyter.

3.2.2 Assumptions and Dependencies:

The assumption users can consider while building this application is that the learners who will use this web application are able to use this in laptop and desktop, browse websites with specific URLs and understand how the application works at a business level.

3.3 Functional Requirements:

This section provides requirements of this application. There are four main functional modules:Book Recommendation Module, Journal recommendation Module using collaborative filtering, Visualization Module, Learner's Search

Module Name : Book Recommendation System(using content based filtering)

Purpose : Recommendation of books to the learners

Input : Text(title of the book), Number(number of books for recommendation)

Function: Perform Content based filtering recommends books based on content like similar authors, similar publications and similar interests. By comparing similarities in book titles, this system suggests books to users. For this, TF-IDF feature vectors for unigrams and bigrams of book titles were made; only those books' data that had at least 80 ratings were taken into consideration.

Output : List of Recommended books

Module Name :Journal recommendation System(using collaborative filtering)

Purpose : Recommendation of Journals to the learners

Input : Text(name of the journal), Number(number of books for recommendation)

Function: Performs collaborative filtering which uses User id for the recommendation and performs similarity recommendation. Look at users that are similar in terms of the materials they have interacted with. These materials could then be recommended to similar users and it's a step towards more personal recommendations.

Output : Recommendations of the journal

Module Name :Visualization Module

Purpose : Analysis and Visualization of Recommendation data

Input : Preprocessed Data in .csv format

Function: Creation of Graphs and figures using Matplotlib. This model enables developers to more easily and quickly comprehend vast amounts of information. It enables a more in-depth understanding of the information's impact on the recommendation system for learners and visually conveys that understanding to both internal and external audiences.

Output : Graphs of visualized data

Module Name : LearnerSearch

Purpose : To perform information search for learners content

Input : Text

Function: Performs Programmable search engine which returns results only on particular websites restricted to the learners. This module helps in restriction based search for learning context. After adding restrictions to the module, when the Learner gives the input to learners search results are displayed only from learning contents when the learner searches for any other content other than educational content no Results will be displayed because input will not belong to learning content.

Output : List of web search pages for learner

3.4 Non-Functional Requirements:

- Reliability: Reliability describes how probably its miles for the software program to paintings without failure for a given time frame. Reliability decreases because of insects in the code that may rely on the percentage of operations which are finished efficiently.
- Response Time: It refers to the total amount of time taken to respond to a request for a service. In this application, when the user requests for a recommendation it notify the model within a fraction of seconds.
- Performance: The Programmable search API should respond in less than 5 seconds, though it may take a few seconds longer if there are a large number of responses to be fetched.
- Correctness: The API should complete its task while delivering accurate results.
- Robustness: When handling exceptions, the code should not break. It should be able to deal with errors that occur during execution. If any exceptions occur in the backend code, it will take 10 to 40 seconds to restart the server.
- Security: The Anvil software provides uplink keys, which ensures that the anvil is well protected.

User requirements:

1. Gathering and arranging data about learners and materials
2. This is an important action. users must be aware of users users' identities and technologies. In this project it is Learners who interact with the recommendation module
3. Compare all users
4. Comparing all the users and grouping them to the similar likelihood so the recommendation can be based on similar users.
5. Ranking and recommending
6. Ranking as a product by a specific parameter as Top recommendation and recommending to other users

3.4 Design Constraints of Learners Recommendation:

3.4.1 Standard Compliance

- The internet connection should be stable
- Anvil software server must be constantly connected
- Jupyter server must be constantly connected
- All packages necessary for the projected must be previously installed

CHAPTER 4 - SYSTEM DESIGN

The system architecture, including internal connections, internal processes, and full system component concepts, are outlined in the system design.

4.1 System Perspective

This section gives an overview of the block diagram of the system

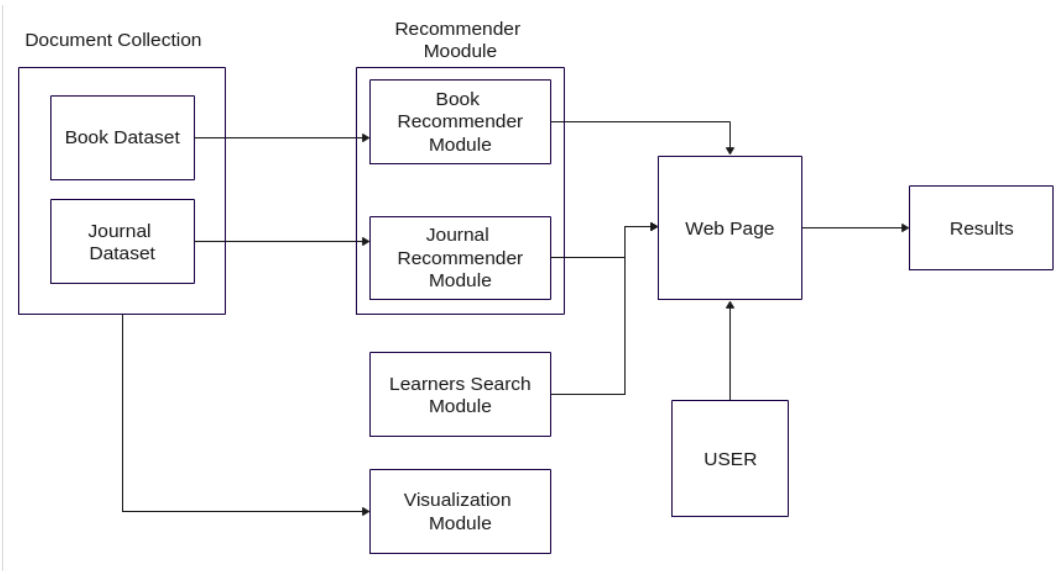


Figure 4.1 Block diagram of Learners Recommendation System

(i) Document collection:

The data set is used from IBM book data set and article dataset. The book data set consists of user ID book ID author name Publication name ratings of the book number of people reading the book age group of the people reading the book. The data set consists of a user rating table in this dataset it consists of user id and ratings. User ID is the primary key of rating data set it and user ID is the foreign key of book data set these two tables can be linked using user ID.

- Users data set consist of user ID location and age group. In the users dataset user ID is the primary key.
- Book dataset consists of ISBN Book title year Publication Publishers.
- Ratings table consists of user ID ISBN book ratings.
- The tables can be linked using primary key and foreign key users table and book table can be linked using user ID ratings table can be linked using ISBN

Data preprocess techniques like missing value detection can be removed or replaced by average values.

Necessary data preprocessing techniques are performed and clean data is fed to the recommendation module.

(ii) Recommender Module:

After the dataset is preprocessed the next step is building a recommendation model. In this project Content based recommendation filtering and collaborative filtering is used to build a book and Journal recommendation Module using collaborative filtering respectively. Recommender model is trained with document collection dataset which is connected to the Web page User will give input as text, keyword to recommendation model the recommendation model bt interacting in web page. Recommender model uses recommendation algorithms and suggests the relevant materials to the user in the form of text, web pages.

(iii)Web Page:

This is where the system and learners interact with each other. This web page is created using Anvil software. The user will give input to the system, the input will be fed to the module and output will be fetched.

(iv)Visualization Module:

This module helps in analysing the dataset. This model enables developers to more easily and quickly comprehend vast amounts of information. It enables a more in-depth understanding of the information's impact on the recommendation system for learners and visually conveys that understanding to both internal and external audiences. Analysis of the dataset in the form of Graphs and figures

(v) Learner's Search:

This module helps in restriction based search for learning context. After adding restrictions to the module, when the Learner gives the input to learners search results are displayed only from learning contents when the learner searches for any other content other than educational content no Results will be displayed because input will not belong to learning content.

(vi)User:

The User here is Learner who uses the recommender module. The user interacts with the web page by giving inputs and get the desired results

Table 4.1 Data Dictionary

Data name	Attribute Type
Book name	String
Number of Book	Number
UserID	String
Number of Journal	Number
Learners search box	String

Module Specification

1. Book recommendation System(using content based filtering):

The purpose of this module is to Recommend User the relevant books in context to the Similar Authors, Similar Interests and Top Rated Books. A book recommendation system gives readers a list of books to read depending on their preferences. The Book Recommendation system uses content based filtering which recommends books based on content like similar authors, similar publications and similar interests.Content Based Recommendation: By comparing similarities in book titles, this system suggests books to users. For this, TF-IDF feature vectors for unigrams and bigrams of book titles were made; only those books' data that had at least 80 ratings were taken into consideration. The main objective of this module is to recommend books according to the user's interest.

2. Journal recommendation System(using collaborative filtering):

A Journal recommendation system gives learner's a list of articles to read depending on their preferences. The journal recommendation uses collaborative filtering which

takes into account similar users. This algorithm looks at users that are similar in terms of the materials they have interacted with. These materials could then be recommended to similar users and it's a step towards more personal recommendations.

Recommends User relevant Journals in context to the following:

- Similar Users
- Similar Interests

3. Visualization Module:

This module helps in analysing the dataset. This model enables developers to more easily and quickly comprehend vast amounts of information. It enables a more in-depth understanding of the information's impact on the recommendation system for learners and visually conveys that understanding to both internal and external audiences. Analysis of the dataset in the form of Graphs and figures

4. Learner’s Search:

This module helps in restriction based search for learning context. After adding restrictions to the module, when the Learner gives the input to learners search results are displayed only from learning contents when the learner searches for any other content other than educational content no Results will be displayed because input will not belong to learning content.

- Returns web pages related to learning context
- Returns Images related to learning context

4.2 Context Diagram:

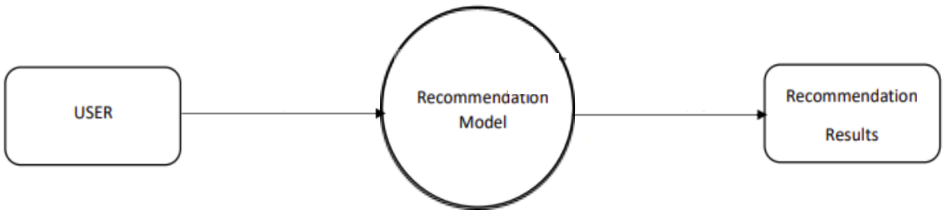


Figure 4.2: Context Diagram for Learners Recommendation

The User is the learner who uses the Recommendation Module. The flow of the Module is User gives input to the recommendation module. The recommendation module then processes the input and displays the recommendation results. Book recommendation module recommends user relevant books in context to similar authors, similar interests and top rated books, Journal recommendation Module using collaborative filtering which recommends user relevant Journals in context to Similar Users and Similar Interests, Visualization Module which Analysis of the dataset in the form of Graphs and figures, Learner's Search which returns web pages related to learning context, Returns Images related to learning context.

Chapter 5 - Detailed Design

5.1 System Design:

The object-oriented approach's main goal is to divide an information system's behaviour and structure into smaller modules that combine data and processes. Making system analysis and design more user-friendly while enhancing quality and efficiency is the fundamental aim of object-oriented design (OOD).[35]

Architecture Diagram:

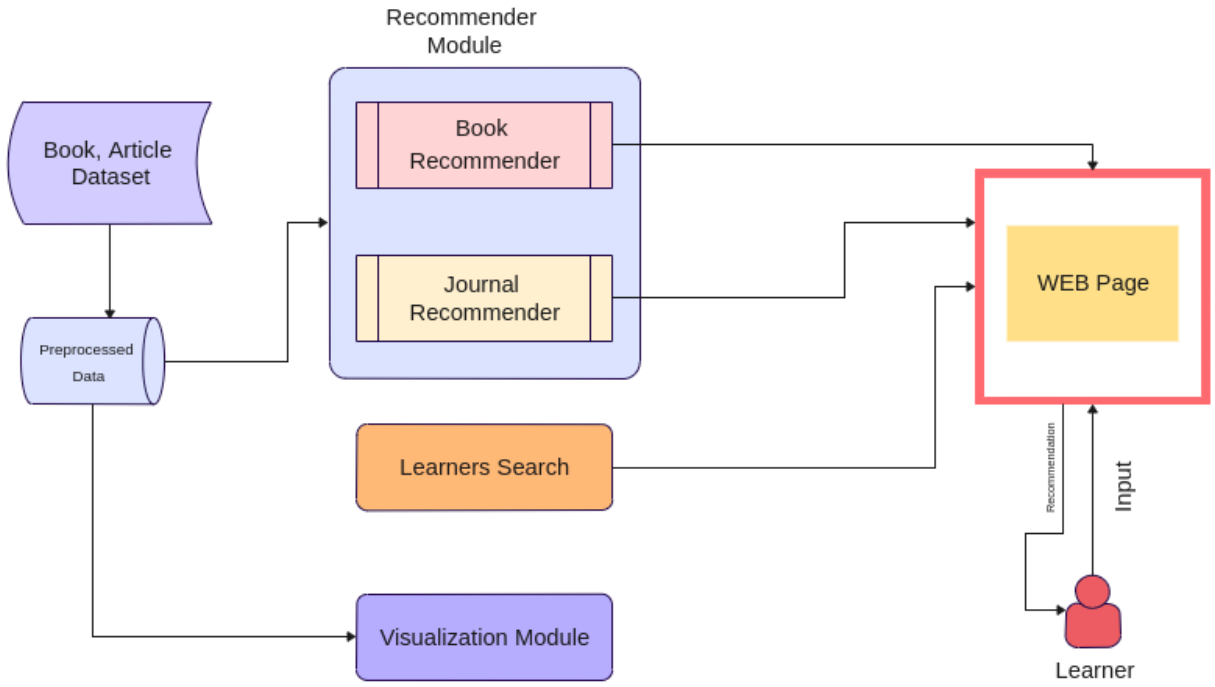


Figure 5.1 Architecture Diagram of learners Book Recommendation

The Figure 5.1 describes the architecture diagram of Learners Book Recommendation. The architecture begins with preprocessing the dataset, which will be stored in the recommendation dataset which will be used for the recommender module. The stored dataset will be fed to algorithms which will return the recommendation to the final learner.

Object Modeling:

Class Diagram : A static diagram is the Class Diagram. It supports an application's static watch. It displays a number of classes, interfaces, relationships, partnerships, and restraints. It is sometimes referred to as a structural map.[36]

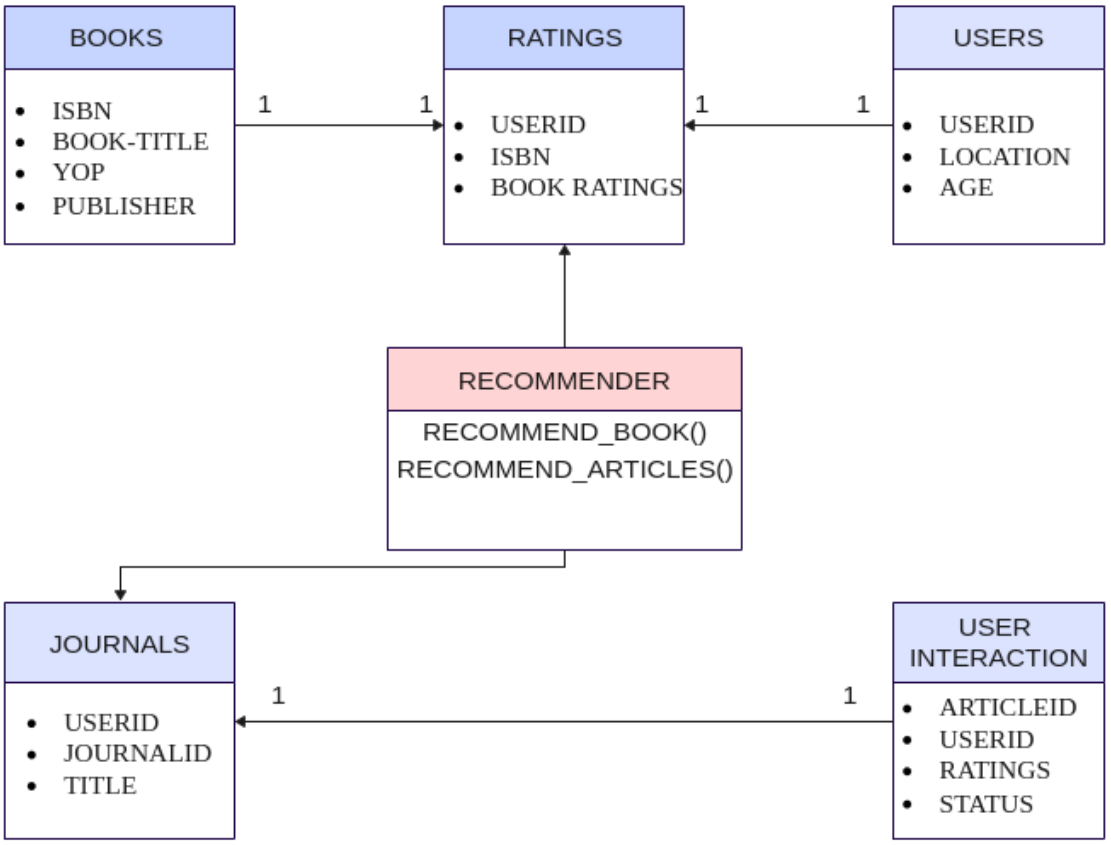


Figure 5.2 Class Diagram of Learners Recommendation

The above figure 5.2 shows the class diagram of the recommender system. There are three tables in the dataset: Books, Users, and Ratings. The data set consists of a user rating table in this dataset it consists of user id and ratings. User ID is the primary key of rating data set it and user ID is the foreign key of book data set these two tables can be linked using user ID. Users table data set consist of user ID location and age group. In the users dataset user ID is the primary key. Book dataset consists of ISBN Book title year Publication Publishers. Ratings table consists of user ID ISBN book ratings. The tables can be linked using primary key and foreign key users table and book table can be linked using user ID ratings table can be linked using

ISBN. Data preprocess techniques like missing value detection can be removed or replaced by average values.Necessary data preprocessing techniques are performed and clean data is fed to the recommendation module. Which is connected to the User Interface which is connected to the Web page User will give input as text, keyword to recommendation model the recommendation model by interacting in the web page. Recommender model uses recommendation algorithms and suggests the relevant materials to the user in the form of text, web pages.

Dynamic Modeling

Use Case Diagram:Use cases are used to determine scheme functionality during the analysis phase of a plan. Actors and use cases are divided up from the entity. Actors represent the roles that system users take on. Users can be other computers, people, or even different tool systems.[37]

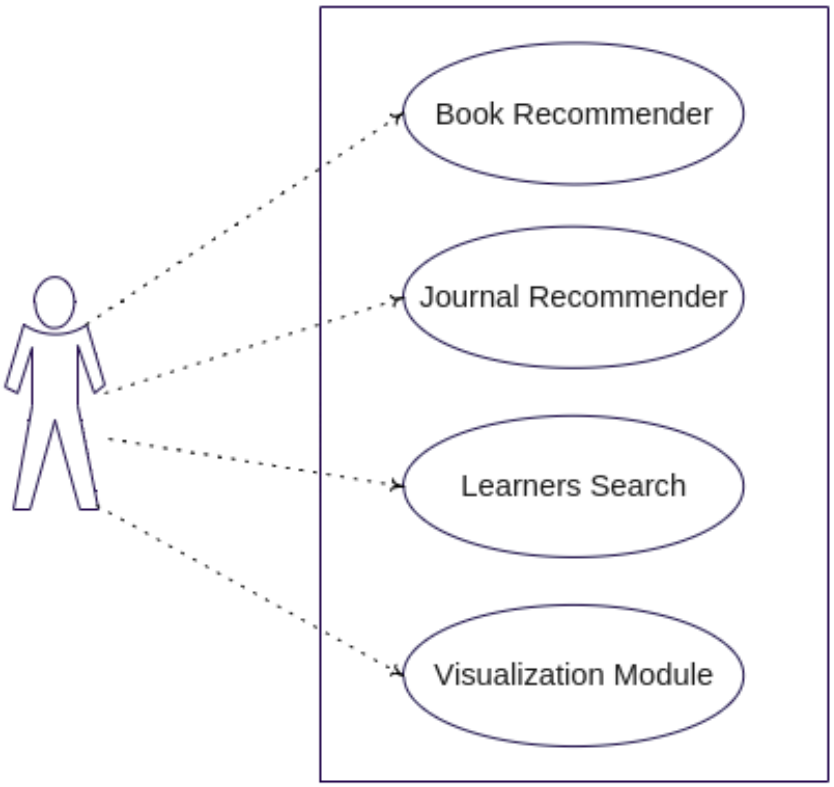


Figure 5.3 UseCase Diagram of Recommendation Module

The above figure 5.3 shows the use case diagram of the recommender system. Learners can get recommendation through Book recommendation module, Journal

recommendation Module using collaborative filtering, He can access Learners search for information search and Visualization module to visualize data..

Functional Modeling:

Data Flow Diagram: A data-flow diagram, or DFD, is a tool for providing statistics on a system or process. Databases on each entity's inputs and outputs as well as the process itself are also provided by the DFD. A data-flow map lacks control flow, loops, and many decision-making rules.[39]

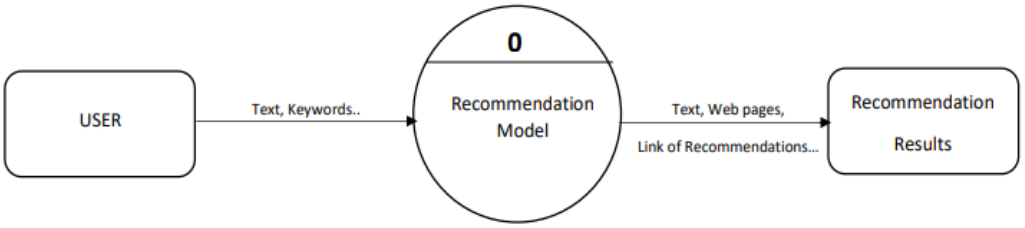


Figure 5.5: DFD Level 0 for Learners Recommendation

The above figure 5.5 shows the level 0 data flow diagram. The user provides input in the form of text or keywords which will be fed to the recommendation module which processes the input and provides the recommendation results.

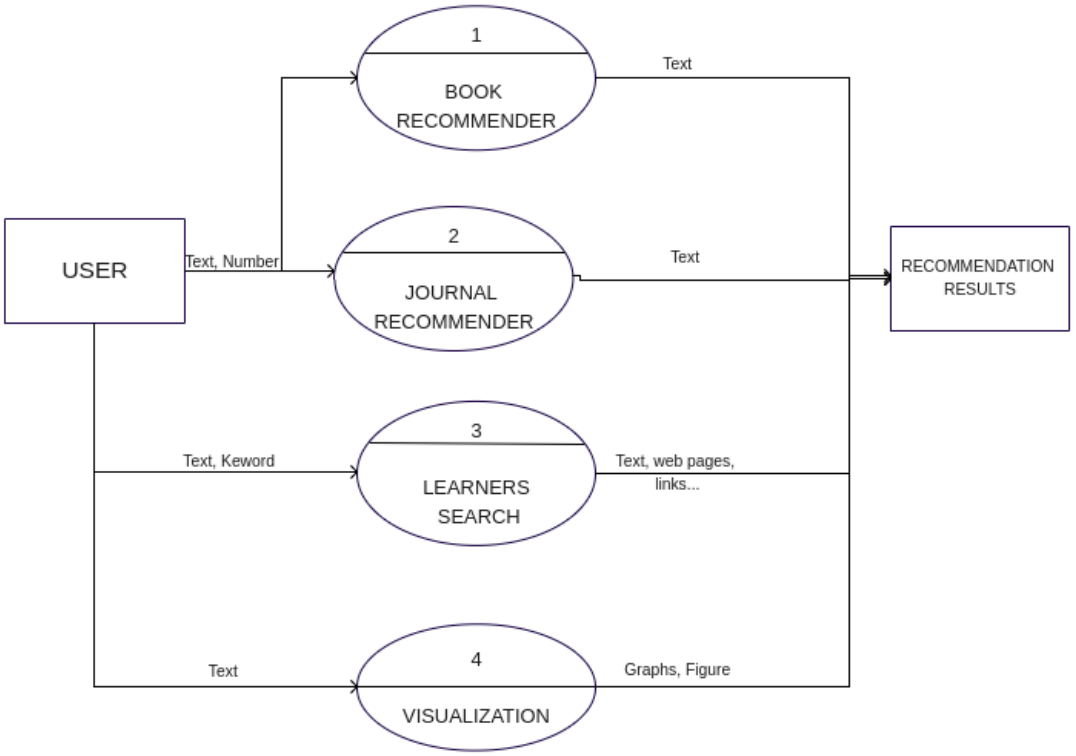


Figure 5.6: DFD level 1 for learners Recommendation

The above figure 5.6 shows the level 1 data flow diagram. The dataset will be cleaned using data preprocessing technique and fed to the recommendation module for training purposes. The user provides input in the form of text or keywords which will be fed to the recommendation module which processes the input and provides the recommendation results.

5.2 Detailed Design

The phase of detailed design is when the design is so well and plans and specifications are prepared.

PDL: Process definition language processes definition, both the graphics and the semantics of a workflow business process. It helps to gain knowledge on how the modules are implemented through algorithms and formulas in a system.

PDL for Learners Recommendation System:

Module1: Book Recommendation System(using content based filtering)

Input: Text(title of the book), Number(number of books for recommendation)

Steps followed for book recommendation module:

1. Perform Data preprocessing to eliminate the following
    a. Null value detection and recovery
    b. Removing of duplicate data
    c. Removing of data with 0 ratings

2. IF RATINGS > 80
        THEN SUGGEST='TRUE'
    SET data TO RECOMMENDED

3. Testing and Validation

Output: List of Recommended books

Module2: Journal recommendation System(using collaborative filtering)

Input : Text(name of the journal), Number(number of books for recommendation)

Performs collaborative filtering which uses User id for the recommendation and perform similarity recommendation

Steps followed for book recommendation module:

1. Data preprocessing.
    a. Null value detection and recovery
    b. Removing of duplicate data
    c. Removing of data with 0 ratings
2. IF USER_CLUSTER IN [LIST OF EXISTING USER]
        THEN SUGGEST=[SIMILAR USER LIST]
    ELSE
        SUGGEST=JOURNAL_RATING IN MAX(RATING)

3. Testing and Validation

Output : Recommendations of the journal

Module3 :Visualization Module

Input : Preprocessed Data in .csv format

Creation of Graphs and figures using Matplotlib. This model enables developers to more easily and quickly comprehend vast amounts of information. It enables a more in-depth understanding of the information's impact on the recommendation system for learners and visually conveys that understanding to both internal and external audiences. Following steps are performed in visualization module:

1. Creation of graphs and figures using matplotlib
2. IF RATING>80

   DISPLAY GRAPH AS TOP RECOMMENDER

   ELSE IF RATING ==0

   DISPLAY GRAPH AS IMPLICIT RATING
3. Analyzing the importance of the data.

Output : Graphs of visualized data

Module4: Learners Search

Input : Text

Performs Programmable search engine which returns results only on particular websites restricted to the learners

Following steps are performed for Learners Search:

1. Adding restriction to the module
2. Performing information filtering for displaying learning contents.

Output : List of web search pages for learner

Chapter 6 – Implementation

This chapter gives the overview of the important code snippets of the project

(i)Importing Modules Snippet:

```
IMPORT pickle
IMPORT operator
IMPORT numpy as np
IMPORT pandas as pd
IMPORT seaborn as sns
```

(ii)Loading Dataset Snippet:

```
SET books TO pd.read_csv(r"Datasets/Books.csv", delimiter=';',
error_bad_lines=False, encoding='ISO-8859-1', warn_bad_lines=False)

SET users TO pd.read_csv(r"Datasets/Users.csv", delimiter=';',
error_bad_lines=False, encoding='ISO-8859-1', warn_bad_lines=False)

SET ratings TO pd.read_csv(r"Datasets/Book-Ratings.csv", delimiter=';',
error_bad_lines=False, encoding='ISO-8859-1', warn_bad_lines=False)

OUTPUT("Books Data:    ", books.shape)
OUTPUT("Users Data:    ", users.shape)
OUTPUT("Books-ratings: ", ratings.shape)
```

(iii)Dataset Preprocessing Snippet:

```
OUTPUT("Columns: ", list(books.columns))
books.head()
SET books.at[187689 ,'Book-Author'] TO 'Other'
SET books.at[128890 ,'Publisher'] TO 'Other'
SET books.at[129037 ,'Publisher'] TO 'Other'
```

```
## Converting year of publication IN Numbers

    SET books['Year-Of-Publication'] TO
    books['Year-Of-Publication'].astype(int)
    OUTPUT(sorted(list(books['Year-Of-Publication'].unique())))

## Replacing Invalid years with max year

    SET count TO Counter(books['Year-Of-Publication'])
    [k FOR k, v IN count.materials() IF v EQUALS max(count.values())]
    SET books.loc[books['Year-Of-Publication'] > 2021, 'Year-Of-Publication']
    TO 2002
    SET books.loc[books['Year-Of-Publication'] EQUALS 0,
    'Year-Of-Publication'] TO 2002

## Check FOR all values present IN Age column

    OUTPUT(sorted(list(users['Age'].unique())))
    SET required TO users[users['Age'] <= 80]
    SET required TO required[required['Age'] >= 10]
    SET mean TO round(required['Age'].mean())

    FOR i IN range(0,len(list_)):
       IF list_[i][0] EQUALS ' ' or list_[i][0] EQUALS '' or list_[i][0]=='n/a' or
    list_[i][0] EQUALS ',': #removing invalid entries too
          city.append('other')
       ELSE:
          city.append(list_[i][0].lower())

## checking ISBN

    SET flag TO 0
    SET reg TO "[^A-Za-z0-9]"
```

```
FOR x IN ratings['ISBN']:
    SET z TO re.search(reg,x)
    IF z:
        SET flag TO 1
IF flag EQUALS 1:
    OUTPUT("False")
ELSE:
    OUTPUT("True")
```

(iv)Data Visualization Snippet:

```
SET publications TO {}
FOR year IN books['Year-Of-Publication']:
    IF str(year) not IN publications:
        SET publications[str(year)] TO 0
    publications[str(year)] +=1
SET publications TO {k:v FOR k, v IN sorted(publications.materials())}
SET fig TO plt.figure(figsize =(55, 15))
SET plt.bar(list(publications.keys()),list(publications.values()), color TO
'blue')
plt.ylabel("Number of books published")
plt.xlabel("Year of Publication")
plt.title("Number of books published yearly")
SET plt.margins(x TO 0)
plt.show()
```

(v)Book Recommendation Module Snippet:

1. Popularity Based (Top In whole collection)

```
DEFINE FUNCTION popularity_based(dataframe, n):
    IF n >= 1 and n <= len(dataframe):
```

```
        SET data TO
    pd.DataFrame(dataframe.groupby('ISBN')['Book-Rating'].count()).sort_value
    s('Book-Rating', ascending=False).head(n)
        SET result TO pd.merge(data, books, on='ISBN', left_index TO True)
        RETURN result
    RETURN "Invalid number of books entered!!"
OUTPUT("Top", number, "Popular books are: ")
popularity_based(dataset1, number)
```

2. Popularity Based (Top In a given place)

```
DEFINE FUNCTION search_unique_places(dataframe, place):
    SET place TO place.lower()
    IF place IN list(dataframe['City'].unique()):
        RETURN dataframe[dataframe['City'] EQUALS place]
    ELSEIF place IN list(dataframe['State'].unique()):
        RETURN dataframe[dataframe['State'] EQUALS place]
    ELSEIF place IN list(dataframe['Country'].unique()):
        RETURN dataframe[dataframe['Country'] EQUALS place]
    ELSE:
        RETURN "Invalid Entry"
SET place TO INPUT("Enter the name of place: ")
SET data TO search_unique_places(dataset1, place)
IF isinstance(data, pd.DataFrame):
    SET data TO popularity_based(data, number)
```

3. Books by same author, publisher of given book name

```
DEFINE FUNCTION OUTPUTBook(k, n):
    SET z TO k['Book-Title'].unique()
    FOR x IN range(len(z)):
        OUTPUT(z[x])
        IF x >= n-1:
            break
```

```
DEFINE FUNCTION get_books(dataframe, name, n):
    OUTPUT("\nBooks by same Author:\n")
    SET au TO dataframe['Book-Author'].unique()
    SET data TO dataset1[dataset1['Book-Title'] != name]
    IF au[0] IN list(dataframe['Book-Author'].unique()):
        SET k2 TO data[data['Book-Author'] EQUALS au[0]]
    SET k2 TO k2.sort_values(by=['Book-Rating'])
    OUTPUTBook(k2, n)
    OUTPUT("\n\nBooks by same Publisher:\n")
    SET au TO dataframe['Publisher'].unique()
    IF au[0] IN list(data['Publisher'].unique()):
        SET k2 TO pd.DataFrame(data[data['Publisher'] EQUALS au[0]])
    k2=k2.sort_values(by=['Book-Rating'])
    OUTPUTBook(k2, n)
IF bookName IN list(dataset1['Book-Title'].unique()):
   SET d TO dataset1[dataset1['Book-Title'] EQUALS bookName]
    get_books(d, bookName, number)
ELSE:
    OUTPUT("Invalid Book Name!!")
```

4. Books popular Yearly

```
SET data TO
pd.DataFrame(dataset1.groupby('ISBN')['Book-Rating'].count()).sort_values(
'Book-Rating', ascending=False)
```

(vi)Journal recommendation Module(using collaborative filtering) Snippet:

1.For Data Analysis

```
SET df_interactions_per_user TO df_interactions.groupby('email').count()
SET df_plot_linear TO df_interactions_per_user.drop(columns='title')
SET df_plot_linear_50 TO df_plot_linear[df_plot_linear['article_id'] < 50]
plt.figure(figsize=(15, 7), dpi=500)
```

```
# Fill IN the median and maximum number of user_article interactios below

    SET median_val TO int(df_interactions_per_user.median()[0])
    SET max_views_by_user TO df_interactions_per_user.max()[0]
    OUTPUT('50% of individuals interact with {} of articles or
    fewer.'.format(median_val))
    OUTPUT('The maximum number of user-article interactions by any user is
    {}.'.format(max_views_by_user))

Explore and remove duplicate articles from the df_content dataframe.
    # Find and explore duplicate articles
    # Remove any rows that have the same article_id - only keep the first

    OUTPUT(df_content.shape)
    SET df_content TO df_content.drop_duplicates(subset='article_id')
    OUTPUT(df_content.shape)
    SET unique_articles TO 714  # The number of unique articles that have at
    least one interaction
    SET total_articles TO 1051  # The number of unique articles on the IBM
    platform
    SET unique_users TO 5148  # The number of unique users
    SET user_article_interactions TO 45993  # The number of user-article
    interactions
```

(vii)Collaborative filtering model Snippet:

```
    SET df_interactions TO pd.read_csv('../data/user_item_interactions.csv')
    SET df_content TO pd.read_csv('../data/articles_community.csv')
    del df_interactions['Unnamed: 0']
    del df_content['Unnamed: 0']
    DEFINE FUNCTION sol_1_test(sol_1_dict):
       SET sol_1_dict_ TO NOT NULL
```

```
    IF sol_1_dict_ EQUALS sol_1_dict:
        OUTPUT("It looks like users have everything right here! Nice job!")
    ELSE:
        FOR k, v IN sol_1_dict.materials():
            IF sol_1_dict_[k] != sol_1_dict[k]:
                OUTPUT("String name".format(k))


DEFINE FUNCTION sol_2_test(top_articles):
    SET top_5 TO top_articles(5)
    SET top_10 TO top_articles(10)
    SET top_20 TO top_articles(20)
    SET checks TO ['top_5', 'top_10', 'top_20']
    FOR idx, file IN enumerate(checks):
        IF set(eval(file)) EQUALS set(pickle.load(open( "{}.p".format(file),
"rb" ))):
            OUTPUT("users {} looks like the solution list! Nice
job.".format(file))
        ELSE:
            OUTPUT("Oops! The {} list doesn't look how users expected. Try
again.".format(file))
```

6.2 Implementation:

This section gives the important screenshots of the project

(i) Implementation Screenshot For Book Recommender Module

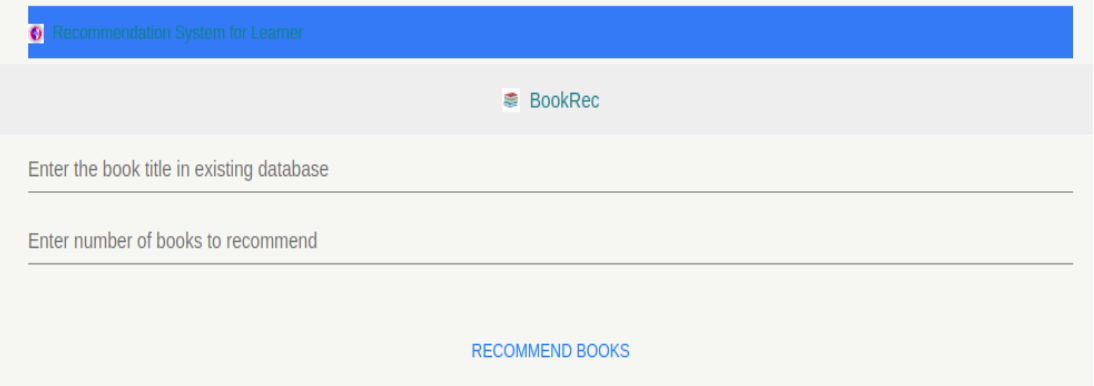


Figure 6.1: Book Recommender module

Above Figure 6.1 shows that Book Recommender module got 2 input fields text and number field and a button which performs book recommendation function

(ii) Implementation Screenshot For Book Recommender Module Input

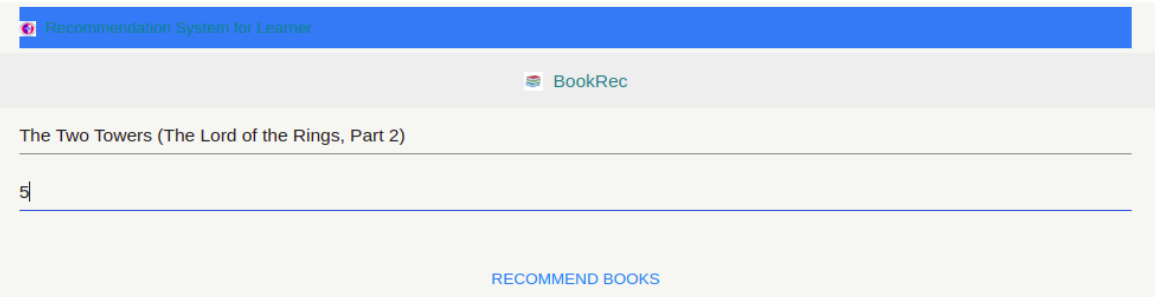


Figure 6.2: Input to the BookRec module

Above Figure 6.2 shows that giving input to the input fields i.e., Title of the book, Number of the book

(iii) Implementation Screenshot For Book Recommender output

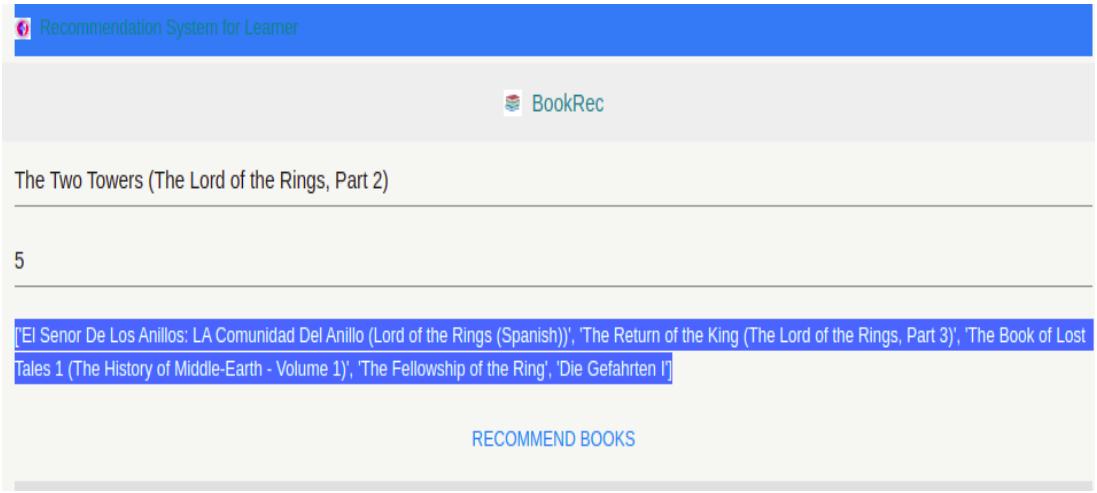


Figure 6.3: Output

Above Figure 6.3 shows that after clicking recommend book button the recommendation function is performed and Output of Recommended books are given

(iii) Implementation Screenshot For Book Recommender output2

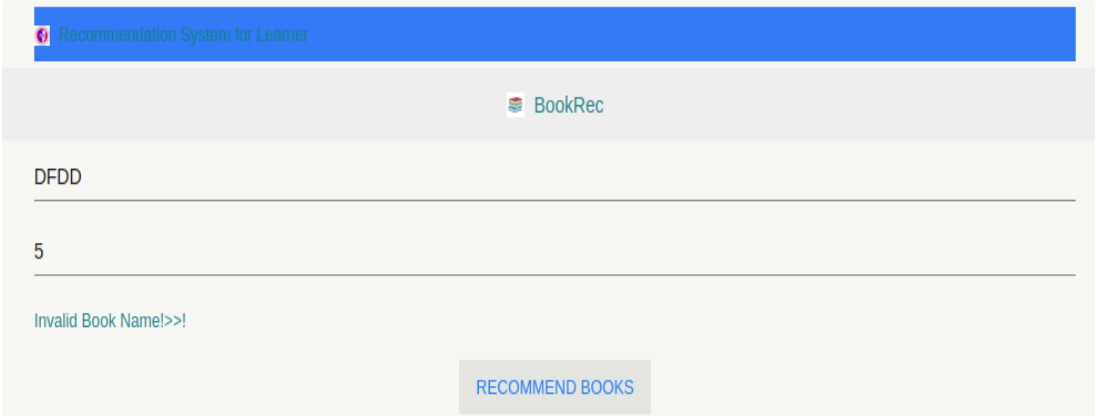


Figure 6.4: Output for invalid book

Above Figure 6.4 shows that after clicking recommend book button the recommendation function is performed and if the entered book name doesn't matches the database book name then the error message will be displayed as invalid book name

(iv) Implementation Screenshot For Journal Recommender

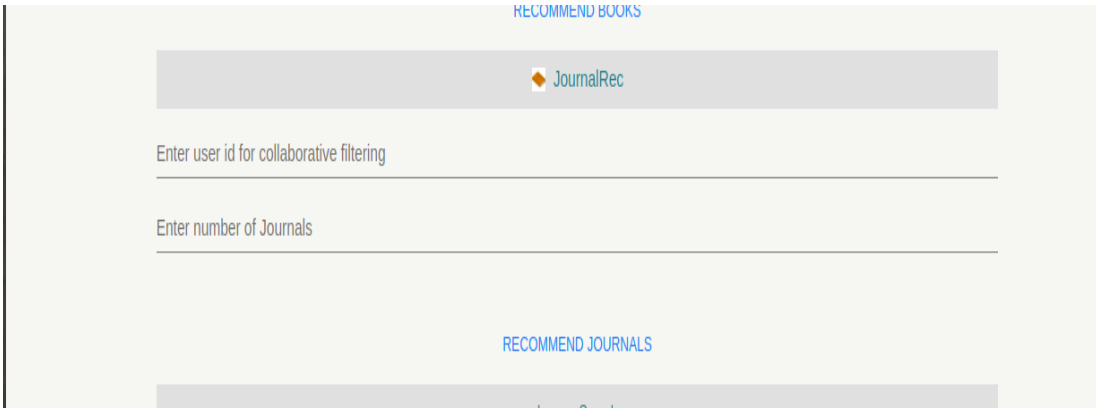


Figure 6.5: Journal recommendation Module using collaborative filtering

Above Figure 6.5 shows Journal recommendation Module using collaborative filtering has got 2 input fields userId and number of journals for recommendation

(v) Implementation Screenshot For Journal Recommender input

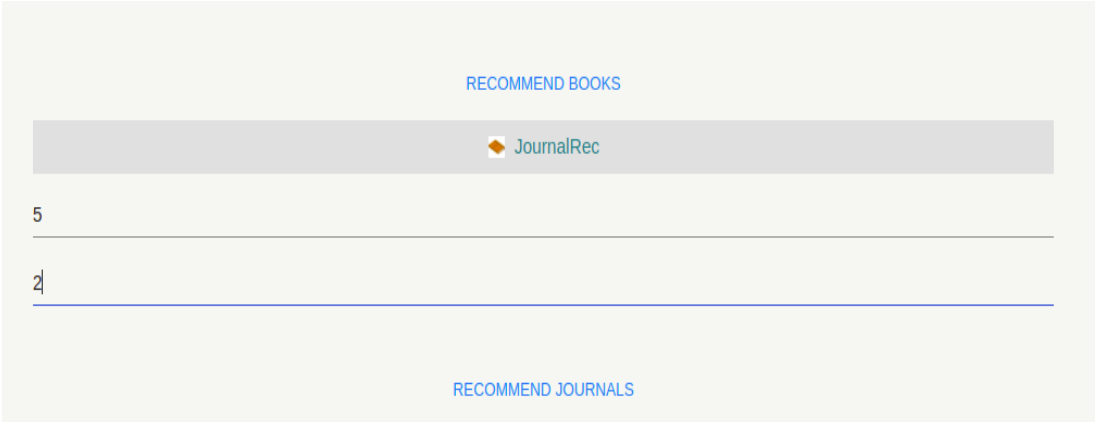


Figure 6.6: Input to the Journal Recommendation

Above Figure depicts 6.6 shows giving input to User ID and Number of books for recommendation

(vi) Implementation Screenshot For Journal Recommender output

Figure 6.7: Output for JournalRec

Above Figure 6.7 depicts - After clicking recommend journal button the recommendation function is performed and Output is be displayed

(vii) Implementation Screenshot For Visualization Graph

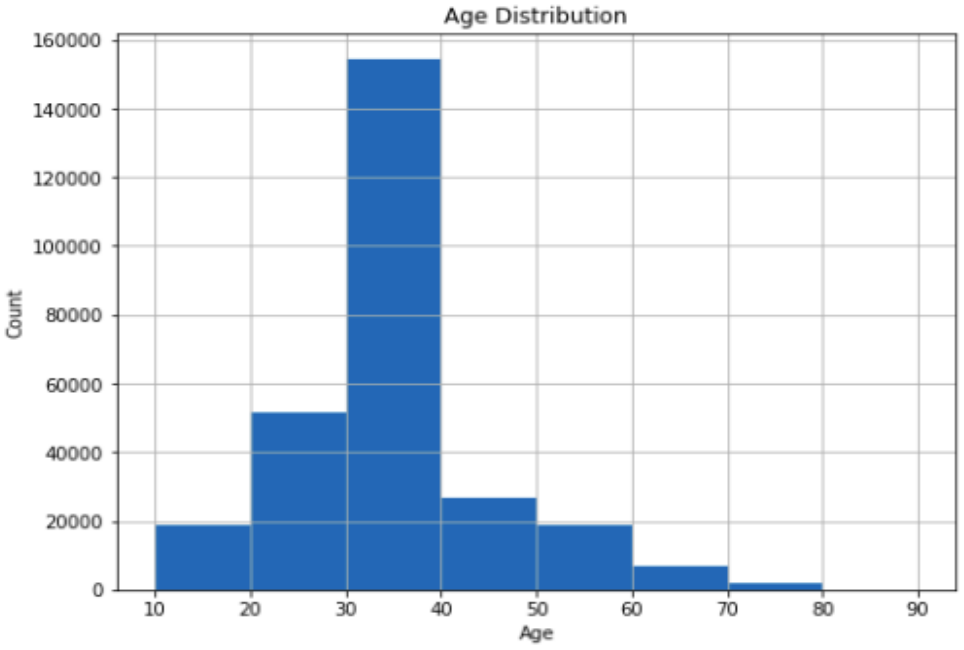
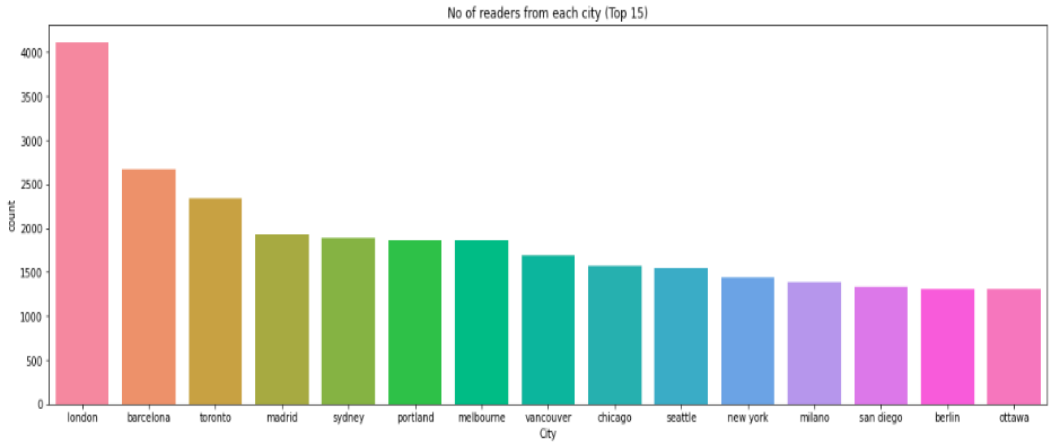


Figure 6.8: Age group of readers and Count of books they read

Above Figure 6.8 describes age in x-axis and count of people in y-axis and the graph depicts the people with age group 30-40 read more books than others

```
]: Text(0.5, 1.0, 'No of readers from each city (Top 15)')
```




```
]: plt.figure(figsize=(20,6))
```

Figure 6.9: City and Count of readers in that city

Above Figure 6.9 describes the city in x-axis and count of people in y-axis.The graph depicts that London has more readers than other cities.

```
Out[57]: Text(0.5, 1.0, 'No of readers from states of USA (Top 15)')
```

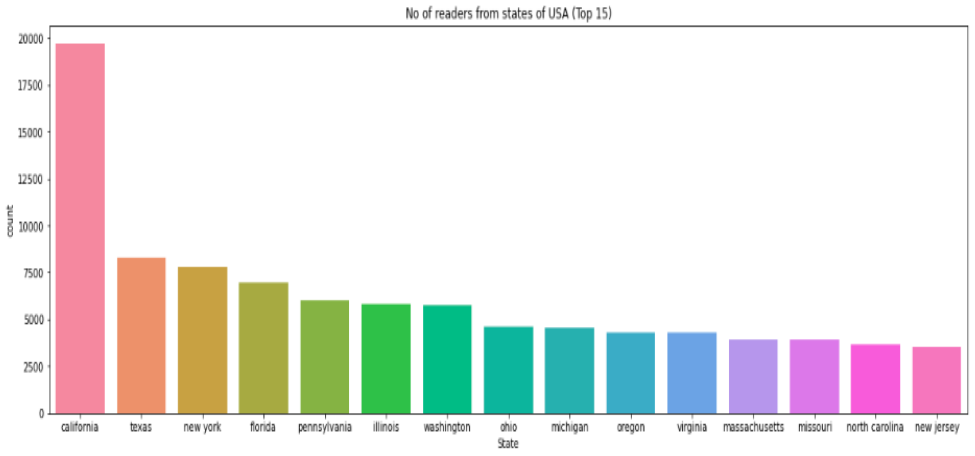


Figure 6.10: State and Count of readers in that State

Above Figure 6.10 describes the state in x-axis and count of people in y-axis and the graph depicts that California has more readers than other states.

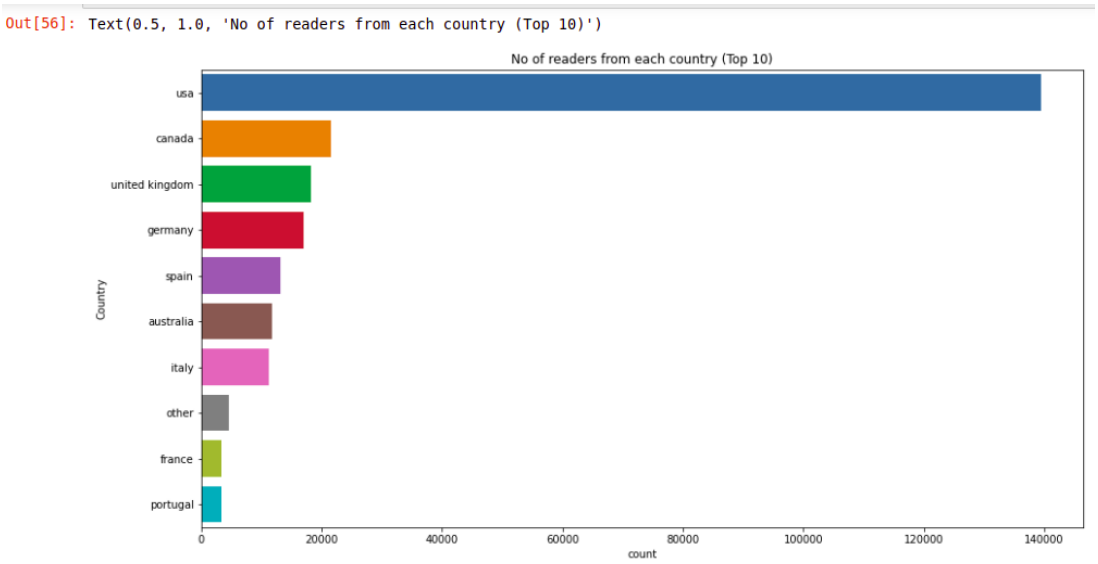


Figure 6.11: No Of readers from each country

Above Figure 6.11 describes the count in x-axis and country in y-axis and the graph depicts that USA has more readers than other countries

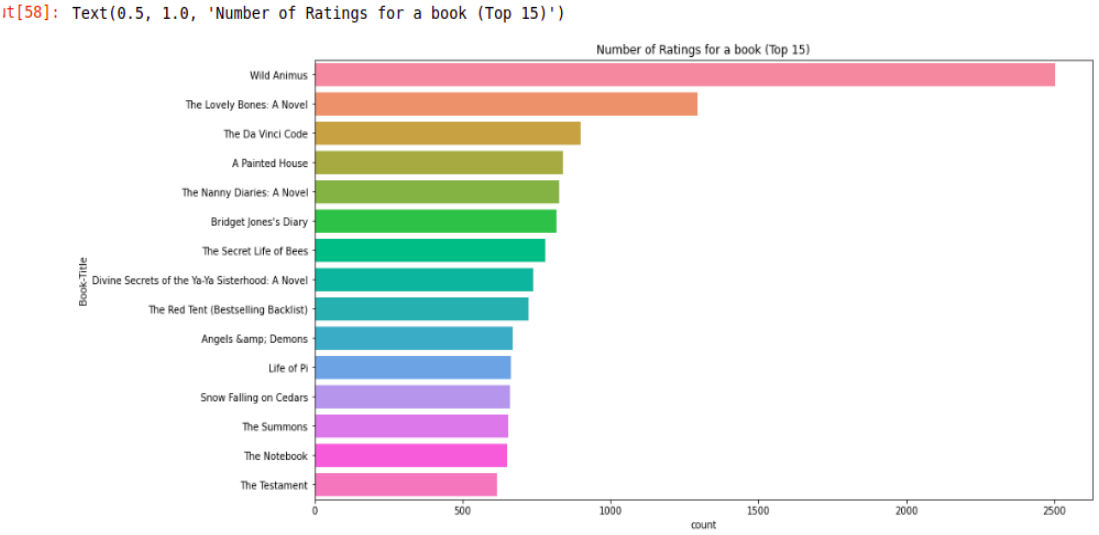


Figure 6.12: Authors and No of Ratings for their books

Above Figure 6.12 describes the ratings in x-axis and book-title in y-axis and the graph depicts that Author Wild Animus has got highest ratings than other authors

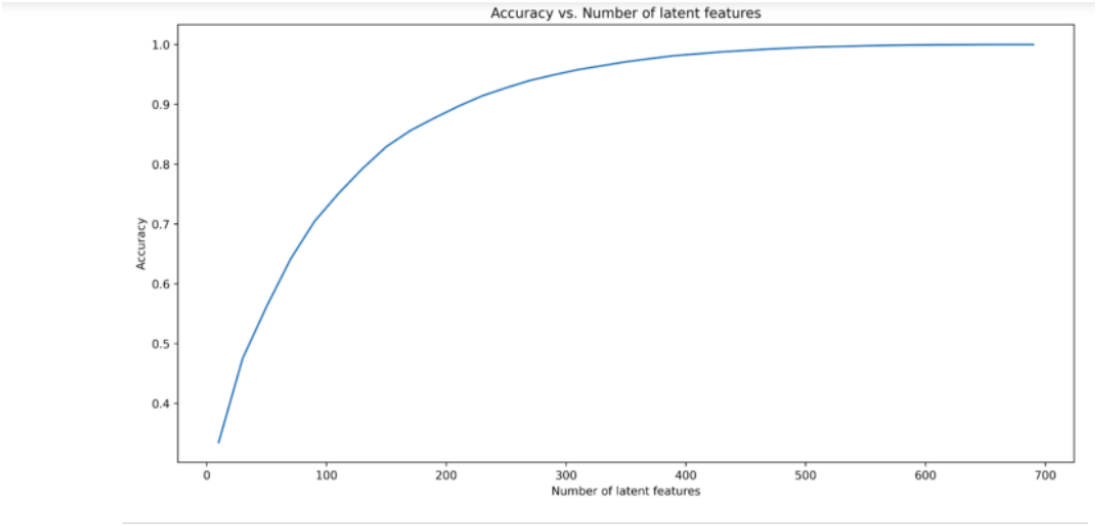


Figure 6.13: Features and Accuracy

Above Figure 6.13 describes the Features in x-axis and accuracy in y-axis and the graph depicts that If feature increases Accuracy also increases

(iix) Implementation Screenshot For Learners Search

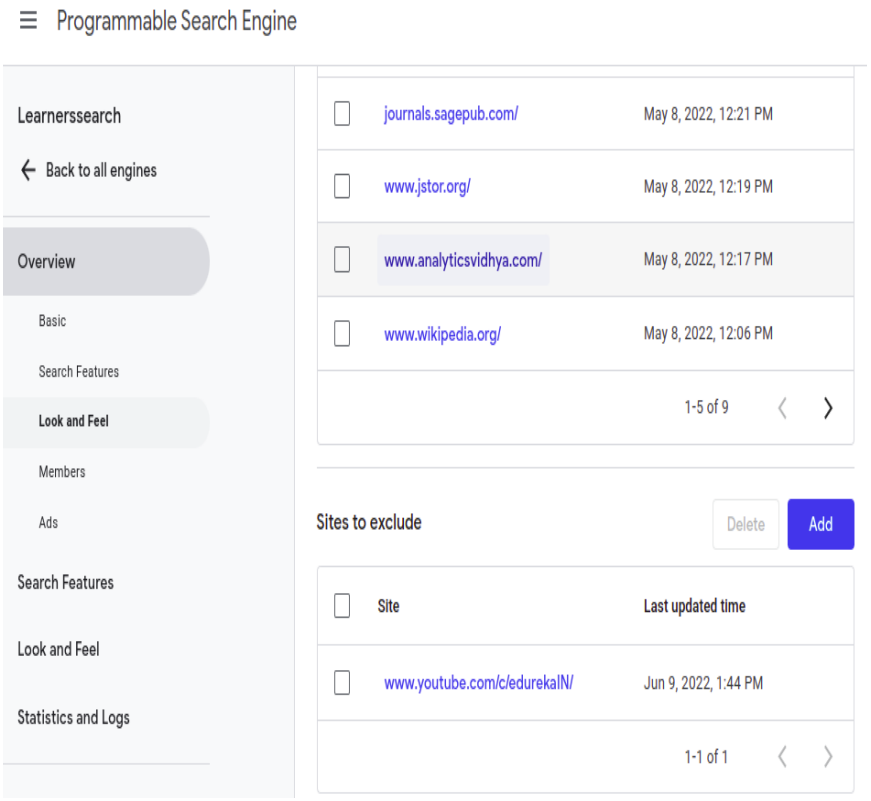


Figure 6.14:Programmable search Engine

Above Figure 6.14 shows the process of adding restrictions to the search results

(ix) Implementation Screenshot For Learners Search Input

Figure 6.15:Input to LearnersSearch

Above Figure depicts the process of Giving input to learners search

(x) Implementation Screenshot For Learners Search output

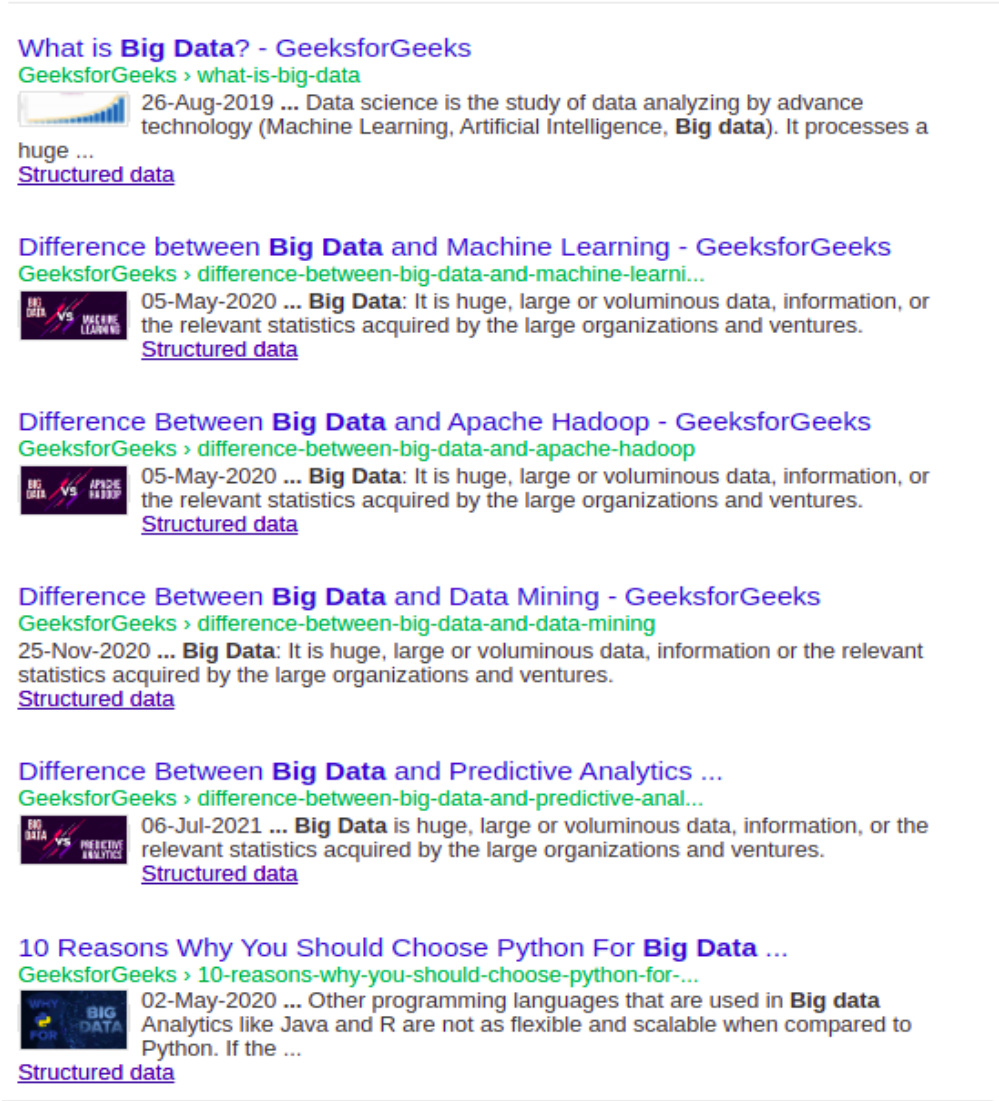


Figure 6.16: Result of the search

Above Figure depicts the process of getting Results which are displayed only from learning contents

(iix) Implementation Screenshot For Learners Search output2

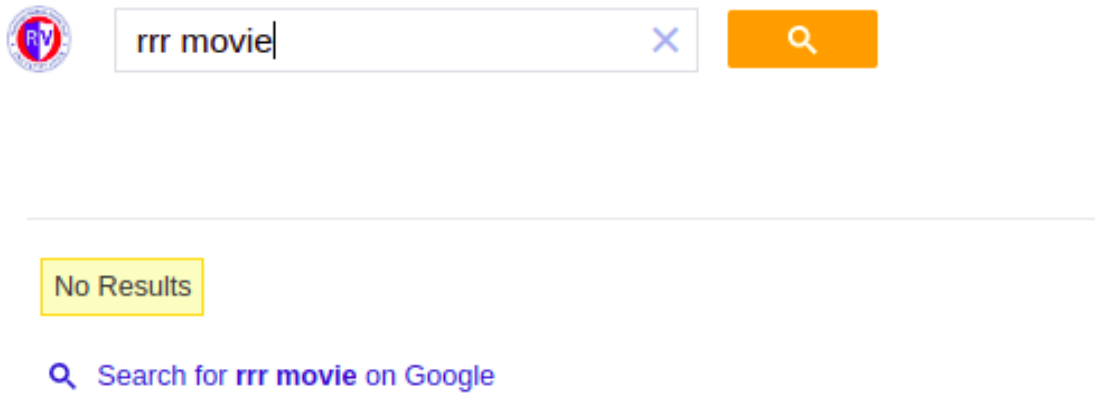


Figure 6.17: output to LearnersSearch

Above Figure depicts the output of No Results because rrr movie is a input which does not belong to learning content

Chapter 7 - Software Testing

Software testing acts as the final assessment of the specification, build, and code and is a crucial component of tool quality assurance. In actuality, testing is the only step in the application engineering process that may be viewed as destructive rather than beneficial.

7.1 Testing and Validation:

Unit Testing: The smallest testable components of an application, known as units, are separately and independently examined for appropriate operation as part of the unit testing phase of software development. The software developers and QA employees perform this testing methodology during the development phase. [40]

Tc_ ID	Feature Tested	Sample Input	Expected Output	Actual Output	Result
UTC 1	Modules To import	import pickle import operator import numpy as np	modules imported successfully	No module named pickle	Fail
	Modules To import	import pickle import operator import numpy as np	modules imported successfully	modules imported successfully	Pass
UTC2	Loading csv file into jupyter	.csv file	Successful loaded data	book data not found	Fail

	Loading csv file into jupyter	.csv file	Successful loaded data	Successful loaded data	Pass
UTC3	converting string to number	.csv file	converted successfully	can't convert non int to int	Fail
	converting string to number	.csv file	converted successfully	converted successfully	Pass

Tc_ID	Feature Tested	Sample Input	Expected Output	Actual Output	Result
UTC4	Drop duplicate rows	.csv file	dropped successfully	dropped successfully	pass
UTC5	Merging of all three Tables	.csv file	Merging Books, Users and Rating Tables in One	Merging Books, Users and Rating Tables in One	Pass

In Unit testing all units in the projects were tested. All the failed test cases were taken care of and necessary changes were made and tested again to produce passed test results. After the unit testing all packages were properly imported and tested. Data was properly loaded and tested. Data preprocessing was properly implemented without any errors.

The failed test cases were taken care as follows:

- In UTC1 the failed test case was taken care of by installing the necessary package by giving the command pip install <module name> in the ipython console.
- In UTC2 the failed test case was taken care of by properly giving the path of the dataset.
- In UTC3 the failed test case was taken care of by dropping the unwanted data.

Integration Testing:

A sort of software testing called integration testing, commonly referred to as integration and testing (I&T), involves testing a software application's many elements, modules, or components as a whole. [41]

Tc_ ID	Feature Tested	Sample Input	Expected Output	Actual Output	Result
ITC1	BookRec module	name of the book	top recommendations of book	error in returning	Fail

	BookRec module	name of the book	top recommendations of book	top recommendations of book	Pass

Tc_ ID	Feature Tested	Sample Input	Expected Output	Actual Output	Result
IT C2	Connecting to wss://anvil.works/uplink	Anvil key	Connected to "Default environment" as SERVER	cannot connect to anvil	Fail
	Connecting to wss://anvil.works/uplink	.Anvil key	Connected to "Default environment" as SERVER	Connected to "Default environment" as SERVER	Pass
IT C3	BookRec module	name of the book	invalid book name	invalid book name	Pass

	BookRec module	name of the book	top recommendations of book	Key Error	Fail

Tc_ID	Feature Tested	Sample Input	Expected Output	Actual Output	Result
ITC4	JournalRec module	name of the journal	invalid journal name	invalid journal name	Pass
	JournalRec	name of the journal	invalid journal name	Key Error	Fail

Tc_ ID	Feature Tested	Sample Input	Expected Output	Actual Output	Result
IT C5	JournalRec module	userId number of journals	top recommendation for the user	key error	Fail
	JournalRec	userId number of journals	top recommendation for the user	top recommendation for the user	Pass
IT C6	programmable Search engine	text	web pages	404 page not found	fail
	programmable Search engine	text	web pages for learners	web pages for learners	pass

In integration testing all modules were properly integrated. All the failed test cases were taken care of and necessary changes were made and tested again to produce passed test results.

The failed test cases were taken care as follows:

- In ITC1 failed test case was taken care by properly giving the parameters to the Book recommendation

- In ITC2 the failed test case was taken care of by giving the uplink key Learners Search module.
- In ITC3 the failed test case was taken care of by giving the validation for the wrong book name in the book recommendation system module.
- In ITC4 the failed test case was taken care of by giving the validation for the wrong journal name in the journal recommendation system module.
- In ITC5 the failed test case was taken care of by properly connecting to the programmable search engine server.

Chapter 8:Conclusion

This chapter gives the summary of the main points of the project

Learners recommendation can be used to manage the recommendation for learners. This can be used to get recommendations in an optimal way which saves time and will be safe for educational purposes.

Learner's recommendation system is a Web-based recommendation system that suggests learners' relevant materials to the users based on users' preference with high accuracy. The main function of this system is to suggest materials based on the learner's context. suggest the materials i.e, learning courses, books, Journals, learning sites etc. Learner's information search module is used to restrict learners for learning content only. Web based recommendation for learners results in returning top recommendations for learners.

The Book Recommendation system uses content based filtering which recommends books based on content like similar authors, similar publications and similar interests. The objective of this module is to recommend books according to the user's interest. The project was successful in establishing this objective.
The journal recommendation uses collaborative filtering which takes into account similar users. This module looks at users that are similar in terms of the materials they have interacted with. These materials could then be recommended to similar users and it's a step towards more personal recommendations. The objective of this module was to recommend articles according to userId. The project is successful in implementing this objective.

The Learners search helps in restriction based search for learning context. After adding restrictions to the module, when the Learner gives the input to learners search results are displayed only from learning contents when the learner searches for any other content other than educational content no Results will be displayed because input will not belong to learning content.

Thus recommendations for learners can be used by learners for quick and easy personalized searching.

Bibliography

[1]"Shun-Feng Su, Xi-Jun Chen, Xi-Jun Chen, Recommender Systems in E-learning, Zhang et al. J Smart Environ Green Compute 2021;1:76-89"

[2] "Mojtaba Salehi, Isa Nakhai Kmalabadi, A Hybrid Attribute-based Recommender System for E-learning Material Recommendation IERI Publication 2 (2021) 565-570"

[3] "Salam Fraihat , Qusai Shambour, A Framework of Semantic Recommender System for e-Learning, Journal of Software Manuscript submitted October 15, 2020; accepted December 28, 2020. doi: 10.17706/jsw.10.3.317-330"

[4] "Shanshan Wan, Zhendong Niu, A Hybrid E-learning Recommendation Approach Based on Learners' Influence Propagation, Zhang et al. J Smart Environ Green Comput 2020;1:76-89"

[5] "Karim Dahdouh1 , Ahmed Dakkak, Lahcen Oughdir and Abdelali Ibriz, Large-scale e-learning recommender system based on Spark and Hadoop, Dahdouh et al Publication. J Big Data (2019) 6:2"

[6] "Pradnya Vaibhav Kulkarni, Sunil Rai and Rohini Kale, Recommender System in eLearning: A Survey, Researchgate (978-981-15-0790-8_13) "

[7] "Hayder Murad, Linda Yang, Personalized E-Learning Recommender System using Multimedia Data , World Academy of Science, Engineering and Technology Vol:11, No:2, 2019"

[8] "Reema Sikka, Amita Dhankhar, Chaavi Rana , A Survey Paper on E-Learning Recommender System, International Journal of Computer Applications (0975 – 888) Volume 47– No.9, June 2019"

[9] "Oana Maria Teodorescu, Building Test Recommender Systems for e-Learning Systems, Proceedings of The 13th International Conference on Educational Data Mining (EDM 2018)"

[10] "John Tarus, Zhendong Niu, Bakhti Khadidja , E-Learning Recommender System Based on Collaborative Filtering and Ontology , Vol:11, No:2, 2019 International Journal of Computer and Information Engineering"

[11] https://www.tandfonline.com/doi/full/10.1080/23311916.2021.2022568

[12] https://globaljournals.org/GJMBR_Volume16/3-Electronic-Commerce-A-Study

[13]https://www.diva-portal.org/smash/get/diva2:1578230/FULLTEXT01

[14]Esteban, A.; Zafra, A.; Romero, C. Helping University Students to Choose Elective Courses by Using a Hybrid Multi-Criteria

[15]Recommendation System with Genetic Optimization. Knowl.-Based Syst. 2020, 194, 105385. [CrossRef]

[16]. Serrano-Guerrero, J.; Herrera-Viedma, E.; Olivas, J.A.; Cerezo, A.; Romero, F.P. A Google Wave-Based Fuzzy Recommender

[17]System to Disseminate Information in University Digital Libraries 2.0. Inf. Sci. 2011, 181, 1503–1516. [CrossRef]

[18] Tejeda-Lorente, Á.; Porcel, C.; Peis, E.; Sanz, R.; Herrera-Viedma, E. A Quality Based Recommender System to Disseminate

Information in a University Digital Library. Inf. Sci. 2014, 261, 52–69. [CrossRef]

[19] He, Q.; Pei, J.; Kifer, D.; Mitra, P.; Giles, L. Context-aware citation recommendation. In Proceedings of the 19th International

Conference on World Wide Web—WWW '10, Raleigh, NC, USA, 26–30 April 2010. [CrossRef]

[20] Wang, D.; Liang, Y.; Xu, D.; Feng, X.; Guan, R. A Content-Based Recommender System for Computer Science Publications.

Knowl.-Based Syst. 2018, 157, 1–9. [CrossRef]

[21]. Park, N.; Roman, R.; Lee, S.; Chung, J.E. User Acceptance of a Digital Library System in Developing Countries: An Application of

the Technology Acceptance Model. Int. J. Inf. Manag. 2009, 29, 196–209. [CrossRef]

[22]. Jeong, H. An Investigation of User Perceptions and Behavioral Intentions towards the E-Library. Libr. Collect. Acquis. Tech. Serv.

2011, 35, 45–60. [CrossRef]

[23]. Achakulvisut, T.; Acuna, D.E.; Ruangrong, T.; Kording, K. Science Concierge: A Fast Content-Based Recommendation System for

Scientific Publications. PLoS ONE 2016, 11, e0158423. [CrossRef]

[24]. Färber, M.; Jatowt, A. Citation Recommendation: Approaches and Datasets. Int. J. Digit. Libr. 2020, 21, 375–405. [CrossRef]

[25]. Brian, D. Netflix Subscriber and Growth Statistics: How Many People Watch Netflix in 2021? Available online: https://backlinko.com/

netflix-users (accessed on 25 February 2021).

[26]. Statista. Disney+'s Number of Subscribers Worldwide from 1st Quarter 2020 to 2nd Quarter 2021. Available online: https://www.statis

ta.com/statistics/1095372/disney-plus-number-of-subscribers-us/ (accessed on 19 August 2021).

[27]. BBC News. Netflix Launches UK Film and TV Streaming Service. Available online: https://web.archive.org/web/20120109175608

/http://www.bbc.co.uk/news/technology-16467432 (accessed on 9 January 2012).Electronics 2022, 11, 141

47 of 48

[28]. PR Newswire. Netflix Launches in Sweden, Denmark, Norway and Finland. Available online: https://web.archive.org/web/201411

27211314/http://www.prnewswire.com/news-releases/netflix-launches-in-sweden-denmark-norway-and-finland-1747495

81.html (accessed on 27 November 2014).

[29]. Variety. Netflix's 'House of Cards' Falls Prey to Piracy. Available online: https://variety.com/2013/digital/news/pirates-swarm-ne

tflixs-house-of-cards-1200333171/ (accessed on 4 April 2013).

[30]. Gupta, G.; Singharia, K. Consumption of OTT Media Streaming in COVID-19 Lockdown: Insights from PLS Analysis. Vision

2021, 25, 36–46. [CrossRef]

[31]. Ryan; Meg; Anousha. Shaken Studios. Empty Theaters. What Hollywood Lost during the Pandemic. Available online: https://www.la

times.com/entertainment-arts/business/story/2020-12-09/everything-hollywood-lost-during-the-pandemic (accessed on 9

December 2020).

[32]. Insider Intelligence. Top OTT VOD Streaming Services in 2021 by Viewer Count and Growth. Available online: https://www.insideri

ntelligence.com/insights/ott-vod-video-streaming-services/ (accessed on 21 July 2021).

[33]. Statista. Facebook Keeps on Growing. Available online: https://www.statista.com/chart/10047/facebooks-monthly-active-users/

(accessed on 4 February 2021).

[34]. Prabhakaran. 50+ Instagram Stats Everyone Must Know in 2020. Available online: https://www.connectivasystems.com/instagra

m-stats-2020/ (accessed on 1 April 2020).

[35]. Statista. Number of Monthly Active Twitter Users Worldwide from 1st Quarter 2010 to 1st Quarter 2019. Available online: https://www.st

atista.com/statistics/282087/number-of-monthly-active-twitter-users/ (accessed on 31 December 2021).

[36]. Hughes, D.J.; Rowe, M.; Batey, M.; Lee, A. A Tale of Two Sites: Twitter vs. Facebook and the Personality Predictors of Social

Media Usage. Comput. Hum. Behav. 2012, 28, 561–569. [CrossRef]

[37]. Highfield, T.; Leaver, T. A Methodology for Mapping Instagram Hashtags. First Monday 2014, 20, 1–11. [CrossRef]

[38]. Sarah, F. The Inside Story of How Facebook Acquired Instagram. Available online: https://onezero.medium.com/the-inside-story-of

-how-facebook-acquired-instagram-318f244f1283 (accessed on 5 August 2020).

[39]. Dave, S. Chart of the Day: Instagram Is Now Bigger Than Twitter. Available online: https://www.businessinsider.com.au/chart-of-

the-day-instagram-is-now-bigger-than-twitter-2014-12 (accessed on 11 December 2014).

A Web Based recommendation system for learner using NLP

ORIGINALITY REPORT

12%	7%	2%	5%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	github.com Internet Source	4%
2	Submitted to (school name not available) Student Paper	2%
3	medium.com Internet Source	1%
4	Submitted to Higher Education Commission Pakistan Student Paper	<1%
5	en.wikipedia.org Internet Source	<1%
6	Submitted to University of Leicester Student Paper	<1%
7	ia801805.us.archive.org Internet Source	<1%
8	Submitted to The British College Student Paper	<1%
9	Submitted to Arab Open University Student Paper	<1%