islington college
(इस्लिङ्टन कलेज)

Module Code & Module Title

CU6051NI - Artificial Intelligence

Assessment Weightage & Type

25% Individual Coursework

Year and Semester

2020-21 Autumn

Student Name: Ashok Lama

London Met ID: 20048836

College ID: NP01CP4S210191

Assignment Due Date: 11th January 2023

Assignment Submission Date: 9th January 2023

I confirm that I understand my coursework needs to be submitted online via Google Classroom under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a marks of zero will be awarded.

Abstract

The paper that follows provides a thorough explanation of a book recommendation system built on collaborative filtering using KNN (K closest neighbour) and the Euclidean formula. The main goal of the project is to provide a practical framework for book proposals to improve the standard of administrations offered by any company or website for a book per user. This paper, which serves as the second AI coursework, summarizes the system's development process and the kinds of AI concepts that were applied. This section offers a visual representation of the Introduction, Background, Solution, Conclusion, References, and Appendix chapters of the AI report. In the introduction, the issue is succinctly described. The development of a recommendation system for applications based on books is the aim of this research. The customer will be given recommendations for books based on this technology. This time-saving and ingenious technology aids book readers who are having difficulty deciding about which book to read. This system is created using data from a previous similar book review.

Table of Contents

Table of Figures

Table of Equations

1 Introduction

1.1 AI Recommendation System

An artificial intelligence (AI) application called a recommendation system uses Big Data and is typically linked to machine learning to provide customers advice or suggest extra products or services. Numerous factors, including prior purchases, search history, demographic information, and other factors, may influence these. Recommender systems are quite useful for helping individuals find goods and services they might not have discovered on their own (Cremonesi and Jannach, 2021).

The recommendation system is utilized in many different IT applications, mostly in e-commerce ones but increasingly now in social networking sites like Facebook and Instagram. To make suggestions, this technology gathers information from user interactions with the products. If someone searches for and watches movies, for example, on Instagram, and engages with the video's genre, such as anime, the recommendation engine may start to suggest content that includes details about, images of, and films about anime (Jannach et al., 2021).

How quickly a customer leaves a company depends on how easily they can use the product or service. Click-through rates can go up and client reach can be extended thanks to an AI recommendation system. They speed up the process of seeking for pertinent and practical things online. It is possible to greatly enhance the user experience, which will increase customer retention and acquisition rates. Cross-sales make for 36% of Amazon's overall sales, according to a statement made by the company's CEO in 2006 (Jannach et al., 2021).

1.2 Online Course Recommendation system

People are currently quite busy learning new things, yet they are unfamiliar with the topic. Therefore, the Online Course Prediction system is essential to making user searches simple. There are many different instances, such as when a very new beginning is unsure of what course to take after finishing the python classes. In these cases, the system suggests that the user go on to a different topic after searching for or finishing a course (Zhang et al., 2018).

With the development of information technology, online learning has become a technique of knowledge acquisition that is more required. But as the amount of available instructional resources grows, it is harder and harder to choose the correct learning materials among them. Users can easily get personalized resources from thousands of information sources thanks to the recommender system's excellent information filtering ability, which helps to address the issue of information overload. Due to recommender systems' high utility, a lot of new research has also been offered in recent years; nonetheless, there are currently few works on online course recommendation (Obeidat et al., 2019).

Today, online course recommendations are employed in many different applications. Examples include Udemy, Coursera, Skillshare, and others. To put it another way, without an online course recommendation system, eLearning platforms would not be able to maintain user interest while using the application (Revathi et al., 2020).

2 Background

2.1 Research work done on chosen topic / problem domain

The project's chosen topic is a recommendation system, which suggests a course to the user after they enrol in or finish a course. Many research papers and issues were discovered while developing this system, some of which are listed below:

2.1.1 Research done in Coursework 1

i) Online course and eLearning

➔ A learning program that is done in a virtual setting and is organized in accordance with a syllabus is known as an online course. Online courses can be informal and focused on learning a specific skill, or formal and leading to a certification or degree. Various elements distinguish different online courses: While some might rigorously follow a syllabus, others might let people sign up for and opt out of the class as their schedules allow. While some might videoconference "live" discussion groups and record lectures to make them accessible to users at any time, others would only schedule recorded discussion sessions (Khan and Joshi, 2006).

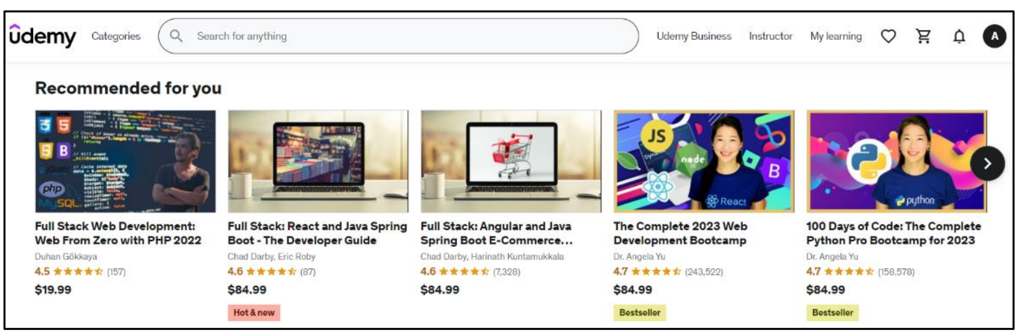

Figure 1: Online course recommendation system

E-learning, also referred to as web-based training, entails giving students access to education through their browsers whenever and wherever they want via the internet or a company intranet. Students, personnel undergoing training, and casual learners can all participate in a structured learning experience using elearning regardless of where they are physically located, in contrast to traditional learning methods (Khan and Joshi, 2006).

ii) Collaborative Filtering

➔ The two types of collaborative filtering they are listed below:

a) User-Based Collaborative filtering

➔ Collaborative User-Based According to the ratings that other users who have likes similar to the target user have given a product, filtering is a technique for predicting the products that a user will like. First, a formula is used to determine how similar the two users are. The target user may then have a lot in common with certain users while being very different from the others. Therefore, it makes sense to assign greater weight to user ratings that are more similar to one another than those that are less similar, and vice versa. This problem can be solved via a method based on weighted averages. This approach multiplies a user similarity factor by each user's rating (Wang et al., 2006).

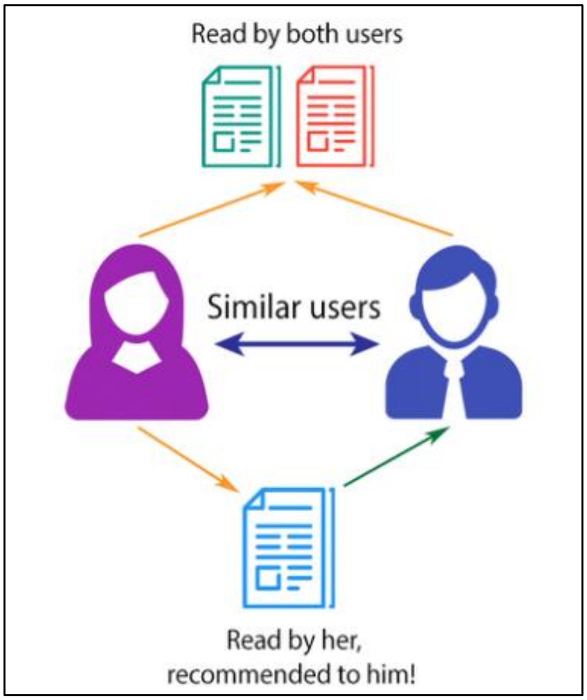

Figure 2: User based collaborative filtering

b) Item based Collaborative filtering

➔ Instead of comparing the user to similar consumers, item-to-item collaborative filtering compares each of the user's purchased and rated goods to comparable items before adding those comparable items to a recommendation list. Finding commonalities between each pair of elements is the first step in modelling. The similarity between two item pairs can be calculated using a variety of techniques. One of the most popular methods is the usage of cosine similarity. In the second step, a recommendation system is put into action. It provides ratings based on the items that are most similar to the missing item (and have already been evaluated by the user). Make predictions based on the ratings of similar products, and then calculate those predictions using a method that assigns a rating to a particular item by applying the weighted average of ratings for similar products (Wang et al., 2006).

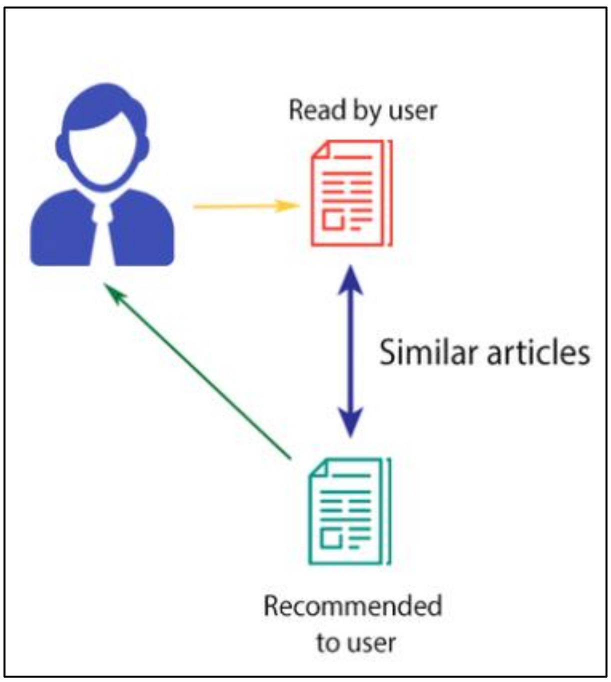

Figure 3: Item based collaborative filtering

iii) Advantages and Disadvantages of online course recommendation system

➔ There are always pros and cons of every system. During the research for this project many drawbacks and advantages were listed out some of them are listed below:

a) Advantages

i) It engages user in the application.

ii) User gets the suggestions and path to another useful topics.

iii) It saves the time of the user.

iv) It suggests the best and top-rated courses which is equivalent to user.

v) It increases the productiveness of the application

vi) The more user interactions the more data is collected which makes increases the system accuracy.

b) Disadvantages

i) The suggested course might not be of their field of interest.

ii) If the data for recommendation is not enough or accurate the user might get not accurate recommendations.

iii) The recommended course might not be interesting for user.

2.2 Review and analysis of existing work in the problem domain

2.2.1 Some of the similar project implementations are listed below

There are many implementations of online course recommendation system in many eLearning platforms like Udemy, Coursera, and many other applications. Some of the application uses content-based algorithm while some uses user based collaborative filtering algorithm.

a) Udemy user based, and content based collaborative filtering

➔ Udemy uses both user and item-based filtering where the first column recommends course according to user previous search of item. Where second column suggests courses by comparing user profile and preferences with another user.

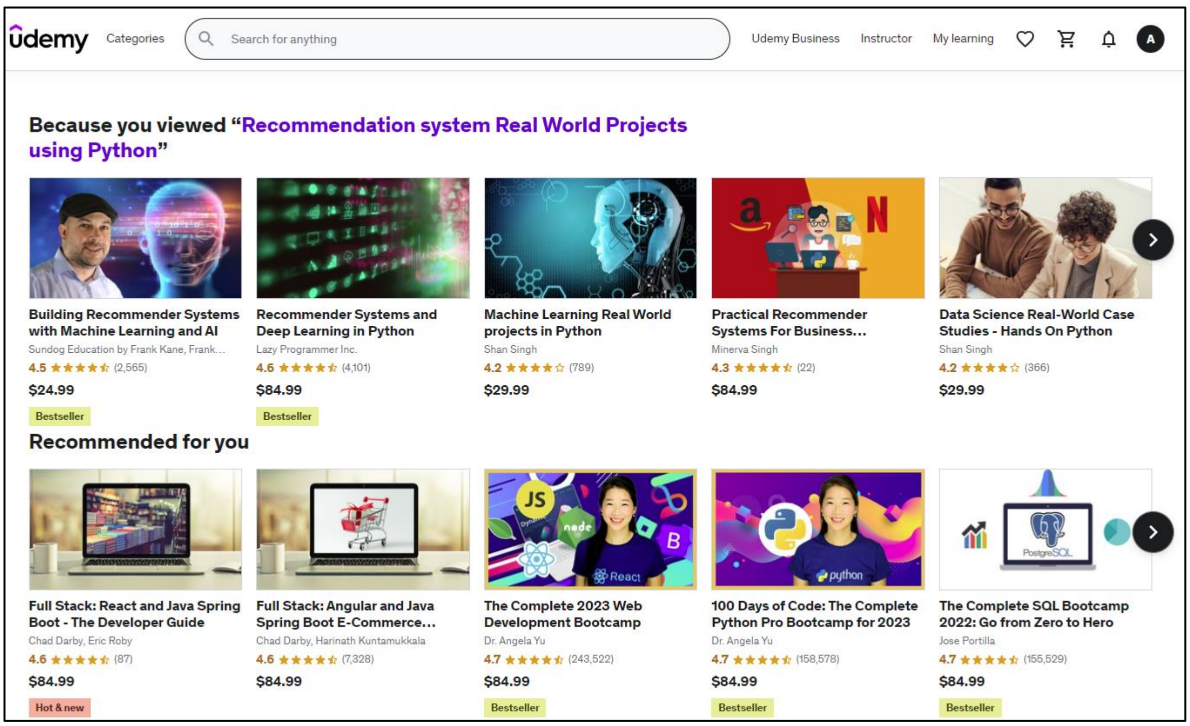

Figure 4: Udemy course recommendation system

b) Coursera item-based recommendation system

➔ Coursera is a big name for eLearning platform on of its feature is course personalization of course based on user search history. It gives the best suggestion based on their available contents.

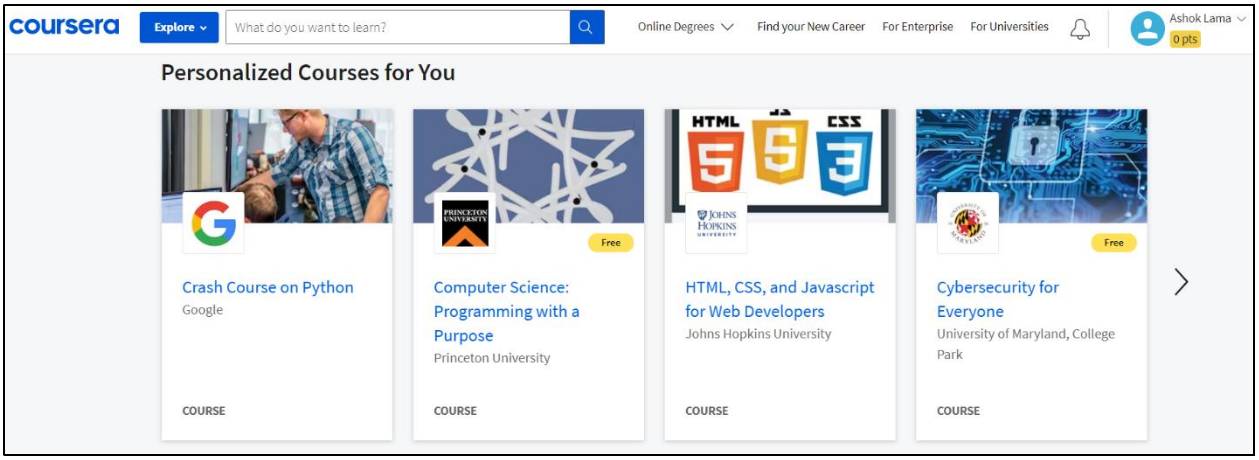

Figure 5: Coursera recommendation system

c) Linked Learning Recommendation system

➔ Linked learning also uses the content-based filtering where the user gets recommendation based on their previous search histories and preview of courses.

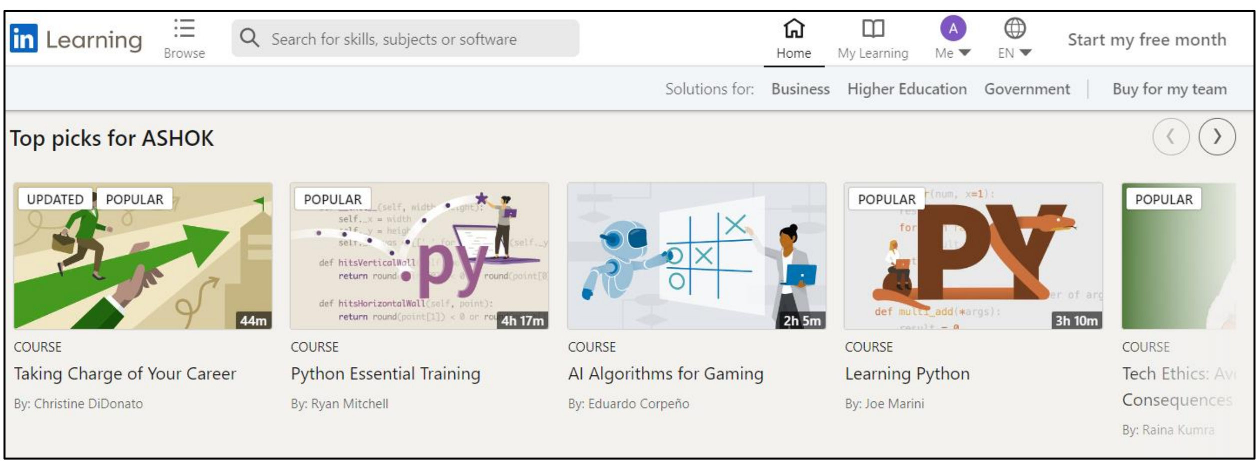

Figure 6: Linked Learning recommendation system

2.3 Research Papers

2.3.1 A Collaborative Online Course Recommendation System

[Published in: 2019 International Conference on Deep Learning and Machine Learning in Emerging Applications (Deep-ML)]

The collaborative recommender system described in this research suggests online courses for students based on similarities in their prior course experience. The suggested method uses data mining techniques to identify trends among the courses. In contrast to association rules produced utilizing the entire collection of courses and students, clustering students into comparable groups based on their individual course choices is crucial for producing association rules of high quality. To develop association rules, the Apriori algorithm was employed twice, once with the entire dataset and once with the clusters that are created based on students' course preferences. The findings show that the rules generated on clusters have higher coverage. Additionally, the SPADE algorithm was applied to the course sequences to evaluate the impact of course dependency on suggestions. The outcomes are consistent with those attained when Apriori was used (Obeidat et al., 2019).

2.3.2 Approach for building course recommendation system

[Conference: 2016 Eighth International Conference on Knowledge and Systems Engineering (KSE)]

This research article presents several techniques for creating course recommendation systems. Students may choose the right courses and predict their learning outcomes early by using a course recommendation system, which will help them make better study plans. Following the description of the approaches, this study compares and evaluates their effectiveness using actual educational data. The best model is then chosen to construct the course recommendation system. The proposed course recommendation system can be used in practice, according to preliminary results (Revathi et al., 2020).

2.3.3 Recommendation system using Content based, collaborative filtering and hybrid techniques

[Part of the Advances in Intelligent Systems and Computing book series (AISC,volume 569)]

Nowadays there are many courses available for students, and sometimes it is hard for a student to perceive information related to those courses and decide which course to take. This research work aims to build a system to suggest online courses to users based on their profile and the similarity with other users. For that three techniques are suggested to extract the information and suggest online courses: Content Based, Collaborative filtering and Hybrid. By combining these three techniques the system can offer more accurate recommendations and only considers the interests of each user. Thus, users will not feel tired while perceiving information of their interest and will keep engaged and interested to use the system (Zhang et al., 2018).

2.3.4 PCRS: Personalized Course Recommender System Based on Hybrid Approach

[https://doi.org/10.1016/j.procs.2017.12.067]

The traditional method of choosing courses to do research is a time-consuming, dangerous, and tiresome activity that not only has a negative impact on a researcher's performance but also on their learning experience. Therefore, picking the right courses during pivotal years could aid in conducting research more effectively. This study offers a recommender system that can help a learner choose the courses that best suit their needs. Ontology has been used with the hybrid methodology to retrieve pertinent data and generate precise recommendations. Such a strategy might help students perform better while also increasing their level of happiness. By addressing the shortcomings of fundamental individual recommender systems, the proposed recommender systems would perform better.

2.3.5 Personalized Course Recommendation System Fusing with Knowledge Graph and Collaborative Filtering

[https://doi.org/10.1155/2021/9590502]

One of the hottest areas in the online education industry is personalized course recommendation technology. An effective recommendation algorithm can pique students' interest and fully exploit their various learning personalities. Currently, the widely used collaborative filtering algorithm disregards the semantic link between suggestion items, leading to subpar recommendation outcomes. This study suggests a method that combines collaborative filtering with knowledge graphs. The semantic information of the items is first embedded into a low-dimensional semantic space using the knowledge graph representation learning method; next, the semantic similarity between the recommended items is determined; finally, this item semantic information is combined with the collaborative filtering recommendation algorithm. This algorithm improves suggestion performance at the semantic level. The findings demonstrate that the suggested algorithm is superior to the conventional recommendation algorithm in terms of precision, recall, and F1 and is capable of effectively recommending courses to learners.

2.4 Summary of Research done

After reading the research report and the existing solution to the problem, many deductions were made. The suggestion list was filtered using the collaborative filtering method across all of the study papers. The dataset structure and features needed for the course were also discovered. Previous solutions to the problem domain made it clear that there are a variety of strategies, and the strategy depends on the dataset structure. Content-based and user-based filtering databases are separated. A hybrid or item-based recommendation system, however, is occasionally the best option with a system like this. It should only contain the aspects that are required for recommendations. csv files for courses since it would increase system effectiveness and performance. Additionally, the data are cleaned with only the necessary dataset in the table following additional data merging.

3 Solution

3.1 Explanation of the recommendation System

For the solution firstly the two .csv file is for the course description and course ratings along with the customer id and rating id. Then in the coding part the codes will be separated into multiple cells of JuPyter notebook.

3.1.1 Importing required libraries

```
#importing for reading csv files
import pandas as pd

#importing for data visualization based on matplotlib
import seaborn as sns

# for spliting the data into train and test
from sklearn.model_selection import train_test_split

#for sparse matrix and nearest neighbor
from scipy.sparse import csr_matrix
from sklearn.neighbors import NearestNeighbors
```

Figure 7: Importing packages

In the above figure the pandas, seaborn and train_test_split package is imported. Pandas is used for reading and manipulating the data o csv file. Seaborn is used for the graphical representation and data visualization. Likewise, the train_test_split is used for splitting the data into trained and test. And csr_matrix is imported for creating a sparse matrix and NearestNeighbours is import for finding the nearest correlation value.

```
# reading the course csv file using pandas
dfCourse = pd.read_csv("./CSV/course.csv")
dfCourse.head()
```

	courseId	courseName	university	level	courseDescription	skills
0	100009	Write A Feature Length Screenplay For Film Or ...	Michigan State University	Beginner	Write a Full Length Feature Film Script In th...	Drama Comedy peering screenwriting film D...
1	100024	Business Strategy: Business Model Canvas Analy...	Coursera Project Network	Beginner	By the end of this guided project, you will be...	Finance business plan persona (user experien...
2	1001	Silicon Thin Film Solar Cells	ï¿½cole Polytechnique	Advanced	This course consists of a general presentation...	chemistry physics Solar Energy film lambda...
3	100145	Finance for Managers	IESE Business School	Intermediate	When it comes to numbers, there is always more...	accounts receivable dupont analysis analysis...
4	100293	Retrieve Data using Single-Table SQL Queries	Coursera Project Network	Beginner	In this course youï¿½ll learn how to effective...	Data Analysis select (sql) database manageme...

Figure 8: Reading the course description file

```
# reading the rating file and storing as dfRating
dfRating = pd.read_csv('./CSV/rating.csv')
dfRating.head()
```

	userId	courseId	rating
0	U0	100009	10
1	U0	100024	7
2	U0	1001	7
3	U196	100145	7
4	U186	100293	7

Figure 9: Reading the course rating file

In the above figure the .csv course description and rating file is read and the top 5 data in the file is listed. The file is read using the panda's library also the data from the files are stored in dfRating and dfCourse.

3.1.2 Merging and cleaning the data

```
# creating new instane df and merging the both dfRating and dfBook and cleaning data
df = pd.merge(dfRating, dfCourse,on='courseId')
df.drop(['university', 'level', 'skills', 'courseDescription'], axis=1, inplace=True)
df.head()
```

	userId	courseId	rating	courseName
0	U0	100009	10	Write A Feature Length Screenplay For Film Or ...
1	U0	100024	7	Business Strategy: Business Model Canvas Analy...
2	U207	100024	7	Business Strategy: Business Model Canvas Analy...
3	U92	100024	1	Business Strategy: Business Model Canvas Analy...
4	U109	100024	7	Business Strategy: Business Model Canvas Analy...

Figure 10: Merging and cleaning the data

In the above figure the data from course description and rating is merged based on their courseId. Since, both the file has the courseId as a common feature. After merging the data, the unwanted feature like university, level, level and description are removed and stored in the df variable. After cleaning and merging the top 5 data of df variable are listed in the tabular format.

3.1.3 Splitting train and test value

```
train, test = train_test_split(df, test_size=0.2, random_state=42)
train.head()
```

	userId	courseId	rating	courseName
1269	U85	190136	9	Functional Programming in Scala Capstone
1532	U184	22996	7	Build Data Analysis tools using R and DPLYR
617	U94	12262	9	Fundamentals of Immunology: T Cells and Signaling
1405	U249	21199	9	Introduction to Programming and Animation with...
1134	U94	170941	9	Gmail: The Foundation To Accessing Google Apps

Figure 11: Splitting train and test and displaying train data

```
test.head()
```

	userId	courseId	rating	courseName
1752	U297	233310	7	System Validation (4): Modelling Software, Pro...
367	U25	11482	7	FPGA computing systems: Background knowledge a...
1247	U270	1880	10	Cloud Networking
849	U8	144460	9	Optimize TensorFlow Models For Deployment with...
1747	U6	233268	10	Academic Skills for University Success: Capstone

Figure 12: Displaying the test data

In the above figure the train and test data are splatted into 80% and 20% using sklearn train_test_split package. The splitted data of train and test are displayed.

3.1.4 Creating the final data for rating and visualizing

```
finalRating = pd.DataFrame(df.groupby('courseName')['rating'].mean())
finalRating['totalRating'] = pd.DataFrame(df.groupby('courseName')['rating'].count())
finalRating.head()
```

courseName	rating	totalRating
"Making" Progress Teach-Out	7.0	1
3D Printing Applications	5.0	1
3D Printing Hardware	7.0	1
3D Printing Software	7.0	1
A Circular Economy of Metals: Towards a Sustainable Societal Metabolism	9.0	1

Figure 13: Creating the final data for the rating

The final rating data is created using the df data and the column are grouped by rating and course names also the total rating column is also added which stores the total repetition of that column or the total rating count of the course.

```
# plotting the total rating and the rating from the dfRating
sns.jointplot(x='totalRating',y='rating',data=finalRating,alpha=1)
```

```
<seaborn.axisgrid.JointGrid at 0x1e13ee44af0>
```

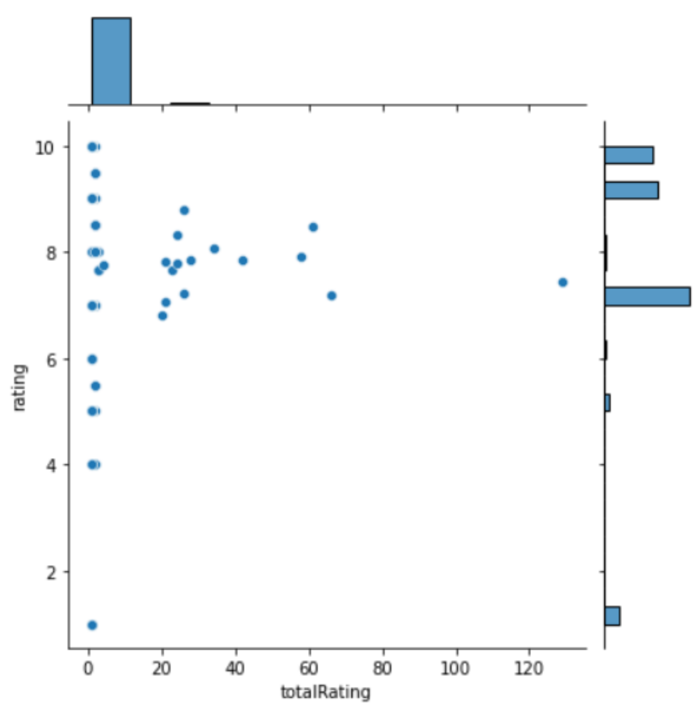

Figure 14: Visualizing the final rating in the graph

The data of the final rating is visualized in the graph. The x axis contains the value of the total rating count, and the y axis contains the average rating of the course. All the data are plotted in the graph using the seaborn.

3.1.5 Creating the pivot table

```
# creating the pivotTable for reorganizing the data and summarizing rating of courses for each user
coursePivot = df.pivot_table(index='courseName',columns='userId',values='rating').fillna(0)
coursePivot
```

userId / courseName	U0	U1	U10	U100	U101	U102	U103	U104	U105	U106	...	U89	U90	U91	U92	U93	U94	U95	U96	U97	U99
"Making" Progress Teach-Out	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3D Printing Applications	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3D Printing Hardware	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3D Printing Software	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
A Circular Economy of Metals: Towards a Sustainable Societal Metabolism	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
Writing in English at University	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Writing in First Person Point of View	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Writing your World: Finding yourself in the academic space	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
eHealth: More than just an electronic record	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
iOS Project: Transreality Game	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

1608 rows × 286 columns

Figure 15: Creating the pivot table for merged and cleaned data

The cleaned and merged data is taken, and the pivot table is created. The pivot table contains the user id in the columns and the first index of every row contains the course name. This table displays the rating of each course provided by the user. And the course that are not rated are replaced with zero. This is the dot product of the rating and the course table. This table is used for finding the nearest similar course.

3.1.6 Defining the recommendation method.

```
sparseCourse=csr_matrix(coursePivot)
model=NearestNeighbors(algorithm='brute')
model.fit(sparseCourse)
```

```
NearestNeighbors(algorithm='brute')
```

```
def courseRecommendation(courseName):
    try:
        courseId=np.where(coursePivot.index.str.lower()==courseName.lower())[0][0]
        distances, suggestions=model.kneighbors(coursePivot.iloc[courseId,:].values.reshape(1,-1), n_neighbors=5)

        for i in range(len(suggestions)):
            print("The recommendations for", courseName,"are:")
            print(coursePivot.index[suggestions[i]])
    except:
        print("Sorry!! No Recommendations for course",courseName)
```

Figure 16: Defining the recommendation method and nearest neighbours

First the sparse matrix for course is created and the brute algorithm is used to find the nearest neighbour and the model is created using the course sparse matrix. In the recommendation function the courseName is taken as parameter and from the pivot table the courseId of the courseName is tried to find out and if the courseId is found then the suggestion and distance of each nearest course are listed else the course id is not discovered then the error message is displayed.

3.1.7 Displaying the recommended course

```
# Taking user input for recommending the courses.
selectCourse = input("Select a course from above list: ")
print ('Selected course is -->', selectCourse)
```

```
Select a course from above list: C++ For C Programmers, Part B
Selected course is --> C++ For C Programmers, Part B
```

```
courseRecommendation(selectCourse)
```

```
The recommendations for C++ For C Programmers, Part B are:
Index(['C++ For C Programmers, Part B', 'Building a Data Science Team',
       'Blended Language Learning: Design and Practice for Teachers',
       'Introduction to solar cells',
       'Business Intelligence Concepts, Tools, and Applications'],
      dtype='object', name='courseName')
```

Figure 17: Recommending course for "C++ For C Programmers, Part B"

The user is asked to select the course name from the list generated by the final Rating and the selected course is used as parameter and the course recommendation method is run. After that if the courseName is valid the course is recommended else the error message is displayed.

3.2 Summary of development

For this recommendation system the content based collaborative filtering is used with the KNN. The closest distance with the other courses is recommended. This system uses the user rating to the course and compares the behaviour or pattern of the course with other courses. Firstly, the two-csv file is merged, and the unwanted data and repeating data are cleaned. After that the merged data is splitted into train and test data after which the train data is converted into the pivot table and from the pivot table the KNN model is created where the brute algorithm is used. After that the nearest member of the selected course is compared and the top 5 nearest members are recommended to the user.

For completing this system most used libraries are pandas for handling and manipulating data NumPy for the data array and the seaborn for plotting graphs and representing the data in graphical format.

3.3 AI/Algorithm used

Machine learning algorithms and deep learning algorithms are two types of artificial intelligence algorithms. The primary objective of these algorithms is to empower computers to learn independently, come to decisions, and recognize relevant patterns. Algorithms for artificial intelligence learn from the data itself (Baloglu et al., 2022).

For this project collaborative filtering algorithm is used. By this approach the recommendation will be provided based on user provided input. The similar courses will be displayed among all the courses with the user search. The steps in the system build are listed below:

Step 1: Start

Step 2: Importing the packages required

Step 3: Reading and storing csv file using panda's library

Step 4: Merging the data of rating and book and cleaning unwanted data

Step 5: Counting the total rating of the books by the rating count

Step 6: Plotting the total rating and average rating of the book using seaborn

Step 7: Pivot table for books rating by user is created.

Step 8: Defining the KNN model and algorithm

Step 9: Defining the recommendation function with one parameter.

Step 9: User input is taken for books.

Step 9: The input course top 5 similar courses are recommended

Step 10 End

The algorithm is very simple and straight. The project will also be done in same format where the import is done, and the calculations are done, and recommendation method id run, and the result are displayed to the users.

3.4 Tools and Formulas used for development and calculation

3.4.1 Tools used

The tools and packages that was used for the development are listed below:

- ❖ **JuPyter Lab Notebook**
- ➔ JupyterLab is an interactive development environment for working with code, data, and notebooks. Most importantly, JupyterLab offers complete support for Jupyter notebooks. JupyterLab also lets to use text editors, terminals, data file viewers, and other custom components alongside notebooks in a tabbed workspace. In JupyterLab, notebooks, papers, and activities are all seamlessly interwoven. JupyterLab has more than 2,000 Python packages available. The platform Jupyter has shown to be incredibly dependable and user-friendly.

3.4.2 Libraries Used

The precise syntax for the Python programming language is provided by the Python libraries. The standard Python distribution comes packaged packages for these libraries. The use of Python is made simpler by the fact that each of these functions handles I/O operations and is written in C. Three libraries from Python were imported to finish this coursework; they are listed below.

- ❖ **NumPy**
- ➔ NumPy package will be used for making NumPy arrays which will help in handling the data in an easy format. The fundamental computational programming techniques are included in the NumPy package. The package offers simple computing methods as well as methods for array and linear algebra objects.

- **Pandas**

➔ For working with "relational" or "labelled" data, Panda's is a Python package that offers rapid, adaptable, and expressive data structures. Its purpose is to provide a solid foundation for conducting accurate, real-world data analysis in Python. It additionally aims to be the most robust and flexible open-source data analysis and manipulation tool available in any language.

- **Seaborn**

➔ With the use of files and datasets from files, the Seaborn program, a Python visualization package built on matplotlib, enables the creation of high-level graphs.

3.4.3 Formula used

There are many places where the calculation must be done for this project. The very first formula that will be used is Euclidean distance formula for calculating the set of similar users who likes the same kinds of courses. The formula for calculating Euclidean distance is displayed below:

$$d(p\ ,q) = d(q\ ,p) = \sqrt{(p1 - q1)^2 + (p2 - q2)^2 \ldots . + (pn - qn)^2}$$

Equation 1: Euclidean distance formula

From the Euclidean distance formula, the list of similar users will be found who likes the same kind of courses, After the similar user are found the similarity score of each user is calculated using the similarity formula. In the equation given below the S(p, q) is a similarity score of user p and user q and the d(p, q) denotes the distance between the user p and q in the graph.

$$S(p\ ,q) = \frac{1}{1 + \mathrm{d(p\ ,q)}}$$

Equation 2: Similarity score formula

3.5 Pseudocode

Sometimes a stage in the process of creating a program is described in depth using pseudocode. It enables lead programmers or designers to define the concept in detail and gives programmers a thorough template for the subsequent step of writing code in a particular programming language. For this recommendation system a simple pseudocode is developed.

```
Import libraries and packages
Import excels .csv files
Read and merge the data
Clean the unwanted data
Group by the rating and book name and remove duplicate data
Add the total rating column
Splitting dataset into train, test, and validation datasets
Create a Pivot table for merged data
Create a KNN model and define algorithm
Def courseRecommendation(course name):
    Try:
            Identify CourseId using course name and pivot table
            Calculate distance of each course with selected course
            Printing the recommendation
    Except:
            Display error message
Display the course name to the user
Take input from the user for course name
Run Recommendation function with user input as parameter.
```

3.6 Diagrammatic representations of the solution

A flowchart is a diagram that shows how a system, computer algorithm, or process works. They are frequently used in many different fields to examine, organize, enhance, and convey frequently complex processes in simple, understandable diagrams. Flow charts, use rectangles, ovals, diamonds and potentially several other forms to describe the type of step, together with connecting arrows to define flow and sequence. The flow chart for the system is displayed below:

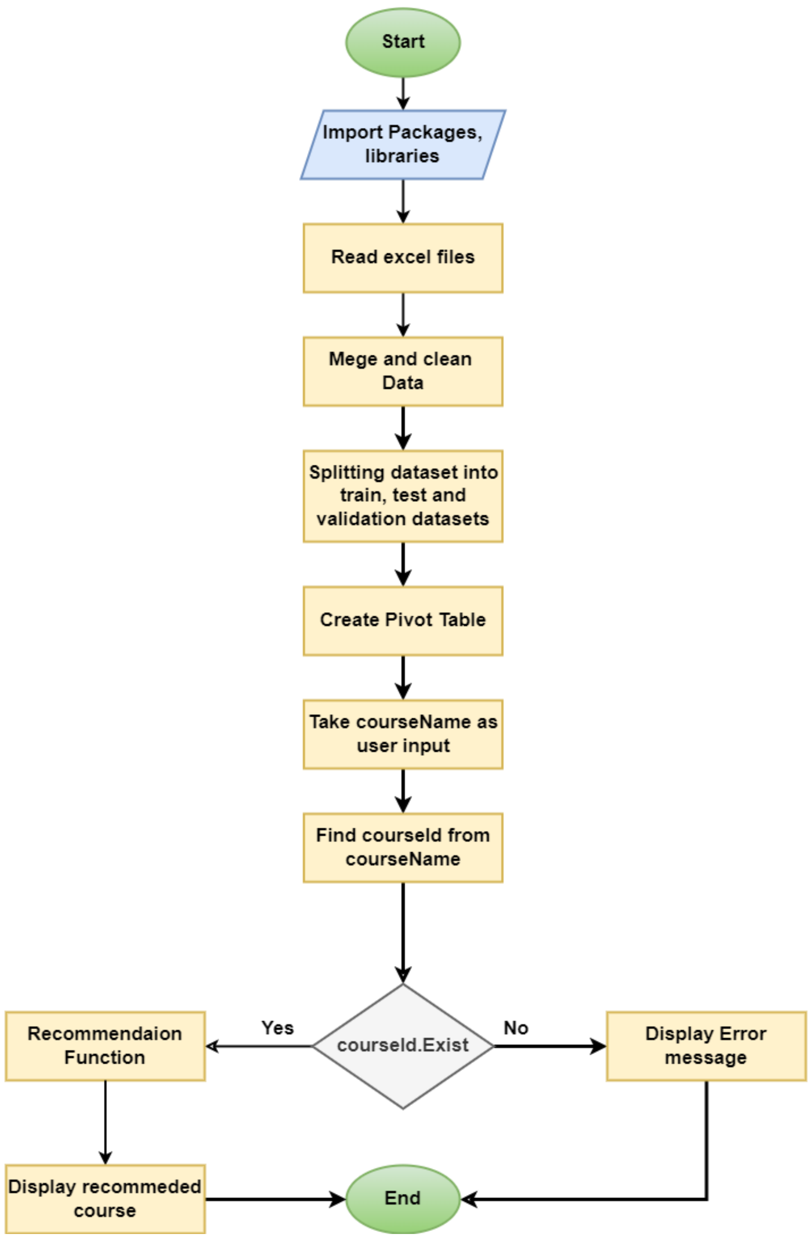

Figure 18: Flowchart for online course recommendation system

4 Conclusion

4.1 Analysis of the work done

While doing research for the online course recommendation system, many facts were uncovered. The analysis made it clear what was needed and what needed to be done still. Additionally, the research and publications clarified the fundamental principles and understanding of the course recommendation system. Through the research work, several difficulties about the algorithm and the best user recommendation approach were addressed, and the essential data and data structure were acquired through study for similar systems. The solution will also help to reduce the workload for the students and boost productivity on the eLearning platform.

4.2 How the solution addresses real world problems

Looking at the eLearning platform's graph, it appears that online learning has nearly supplanted traditional learning methods. So, the recommendation system is highly helpful for increasing the effectiveness of online learning. The student's precious time is more productive because it takes less time and effort to look for the course. Without a suggestion system, it would have been extremely challenging for users to finish their knowledge on specific topics. Therefore, the mechanism for recommending courses is a fantastic answer for eLearning sites.

4.3 Further work

The additional effort for this task entails writing the code and putting it into use on a few applications to verify if it functions or not. Additionally, the datasets are rectified if there are too few or too many. This is a simple prototype system that will be improved in accordance with demand. The final-year project that will be used to promote the course in the eLearning platform may also employ this system.

5 Bibliography and References

a) BALOGLU, O., LATIFI, S. Q. & NAZHA, A. 2022. What is machine learning? *Archives of Disease in Childhood-Education and Practice,* 107**,** 386-388.

b) CREMONESI, P. & JANNACH, D. 2021. Progress in recommender systems research: Crisis? What crisis? *AI Magazine,* 42**,** 43-54.

c) JANNACH, D., PU, P., RICCI, F. & ZANKER, M. 2021. Recommender systems: Past, present, future. *Ai Magazine,* 42**,** 3-6.

d) KHAN, B. H. & JOSHI, V. 2006. E-Learning Who, What and How? *Journal of Creative Communications,* 1**,** 61-74.

e) OBEIDAT, R., DUWAIRI, R. & AL-AIAD, A. A collaborative recommendation system for online courses recommendations. 2019 International Conference on Deep Learning and Machine Learning in Emerging Applications (Deep-ML), 2019. IEEE, 49-54.

f) REVATHI, A., KALYANI, D., RAMASUBBAREDDY, S. & GOVINDA, K. 2020. Critical review on course recommendation system with various similarities. *embedded systems and artificial intelligence***,** 843-852.

g) WANG, J., DE VRIES, A. P. & REINDERS, M. J. Unifying user-based and item-based collaborative filtering approaches by similarity fusion. Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval, 2006. 501-508.

h) ZHANG, H., HUANG, T., LV, Z., LIU, S. & ZHOU, Z. 2018. MCRS: A course recommendation system for MOOCs. *Multimedia Tools and Applications,* 77**,** 7051-7069.