CSE4077- RECOMMENDER SYSTEMS

J Component – Project Report

TITLE: *Coursera Course Recommendation System*

By

19MIA1008 -	P. Subhashri
19MIA1030 -	Deekshitha. L

M.Tech (Int.) CSE with Specialization in Business Analytics

Submitted to

Dr. A. Bhuvaneswari,
Assistant Professor Senior,SCOPE,
VIT, Chennai.

School of Computer Science and Engineering

VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

November 2022

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

BONAFIDE CERTIFICATE

Certified that this project report entitled “Course Recommendation System” is a bonafide work of P. Subhashri – 19MIA1008, Deekshitha. L – 19MIA1030 who carried out the J-component under my supervision and guidance. The contents of this Project work, in full or in parts, have neither been taken from any other source nor have been submitted to any other Institute or University for award of any degree or diploma and the same is certified.

Dr.A.Bhuvaneswari,

Assistant Professor Senior,

SCOPE, VIT, Chennai

ACKNOWLEDGEMENT

We wish to express our sincere thanks and deep sense of gratitude to our project guide, **Dr. A. Bhuvaneswari Assistant** Professor, School of Computer Science Engineering, for his consistent encouragement and valuable guidance offered to us in a pleasant manner throughout the course of the project work.

We express our thanks to our HOD Dr. Siva Balakrishnan for his support throughout the course of this project.

We also take this opportunity to thank all the faculty of the school for their support and their wisdom imparted to us throughout the course.

We thank our parents, family, and friends for bearing with us throughout the course of our project and for the opportunity they provided us in undergoing this course in such a prestigious institution.

P. Subhashri – 19MIA1008

Deekshitha. L – 19MIA1030

School of Computer Science and Engineering

VIT Chennai

Vandalur - Kelambakkam Road, Chennai - 600 127FALL SEM 22-23

Worklet details

Programme	M.Tech (Integrated) Computer Science and Engineering with Specialization in Business Analytics	
Course Name / Code	RECOMMENDER SYSTEMS (CSE4077)	
Slot	E1	
Faculty Name	Dr. A. Bhuvaneswari	
Component	J – Component	
J Component Title	Course recommendation system	
Team Members Name | Reg. No	P Subhashri	19MIA1008
	Deekshitha L	19MIA1030

Team Members(s) Contributions – Tentatively planned for implementation:

Worklet Tasks	*Contributor's Names*
Data collection and extraction	Deekshitha L
Preprocessing	Subhashri P
Data Visualization	Subhashri P
Model building	Deekshitha L (Tf-idf) Subhashri P (CountVectorizer)
Model evaluation	Deekshitha L
Recommendation	Deekshitha L & Subhashri P

TABLE OF CONTENTS

ABSTRACT

Technology Enhanced Learning (TEL) introduces the use of technology for the learning purposes. Exploration of the possibilities of TEL led to the development of many solutions and recently to Massive Open Online Courses (MOOCs). MOOCs are capable of providing severalten thousand of learners with access to courses over the web. MOOCs have recently gained much attention especially in leading universities and are now often considered as a highly promising form of teaching. In recent years, MOOCs have gained popularity with learners and providers, and thus MOOC providers have started to further enhance the use of MOOCs through recommender systems.

Since information retrieval and searching for the appropriate learning resources is an essential activity in TEL, the development of recommender systems for learning has seen increased attention. In this work, we address this major problem – the difficulty for learners to find courses which best fit their personal interests. We propose a system that recommends appropriate course from Coursera in response to a specific request of the learner. Using the Content - Based filtering methods and a special retrieval information technique, the system proposes to the learners the most appropriate courses fitting her/his request based on learner profile, needs and knowledge. Thus, users will not feel tired while perceiving information of their interest and will keep engaged and interested to use the system as it works upon the interest and likes of the user.

INTRODUCTION

In recent years, the emergence of online education platforms and massive open online courses (MOOCs) has attracted widespread interest. The establishment of various MOOCs platforms, including XuetangX, Chinese University MOOC, and Coursera, provides convenient education for more than 100 million users around the world, and provides a low-cost opportunity to access excellent courses in many top universities. Due to its convenience and abundance of teaching resources, online learning has gradually become a common way of learning. Especially due to the influence of COVID-19 in recent two years, traditional teaching methods have become difficult to implement in many places. The rapid development of online learning has changed the traditional teaching mode and made people study anytime and anywhere a reality. Online learning has become an important way for people to learn knowledge, expand their skills, and conduct academic research.

However, while online learning brings many conveniences, it also leads to the increasingly serious problem of information overload. Due to the rapid growth of the number of educational resources, it has gradually become difficult for people to choose suitable courses for learning. The recommender system is an effective means to solve the information overload. Given the historical interaction data of users and items, the recommender system can effectively capture hidden information such as the user’s personalized preferences and item attributes based on these data, so as to obtain information from appropriate materials filtered out from the massive resources and displayed to users. At present, recommendation systems have been widely used in the fields of e-commerce and social media platforms, and have become an important part of many portals, playing an increasingly important role.

LITERATURE SURVEY

S.NO	*TITLE*	*AUTHORS*	*METHOD*	*RESULT*
1.	A Recommendation System for Online Courses	David Estrela, Sérgio Batista, Diogo Martinho, and Goreti Marreiros	Content based, Collaborative based, Hybrid	Works well accordingly but has few drawbacks like not able to recommend immediately after being registered since there is no sufficient data.
2.	A Course Recommender System Based on Graduating Attributes	Behdad Bankshinategh, Gerasimos Spanakis, Osmar Zaiane and Samira ElAtia	Graduating Attributes	Provides further avenues of improving and experimenting
3.	Online course recommendation System	Supreeth S Avadhani , Siddharth Somani , Vaibhav Nayak , Sudhanva.BS	KNN, K-means and Collaborative Filtering methods	Results with the point that Collaborative filtering is not an effective method.
4.	A review on Recommender Systems for course selection in higher education	N D Lynn and A W R Emanuel	Hybrid, Knowledge based, Content based, Collaborative based.	Hybrid approach is concluded as the best one.

S.NO	***TITLE***	***AUTHORS***	***METHOD***	***RESULT***
5.	A Recommender System for the Massive Open Online Courses (MOOC)	Henda Chorfi Ouertani , Monerah Mohammed Alawadh	Content based methods	
6.	MOOC-Rec: A case based recommender system for MOOCs	Bousbahi, Fatiha & Chorfi, Henda.	Case-based recommenders (CBR) are a form of content-based recommendation. To ensure that the most similar cases are retrieved, the retrieval algorithm computes similarity bounds.	A binary tree is used to split the case library into groups of cases in such a way that each group contains cases that are similar to each other. The data base does not need to be excessively large, since we need only enough features to search for similar cases.
7.	Recommender System in eLearning: A survey(2020)	Pradnya V. Kulkarni, Sunil Rai, Rohini Kale	Three level hidden Bayesian linkPrediction (3-HBP) model - Collaborative DeepLearning	Convolutional Neural Network (CNN) is one of the alternatives for Collaborative Deep Learning (CDL)

DATASET AND TOOLS

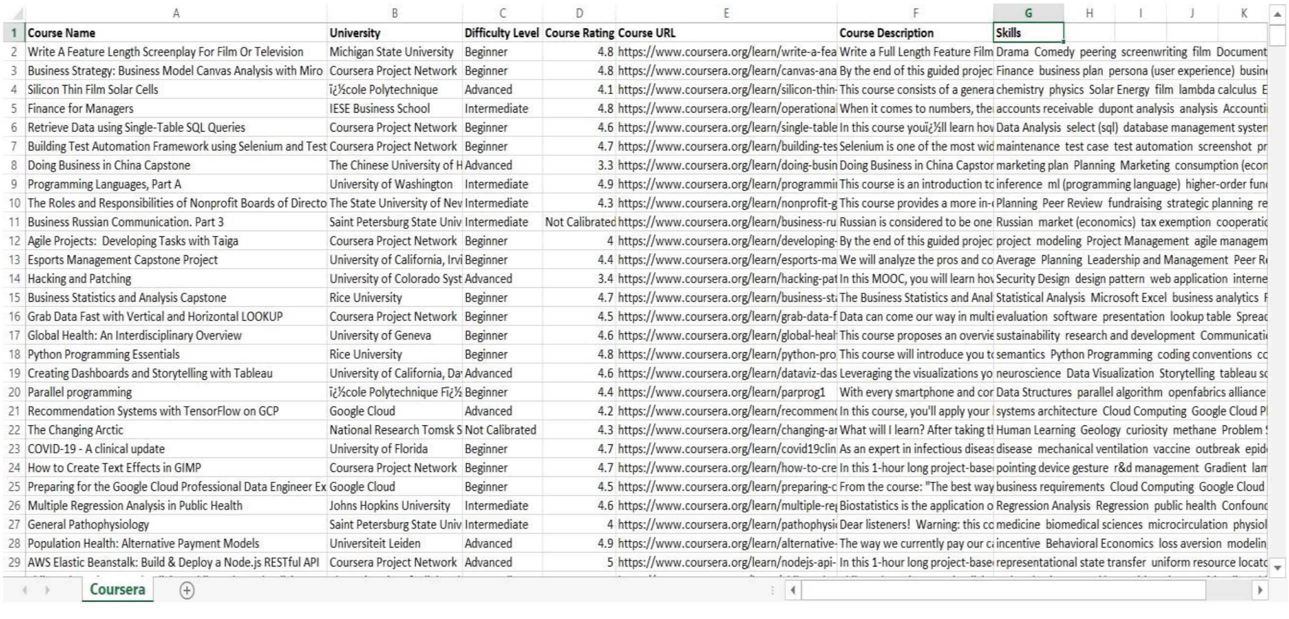
The Dataset is taken from Kaggle repository. This dataset was scraped off the publiclyavailable information on the Coursera website in September 2021 and manually enteredin the case where the data was improperly scraped. It can be used in Recommender Systems to promote Coursera courses based on the Difficulty Level and the Skills needed.

Its size is about 2MB. This contains 7 columns in it. Those attributes are:

1. Course Name
2. University
3. Difficulty Level
4. Course Rating
5. Course URL
6. Course Description
7. Skills

The tool used to carry forward the implementation of the project is Google Collab.

COURSERA DATASE



	A	B	C	D	E	F	G	H	I	J	K
1	Course Name	University	Difficulty Level	Course Rating	Course URL	Course Description	Skills				
2	Write A Feature Length Screenplay For Film Or Television	Michigan State University	Beginner	4.8	https://www.coursera.org/learn/write-a-fea	Write a Full Length Feature Film	Drama Comedy peering screenwriting film Document				
3	Business Strategy: Business Model Canvas Analysis with Miro	Coursera Project Network	Beginner	4.8	https://www.coursera.org/learn/canvas-ana	By the end of this guided projec	Finance business plan persona (user experience) busin				
4	Silicon Thin Film Solar Cells	ï¿½cole Polytechnique	Advanced	4.1	https://www.coursera.org/learn/silicon-thin-	This course consists of a genera	chemistry physics Solar Energy film lambda calculus E				
5	Finance for Managers	IESE Business School	Intermediate	4.8	https://www.coursera.org/learn/operational	When it comes to numbers, the	accounts receivable dupont analysis analysis Accounti				
6	Retrieve Data using Single-Table SQL Queries	Coursera Project Network	Beginner	4.6	https://www.coursera.org/learn/single-table	In this course youï¿½ll learn hov	Data Analysis select (sql) database management systen				
7	Building Test Automation Framework using Selenium and Test	Coursera Project Network	Beginner	4.7	https://www.coursera.org/learn/building-tes	Selenium is one of the most wid	maintenance test case test automation screenshot pr				
8	Doing Business in China Capstone	The Chinese University of H	Advanced	3.3	https://www.coursera.org/learn/doing-busin	Doing Business in China Capstor	marketing plan Planning Marketing consumption (econ				
9	Programming Languages, Part A	University of Washington	Intermediate	4.9	https://www.coursera.org/learn/programmi	This course is an introduction tc	inference ml (programming language) higher-order fun				
10	The Roles and Responsibilities of Nonprofit Boards of Directo	The State University of Nev	Intermediate	4.3	https://www.coursera.org/learn/nonprofit-g	This course provides a more in-(	Planning Peer Review fundraising strategic planning re				
11	Business Russian Communication. Part 3	Saint Petersburg State Univ	Intermediate	Not Calibrated	https://www.coursera.org/learn/business-ru	Russian is considered to be one	Russian market (economics) tax exemption cooperatic				
12	Agile Projects: Developing Tasks with Taiga	Coursera Project Network	Beginner	4	https://www.coursera.org/learn/developing-	By the end of this guided projec	project modeling Project Management agile managem				
13	Esports Management Capstone Project	University of California, Irvi	Beginner	4.4	https://www.coursera.org/learn/esports-ma	We will analyze the pros and co	Average Planning Leadership and Management Peer R				
14	Hacking and Patching	University of Colorado Syst	Advanced	3.4	https://www.coursera.org/learn/hacking-pat	In this MOOC, you will learn hov	Security Design design pattern web application interne				
15	Business Statistics and Analysis Capstone	Rice University	Beginner	4.7	https://www.coursera.org/learn/business-st	The Business Statistics and Anal	Statistical Analysis Microsoft Excel business analytics F				
16	Grab Data Fast with Vertical and Horizontal LOOKUP	Coursera Project Network	Beginner	4.5	https://www.coursera.org/learn/grab-data-f	Data can come our way in multi	evaluation software presentation lookup table Spreac				
17	Global Health: An Interdisciplinary Overview	University of Geneva	Beginner	4.6	https://www.coursera.org/learn/global-heal	This course proposes an overvie	sustainability research and development Communicati				
18	Python Programming Essentials	Rice University	Beginner	4.8	https://www.coursera.org/learn/python-pro	This course will introduce you tc	semantics Python Programming coding conventions cc				
19	Creating Dashboards and Storytelling with Tableau	University of California, Da	Advanced	4.6	https://www.coursera.org/learn/dataviz-das	Leveraging the visualizations yo	neuroscience Data Visualization Storytelling tableau sc				
20	Parallel programming	ï¿½cole Polytechnique Fï¿½	Beginner	4.4	https://www.coursera.org/learn/parprog1	With every smartphone and cor	Data Structures parallel algorithm openfabrics alliance				
21	Recommendation Systems with TensorFlow on GCP	Google Cloud	Advanced	4.2	https://www.coursera.org/learn/recommenc	In this course, you'll apply your	systems architecture Cloud Computing Google Cloud Pl				
22	The Changing Arctic	National Research Tomsk S	Not Calibrated	4.3	https://www.coursera.org/learn/changing-ar	What will I learn? After taking tl	Human Learning Geology curiosity methane Problem S				
23	COVID-19 - A clinical update	University of Florida	Beginner	4.7	https://www.coursera.org/learn/covid19clin	As an expert in infectious diseas	disease mechanical ventilation vaccine outbreak epid				
24	How to Create Text Effects in GIMP	Coursera Project Network	Beginner	4.7	https://www.coursera.org/learn/how-to-cre	In this 1-hour long project-base	pointing device gesture r&d management Gradient lam				
25	Preparing for the Google Cloud Professional Data Engineer Ex	Google Cloud	Beginner	4.5	https://www.coursera.org/learn/preparing-c	From the course: "The best way	business requirements Cloud Computing Google Cloud				
26	Multiple Regression Analysis in Public Health	Johns Hopkins University	Intermediate	4.6	https://www.coursera.org/learn/multiple-re	Biostatistics is the application o	Regression Analysis Regression public health Confounc				
27	General Pathophysiology	Saint Petersburg State Univ	Intermediate	4	https://www.coursera.org/learn/pathophysi	Dear listeners! Warning: this cc	medicine biomedical sciences microcirculation physiol				
28	Population Health: Alternative Payment Models	Universiteit Leiden	Advanced	4.9	https://www.coursera.org/learn/alternative-	The way we currently pay our c	incentive Behavioral Economics loss aversion modelin				
29	AWS Elastic Beanstalk: Build & Deploy a Node.js RESTful API	Coursera Project Network	Advanced	5	https://www.coursera.org/learn/nodejs-api-	In this 1-hour long project-base	representational state transfer uniform resource locatc				

PROPOSED METHODOLOGY

A content-based course recommendation system will recommend courses related to the previous experiences of its users, according to their preferences and needs by calculating the similarity measures. Content-based similarity filtering based on the course tags which the users either watch or search is being used. Through Data Exploration and Visualization, we planned to extract the required data alone for the further process of work. The important attributes to be used for model building are Course Title, Course Description, Difficulty level, Skills. Then we started with NLP model for building the recommendation engine using TF-IDF vectorizer. This model is built in such a way that, when you search for a course using a description (tags), the recommender engine recommenders us the Top N related and similar Courses' Title. We created the TAGS column, where this tags column is the combination of the following columns: Course Name, Difficulty Level, Course Description, Skills. The NLP process includes data preprocessing. An important part of the process is to pre-process the data into usable format for the recommendation system. We removed the spaces between the words.

In the later part of implementing Count Vectorization, A data frame was created which contains only the Course Name and Tags. After this process, we get the recommendation engine which recommends the relevant courses for us.

ALGORITHMS USED

- Term Frequency Inverse Document Frequency (TF-IDF)

TF-IDF (term frequency-inverse document frequency) is a statistical measure that evaluates how relevant a word is to a document in a collection of documents. TF-IDF for a word in a document is calculated by multiplying two different metrics:

The term-frequency (tf) of a word in a document: There are several ways of calculating this frequency, with the simplest being a raw count of instances a word appears in a document. Then, there are ways to adjust the frequency, by length of a document, or by the raw frequency of the most frequent word in a document.

The inverse document frequency (idf) of the word across a set of documents. IDF shows how common or rare a word is in the entire document set. The closer it is to 0, the more common a word is. This metric can be calculated by taking the total number of documents, dividing it by the number of documents that contain a word, and calculating the logarithm.

So, if the word is very common and appears in many documents, this number will approach 0. Otherwise, it will approach 1.

Multiplying these two numbers of results in the TF-IDF score of a word in a document. The higher the score, the more relevant that word is in that particular document.

- Count Vectorizer

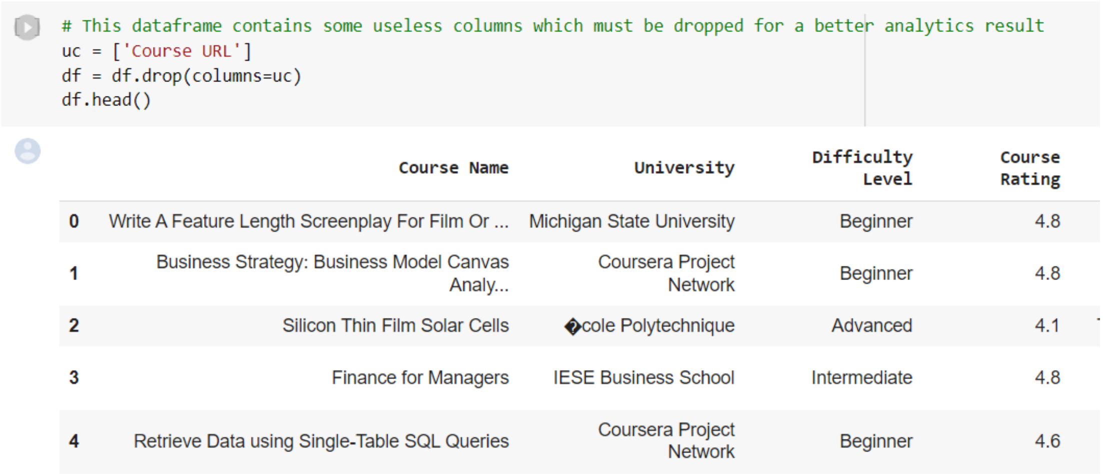
Count Vectorizer means breaking down a sentence or any text into Words by Performing preprocessing tasks like converting all words to lowercase, thus removing special characters. In NLP Models can’t understand textual data, they only accept numbers, so this textual data needs to be vectorized.

- Cosine Similarity

  Cosine similarity is a metric, helpful in determining, how similar the data objects are irrespective of their size. It ranges from -1 to 1. 1 indicates the items are the same whereas -1 represents the compared items are dissimilar. Cosine similarity is independent of the magnitude or the size of the vectors. The cosine similarity is beneficial because even if the two similar data objects are far apart by the Euclidean distance because of the size, they could still have a smaller angle between them. Smaller the angle, higher the similarity.

EXPERIMENTAL RESULTS

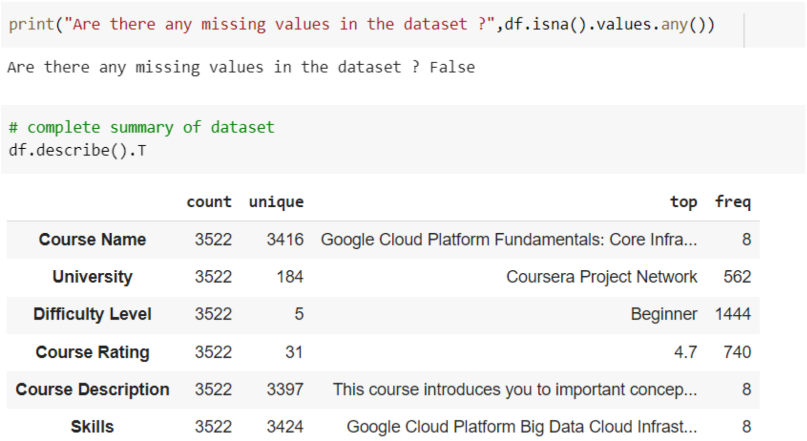
- ## DATA CLEANING



```
# This dataframe contains some useless columns which must be dropped for a better analytics result
uc = ['Course URL']
df = df.drop(columns=uc)
df.head()
```

	Course Name	University	Difficulty Level	Course Rating
0	Write A Feature Length Screenplay For Film Or ...	Michigan State University	Beginner	4.8
1	Business Strategy: Business Model Canvas Analy...	Coursera Project Network	Beginner	4.8
2	Silicon Thin Film Solar Cells	�cole Polytechnique	Advanced	4.1
3	Finance for Managers	IESE Business School	Intermediate	4.8
4	Retrieve Data using Single-Table SQL Queries	Coursera Project Network	Beginner	4.6

- ## REMOVING NULL VALUES



```
print("Are there any missing values in the dataset ?",df.isna().values.any())
```

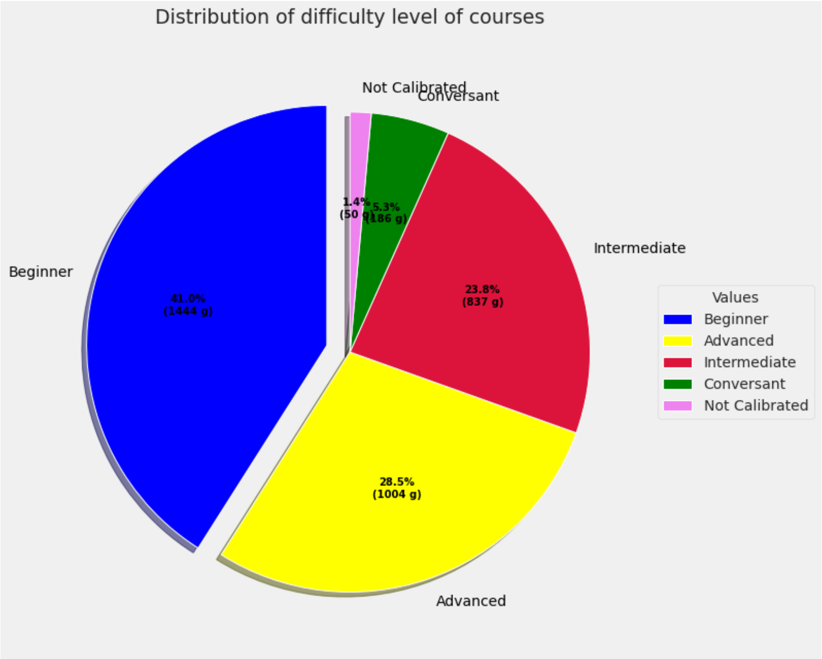
```
Are there any missing values in the dataset ? False
```

```
# complete summary of dataset
df.describe().T
```

	count	unique	top	freq
Course Name	3522	3416	Google Cloud Platform Fundamentals: Core Infra...	8
University	3522	184	Coursera Project Network	562
Difficulty Level	3522	5	Beginner	1444
Course Rating	3522	31	4.7	740
Course Description	3522	3397	This course introduces you to important concep...	8
Skills	3522	3424	Google Cloud Platform Big Data Cloud Infrast...	8

- <u>VISUALIZATION</u>

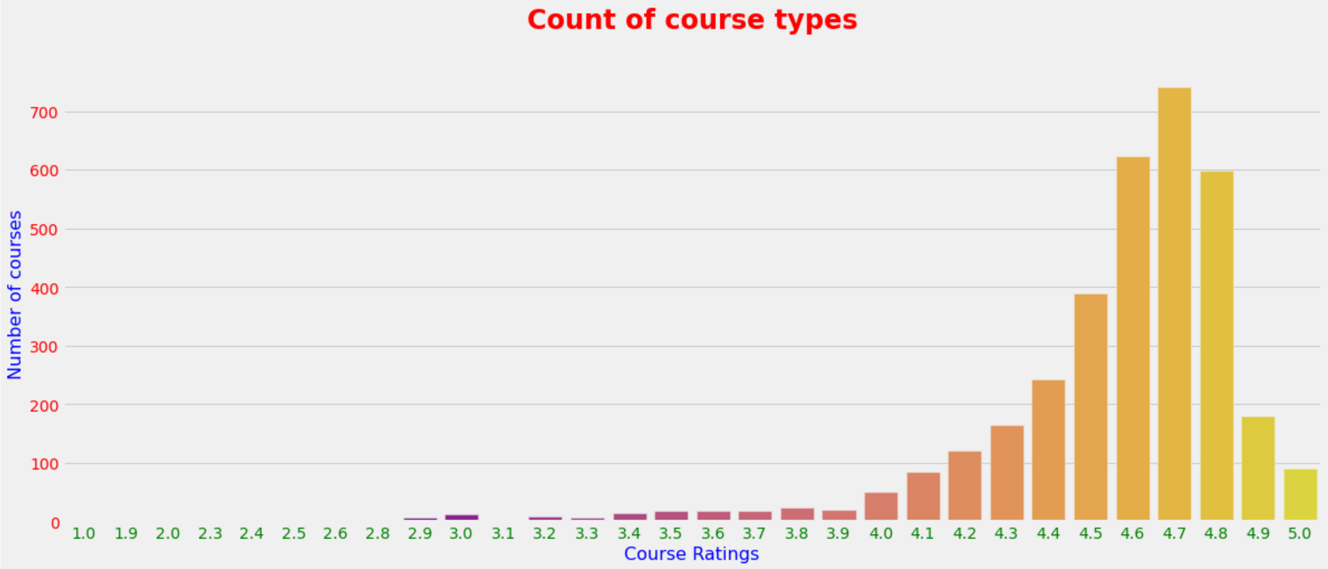
❖ Difficulty level of courses



We visualized the distribution of difficulty level of the courses using a Pie-chart. It states us the data weightage of the attribute's each value. It contains the types, Beginner, Advanced, Intermediate, Conversant and Not calibrated.

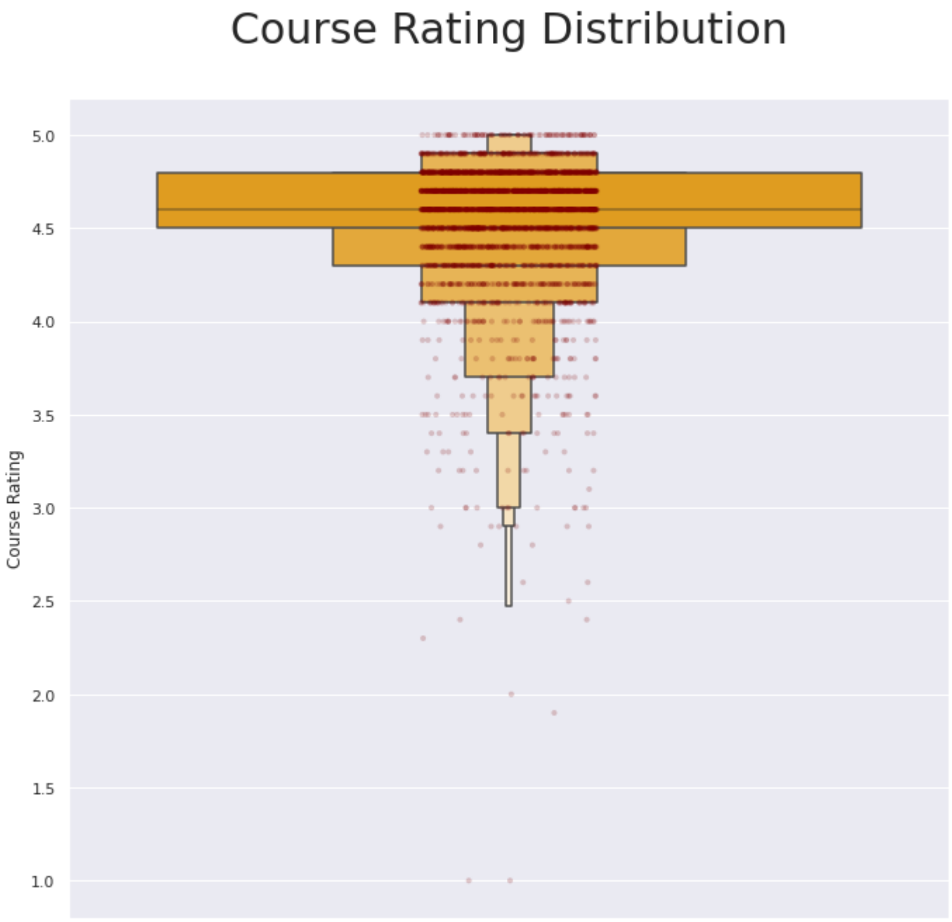
From this plot, we get to know that The Beginner level courses are available the most.

❖ Count of Course type

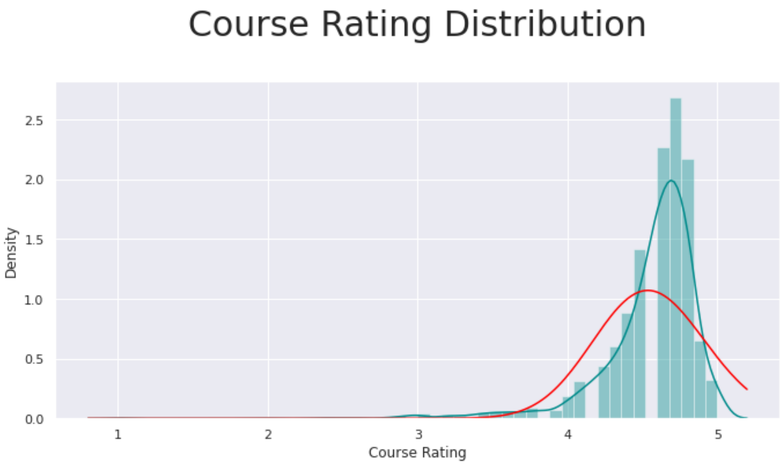


This plot shows us the count of every type of course. This plot is plotted against the Number of courses and the Course ratings.

- Course rating Distribution

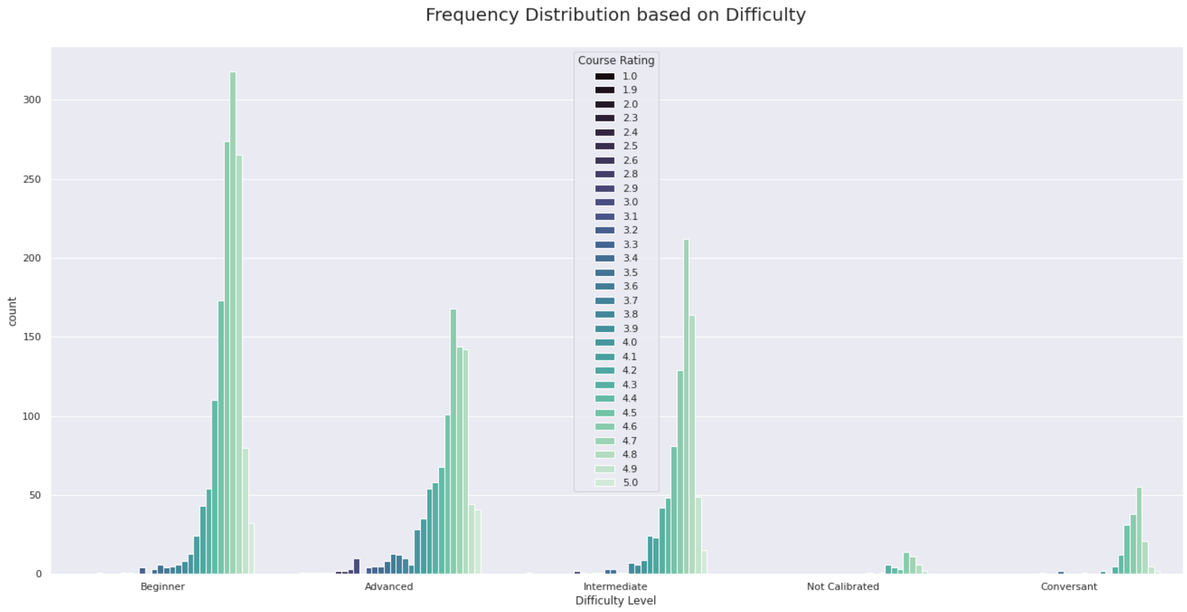


This Box plot shows us the range and distribution of the course ratings.

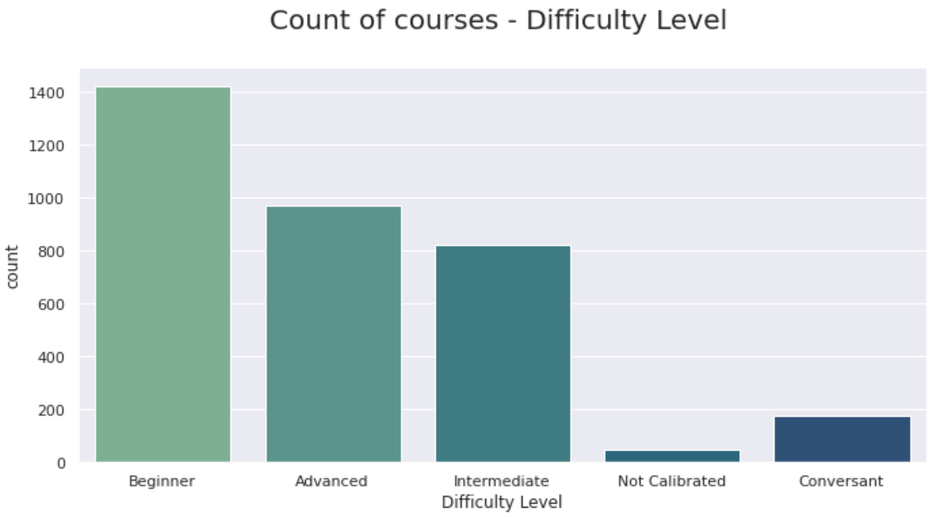


Again, this plot also plotted against Density and the course ratings to get the distribution of the ratings.

- ❖ Frequency based on Difficulty level



- ❖ Distribution per course type

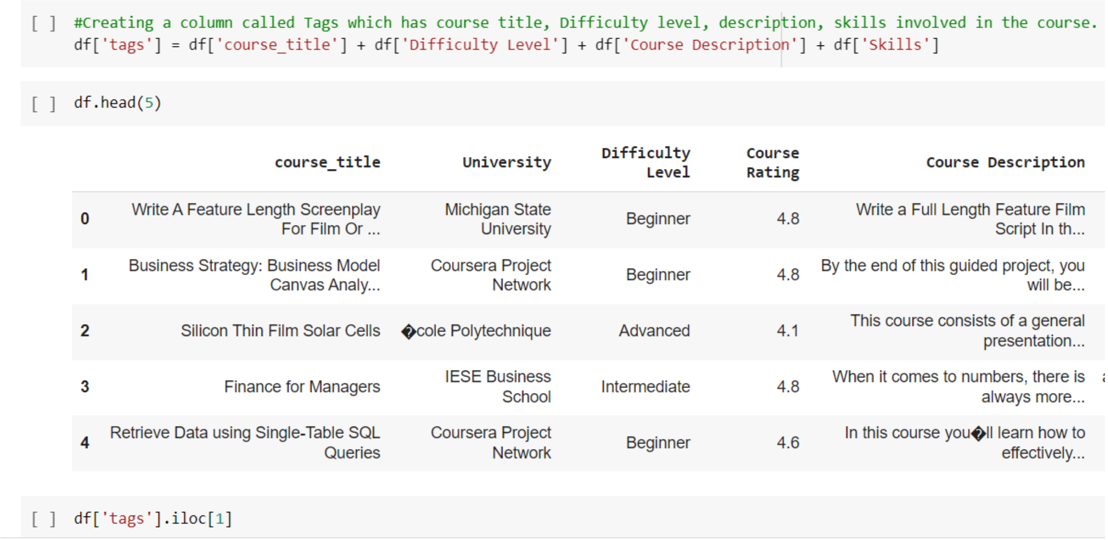


- Word cloud

Product
documents
Web
Management
presentation
story
Leadership
project
modeling
agile
strategy
Development
Design
user
business

pathophysiology
inflammations
biomedical
science
life
basic
physiology
biochemistry
internal
pathology
medicine
microcirculation
endocrinology

CREATING TAGS

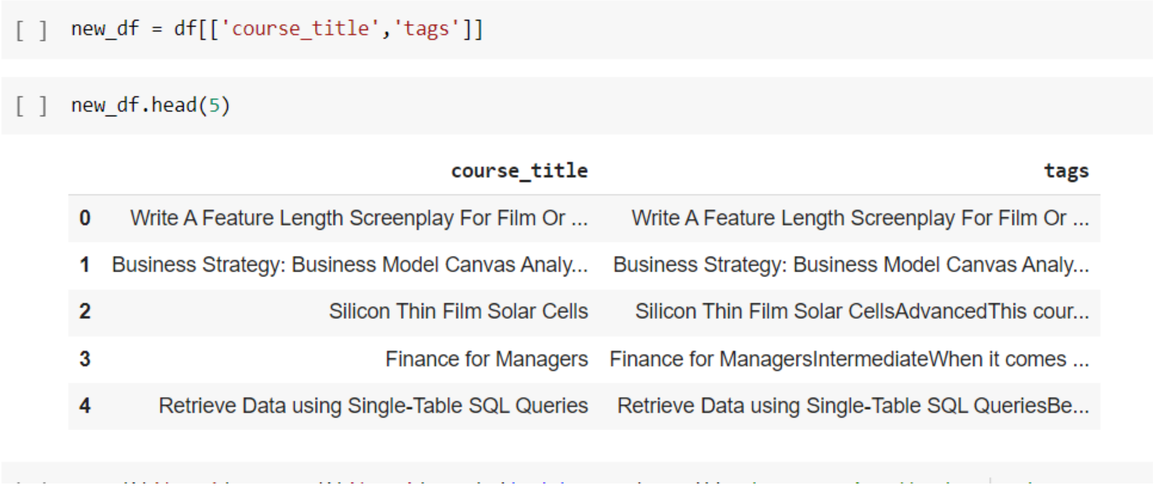


```
[ ] #Creating a column called Tags which has course title, Difficulty level, description, skills involved in the course.
    df['tags'] = df['course_title'] + df['Difficulty Level'] + df['Course Description'] + df['Skills']
```

```
[ ] df.head(5)
```

	course_title	University	Difficulty Level	Course Rating	Course Description
0	Write A Feature Length Screenplay For Film Or ...	Michigan State University	Beginner	4.8	Write a Full Length Feature Film Script In th...
1	Business Strategy: Business Model Canvas Analy...	Coursera Project Network	Beginner	4.8	By the end of this guided project, you will be...
2	Silicon Thin Film Solar Cells	�cole Polytechnique	Advanced	4.1	This course consists of a general presentation...
3	Finance for Managers	IESE Business School	Intermediate	4.8	When it comes to numbers, there is always more...
4	Retrieve Data using Single-Table SQL Queries	Coursera Project Network	Beginner	4.6	In this course you�ll learn how to effectively...

```
[ ] df['tags'].iloc[1]
```



```
[ ] new_df = df[['course_title','tags']]
```

```
[ ] new_df.head(5)
```

	course_title	tags
0	Write A Feature Length Screenplay For Film Or ...	Write A Feature Length Screenplay For Film Or ...
1	Business Strategy: Business Model Canvas Analy...	Business Strategy: Business Model Canvas Analy...
2	Silicon Thin Film Solar Cells	Silicon Thin Film Solar CellsAdvancedThis cour...
3	Finance for Managers	Finance for ManagersIntermediateWhen it comes ...
4	Retrieve Data using Single-Table SQL Queries	Retrieve Data using Single-Table SQL QueriesBe...

TEXT PREPROCESSING

Initially, before building the engine we did Text preprocessing using Stemming process.

Text preprocessing involves transforming text into a clean and consistent format that can then be fed into a model for further analysis and learning. Text preprocessing techniques may be general so that they are applicable to many types of applications, or they can be specialized for a specific task.

Some of the common text preprocessing / cleaning steps are:

1. Lower casing.
2. Removal of Frequent words.
3. Stemming

Stemming is the process of removing a part of a word or reducing a word to its stem or root. Stemming is important in natural language understanding (NLU) and natural language processing (NLP).

FEATURE EXTRACTION

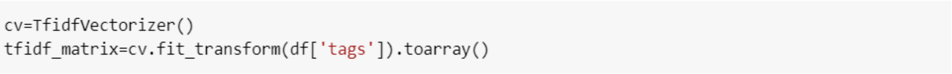
Feature extraction is a process of dimensionality reduction by which an initial set of raw data is reduced to more manageable groups for processing. Feature extraction is the name for methods that select and /or combine variables into features, effectively reducing the amount of data that must be processed, while still accurately and completely describing the original data set.

Some of the feature extraction techniques to convert text into a matrix (or vector) of features. Some of the most popular methods of feature extraction are:

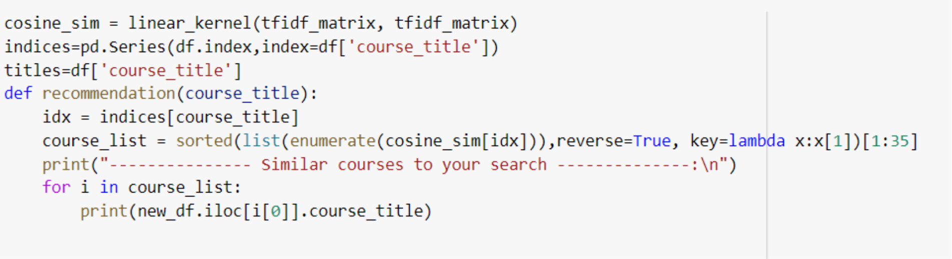
- Bag-of-Words
- TF – IDF

TF-IDF

TF-IDF (term frequency-inverse document frequency) is a statistical measure that evaluates how relevant a word is to a document in a collection of documents.

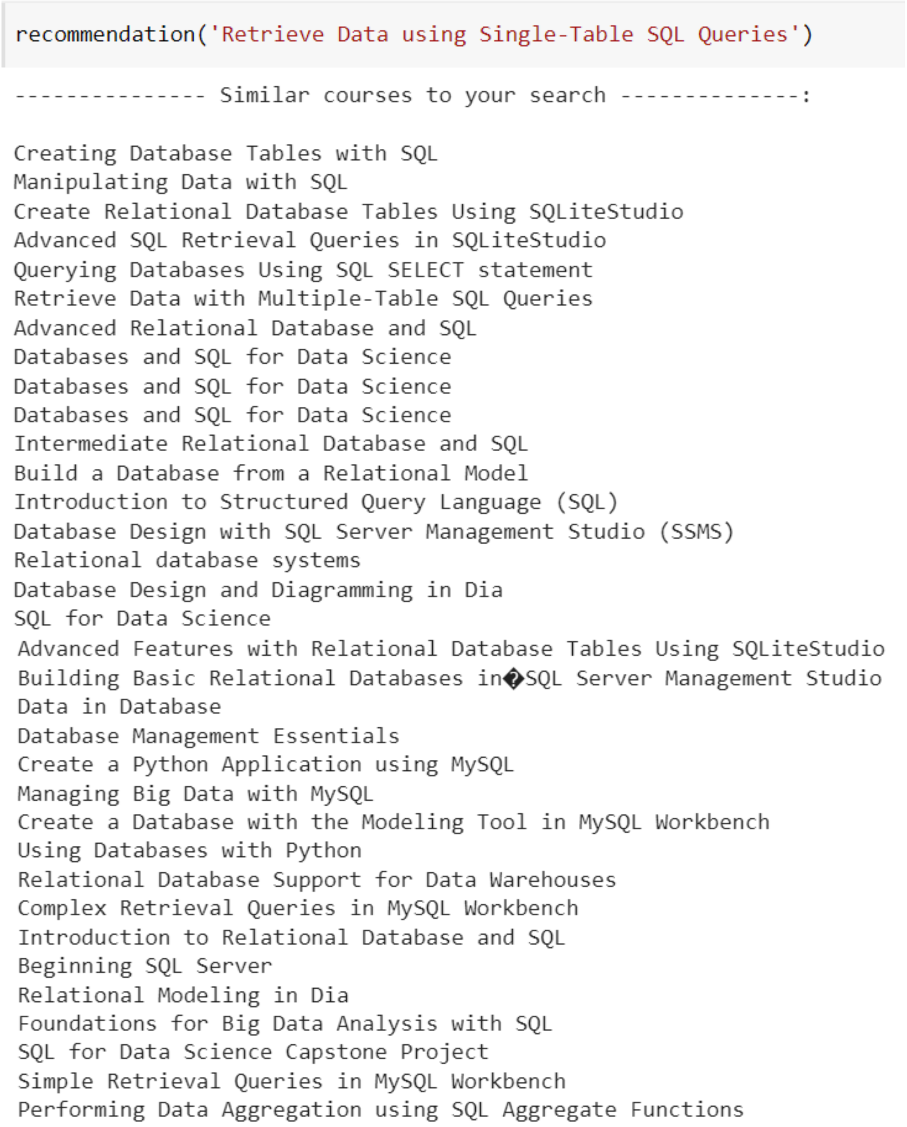


```
cv=TfidfVectorizer()
tfidf_matrix=cv.fit_transform(df['tags']).toarray()
```



```
cosine_sim = linear_kernel(tfidf_matrix, tfidf_matrix)
indices=pd.Series(df.index,index=df['course_title'])
titles=df['course_title']
def recommendation(course_title):
    idx = indices[course_title]
    course_list = sorted(list(enumerate(cosine_sim[idx])),reverse=True, key=lambda x:x[1])[1:35]
    print("--------------- Similar courses to your search --------------:\n")
    for i in course_list:
        print(new_df.iloc[i[0]].course_title)
```

RECOMMENDATION ENGINE



```
recommendation('Retrieve Data using Single-Table SQL Queries')
```

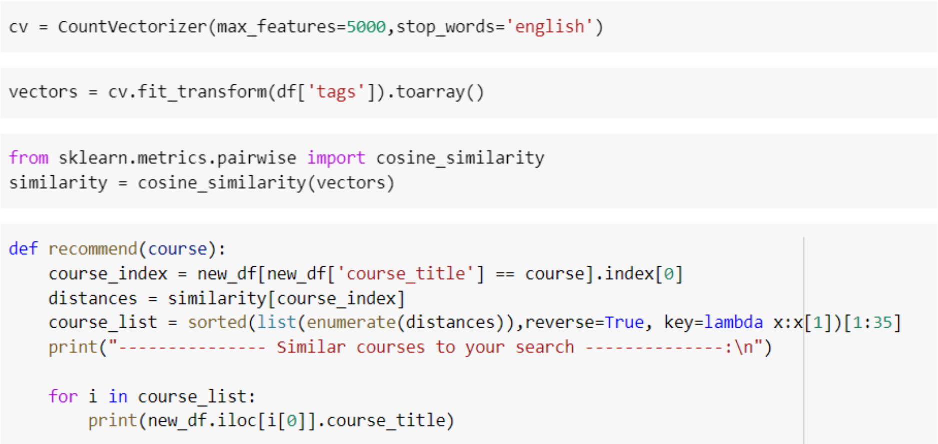
```
--------------- Similar courses to your search --------------:

Creating Database Tables with SQL
Manipulating Data with SQL
Create Relational Database Tables Using SQLiteStudio
Advanced SQL Retrieval Queries in SQLiteStudio
Querying Databases Using SQL SELECT statement
Retrieve Data with Multiple-Table SQL Queries
Advanced Relational Database and SQL
Databases and SQL for Data Science
Databases and SQL for Data Science
Databases and SQL for Data Science
Intermediate Relational Database and SQL
Build a Database from a Relational Model
Introduction to Structured Query Language (SQL)
Database Design with SQL Server Management Studio (SSMS)
Relational database systems
Database Design and Diagramming in Dia
SQL for Data Science
 Advanced Features with Relational Database Tables Using SQLiteStudio
 Building Basic Relational Databases in�SQL Server Management Studio
 Data in Database
 Database Management Essentials
 Create a Python Application using MySQL
 Managing Big Data with MySQL
 Create a Database with the Modeling Tool in MySQL Workbench
 Using Databases with Python
 Relational Database Support for Data Warehouses
 Complex Retrieval Queries in MySQL Workbench
 Introduction to Relational Database and SQL
 Beginning SQL Server
 Relational Modeling in Dia
 Foundations for Big Data Analysis with SQL
 SQL for Data Science Capstone Project
 Simple Retrieval Queries in MySQL Workbench
 Performing Data Aggregation using SQL Aggregate Functions
```

- Count Vectorizer – (Bag Of Words)

  A technique for natural language processing that extracts the words (features) used in a sentence, document, website, etc. and classifies them by frequency of use. This technique can also be applied to image processing.

Count Vectorizer



```
cv = CountVectorizer(max_features=5000,stop_words='english')
```

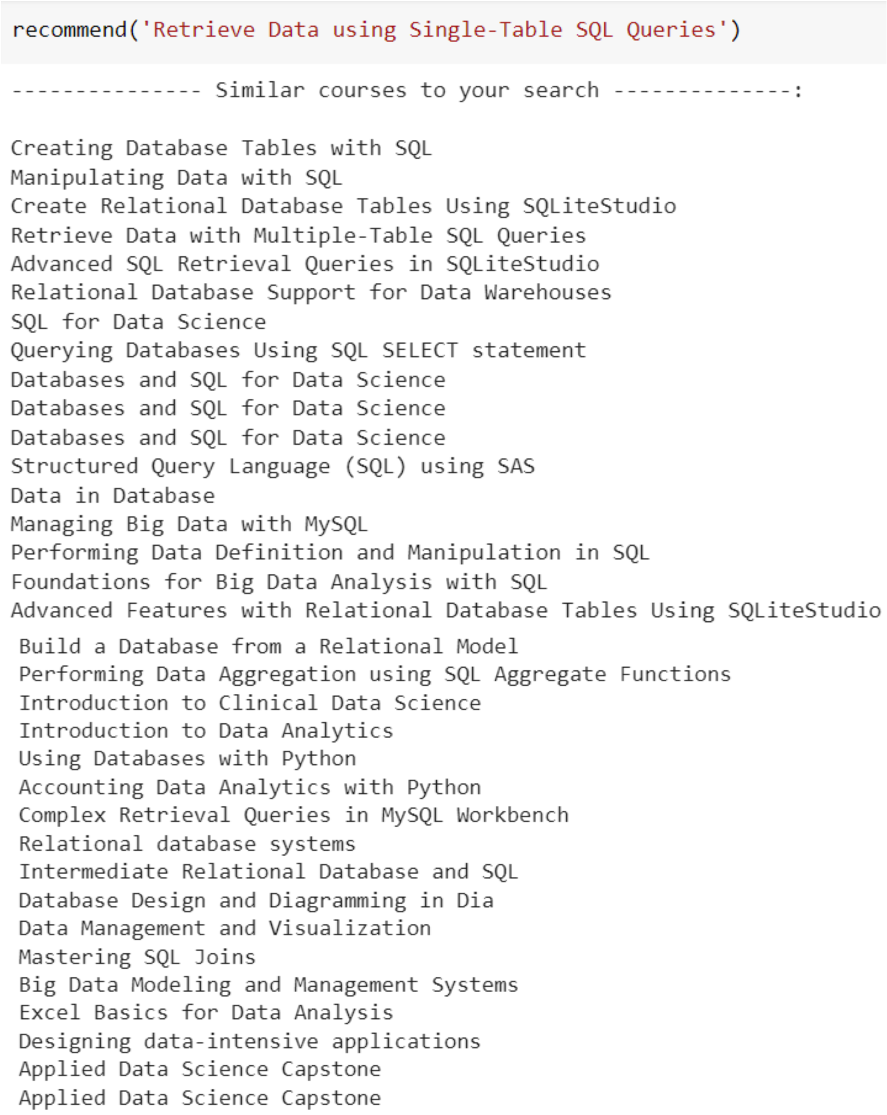
```
vectors = cv.fit_transform(df['tags']).toarray()
```

```
from sklearn.metrics.pairwise import cosine_similarity
similarity = cosine_similarity(vectors)
```

```
def recommend(course):
    course_index = new_df[new_df['course_title'] == course].index[0]
    distances = similarity[course_index]
    course_list = sorted(list(enumerate(distances)),reverse=True, key=lambda x:x[1])[1:35]
    print("-------------- Similar courses to your search -------------:\n")

    for i in course_list:
        print(new_df.iloc[i[0]].course_title)
```

RECOMMENDATION ENGINE - Recommending courses related to Database and SQL



```
recommend('Retrieve Data using Single-Table SQL Queries')
```

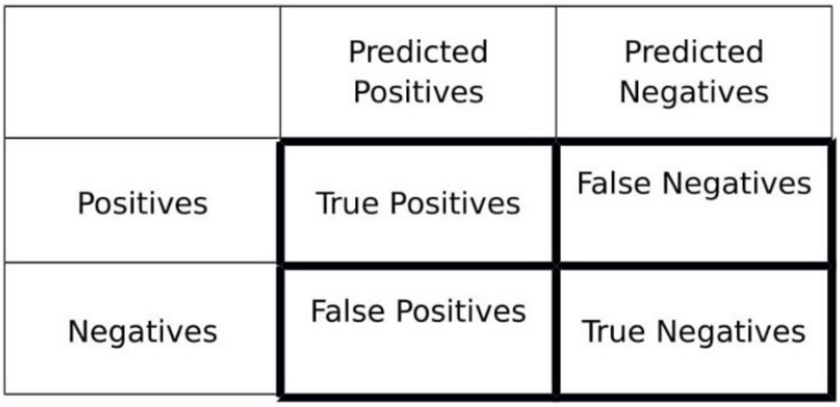
```
-------------- Similar courses to your search -------------:

Creating Database Tables with SQL
Manipulating Data with SQL
Create Relational Database Tables Using SQLiteStudio
Retrieve Data with Multiple-Table SQL Queries
Advanced SQL Retrieval Queries in SQLiteStudio
Relational Database Support for Data Warehouses
SQL for Data Science
Querying Databases Using SQL SELECT statement
Databases and SQL for Data Science
Databases and SQL for Data Science
Databases and SQL for Data Science
Structured Query Language (SQL) using SAS
Data in Database
Managing Big Data with MySQL
Performing Data Definition and Manipulation in SQL
Foundations for Big Data Analysis with SQL
Advanced Features with Relational Database Tables Using SQLiteStudio
 Build a Database from a Relational Model
 Performing Data Aggregation using SQL Aggregate Functions
 Introduction to Clinical Data Science
 Introduction to Data Analytics
 Using Databases with Python
 Accounting Data Analytics with Python
 Complex Retrieval Queries in MySQL Workbench
 Relational database systems
 Intermediate Relational Database and SQL
 Database Design and Diagramming in Dia
 Data Management and Visualization
 Mastering SQL Joins
 Big Data Modeling and Management Systems
 Excel Basics for Data Analysis
 Designing data-intensive applications
 Applied Data Science Capstone
 Applied Data Science Capstone
```

MODEL EVALUATION

We have evaluated the recommendations using one of the evaluation metrics called Precision. Precision is a metric that quantifies the number of correct positive predictions made. Precision, therefore, calculates the accuracy for the minority class. It is calculated as the ratio of correctly predicted positive examples divided by the total number of positive examples that were predicted. Precision is the percentage of your results which are relevant. It is a good measure to determine, when the cost of False Positive is high.

CONFUSION MATRIX:



	Predicted Positives	Predicted Negatives
Positives	True Positives	False Negatives
Negatives	False Positives	True Negatives

$$\textbf{Precision} = \frac{True\ Positives}{True\ Positives + False\ Positives}$$

DISCUSSION ON RESULTS

Precision evaluates how precise a model is in predicting positive labels. We calculated the value of precision manually by recommending Top 35 courses that are related to the courses searched by the user.

The precision values for our built models are as follows:

For 35 samples,

- Precision of Tf-Idf model = 0.91
- Precision of CountVectorisation = 0.73

CONCLUSION

TfidfVectorizer and CountVectorizer both are methods for converting text data into vectors as model can process only numerical data. In CountVectorizer, we only count the number of times a word appears in the document which results in biasing in favor of most frequent words.

On comparing the results of recommendations of two different models, TF-IDF model is performing better than Count Vectorizers because it not only focuses on the frequency of words present in the corpus but also provides the importance of the words. Thus we were able to recommend the related courses based on the content or description of the course.

REPOSITORY LINK

https://github.com/Deekshi2020/Recommender-Systems-project

REFERENCES:

- Salehi, M. and Kamalabadi, I.N. (2013) Hybrid Recommendation Approach for Learning Material Based on Sequential Pattern of The Accessed Material and The Learner's Preference Tree. Knowledge-BasedSystems, 48, 57-69.
- https://doi.org/10.1016/j.knosys.2013.04.012

- Lu, J. (2022) A Survey of Online Course Recommendation Techniques. *Open Journal of Applied Sciences*, **12**, 134-154. doi: 10.4236/ojapps.2022.121010.

- Bousbahi, Fatiha & Chorfi, Henda. (2015). MOOC-Rec: A case based recommender system for MOOCs. Procedia - Social and Behavioral Sciences. 195. 1813-1822. 10.1016/j.sbspro.2015.06.395.

- Ouertani, H.C., Alawadh, M.M. (2017). MOOCs Recommender System: A Recommender System for the Massive Open Online Courses. In: , *etal.* Innovations in Smart Learning. Lecture Notes in Educational Technology. Springer, Singapore. https://doi.org/10.1007/978-981-10- 2419-1_20

- Kulkarni, Pradnya & Rai, Sunil & Kale, Rohini. (2020). Recommender System in eLearning: A Survey. 10.1007/978-981-15-0790-8_13.

- https://www.irjet.net/archives/V6/i4/IRJET-V6I41117.pdf

- https://practicaldatascience.co.uk/data-science/how-to-create-content- recommendations-using-tf-idf

- https://www.analyticsvidhya.com/blog/2021/09/creating-a-movie- reviews-classifier-using-tf-idf-in-python/

- https://indiaai.gov.in/article/cosine-similarity-the-metric-behind- recommendation-systems