Machine Learning Capstone (IBM)

PROJECT REPORT SUBMITTED TO

MANIPAL
ACADEMY of HIGHER EDUCATION
(Institution of Eminence Deemed to be University)

in partial fulfilment for the award of the degree

of

MASTER OF SCIENCE

Data Science

Submitted by:

Prajval P

Reg. No: 22154090137

August, 2024

Contents

1. Introduction

The main goal of this project is to improve learners' learning experience via helping them quickly find new interested courses and better paving their learning paths. Meanwhile, with more learners interacting with more courses via the recommender systems, your company's revenue may also be increased.

The focus at this moment is to explore and compare various machine learning models and find one with the best performance in off-line evaluations.

The main tasks to be carried out in this project are summarized in the below workflow:

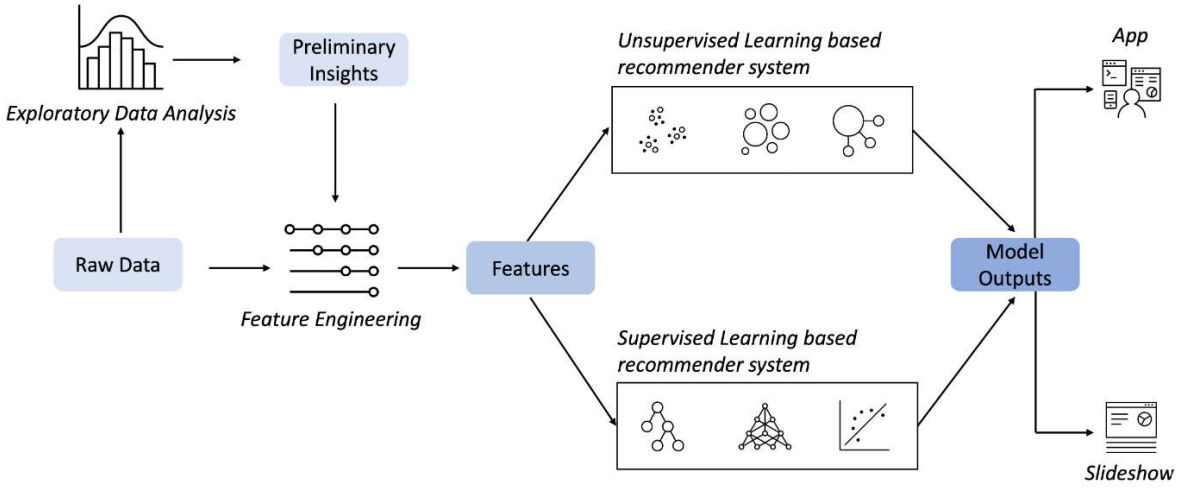

Fig1: Procedure to build Recommender Systems

- Collecting and understanding data
- Performing exploratory data analysis on online course enrollments datasets
- Extracting Bag of Words (BoW) features from course textual content
- Calculating course similarity using BoW features.
- Building content-based recommender systems using various unsupervised learning algorithms, such as:
    - Distance/Similarity measurements, K-means, Principal Component Analysis (PCA), etc.
- Building collaborative-filtering recommender systems using various supervised learning algorithms:
    - K Nearest Neighbors, Non-negative Matrix Factorization (NMF), Neural Networks, Linear Regression, Logistic Regression, Random Forest, etc.
- Creating an insightful and informative slideshow and presenting it.

2. Procedure

Techniques learnt and applied:

- Identify keywords in course titles using a WordCloud, calculate the summary statistics and visualizations of the online course content dataset, determine popular course genres, calculate the summary statistics and create visualizations of the online course enrollment dataset and identify courses with the greatest number of enrolled students
- Extract Bag of Words (BoW) features from course titles and descriptions and build a course BoW dataset to be used for building a content-based recommender system.
- Calculate the similarity between any two courses using BoW feature vectors.
- Generate a user profile based on course genres and rating and generate course recommendations based on a user's profile and course genres.
- Obtain the similarity between courses from a course similarity matrix and use the course similarity matrix to find and recommend new courses which are similar to enrolled courses.
- Perform k-means clustering on the original user profile feature vectors. Apply PCA (Principle Component Analysis) on user profile feature vectors to reduce dimensions. Perform k-means clustering on the PCA transformed main components and generate course recommendations based on other group members' enrolment history.
- Learned and implemented KNN-based collaborative filtering.
- Learned and practiced NMF-based collaborative filtering.
- Used tensorflow to train neural networks to extract the user and item latent features from the hidden layers and predict course ratings with trained neural networks
- Built regression models to predict ratings using the combined embedding vectors.
- Built classification models to predict rating modes using the combined embedding vectors.

Tools/ Skills:

- Python
- Google Colab
- Python Packages/ Libraries used:
    - Pandas
    - NumPy
    - Matplotlib
    - Seaborn
    - SKLearn
    - SciPy
    - WordCloud
    - NLTK
    - Gensim
    - Surprise
    - TensorFlow
    - Keras

3. Key Takeaways

Exploratory Data Analysis

- To begin with, we want to examine the variety of genres available in the courses.
- A bar chart was utilized to illustrate the distribution of genres across all courses.
- The dominant genres among the courses are backend development, machine learning, and databases.

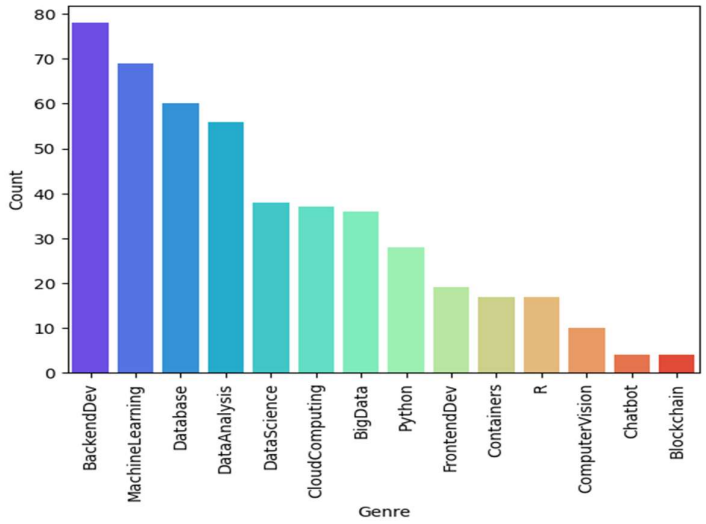

Fig2: Course Genre vs Course Count per Genre.

- Additionally, we can generate a histogram displaying the distribution of enrollments, such as the number of users who rated only one item or those who rated ten items, and so on.
- As a result, we observe that most of the enrollments fall within the range of 0 to 10.

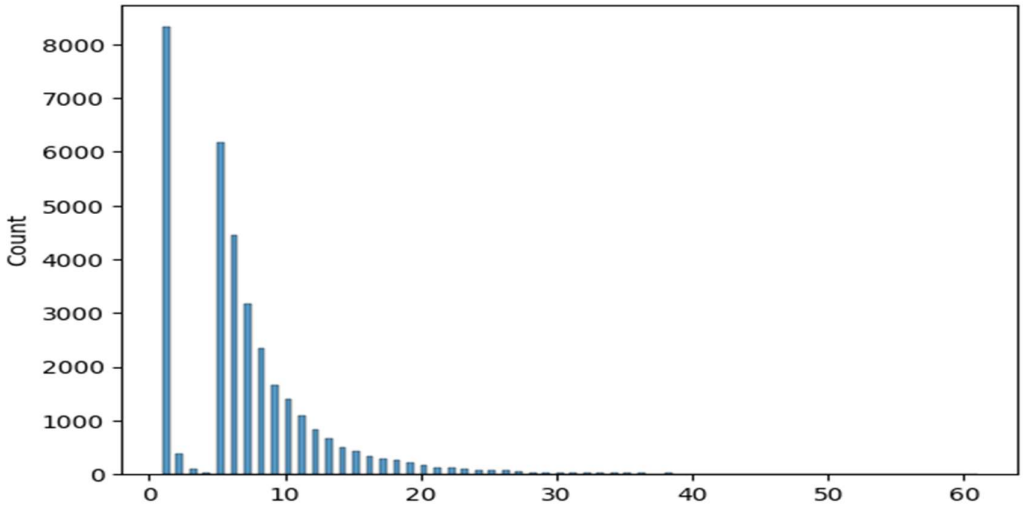

Fig3: Histogram of User Rating Counts.

- We tried to check out the 20 most popular courses of them all:
  - Based on the data, we can see that "PY0101EN" (Python for Data Science) holds the top spot in terms of ratings or enrolments, followed by "DS0101EN".
  - This indicates that courses related to data science are currently in high demand.
- A WordCloud was generated to help visualize the prevalent themes in the course titles.
- From the visualization, it is clear that Python, machine learning, and data science are the most prominent topics.

Recommender Systems

1. Content-based Recommender System using Unsupervised Learning

- Initially, we took the raw textual data from the course titles and descriptions and transformed it into numerical form for machine learning, allowing us to detect patterns. We applied the Bag of Words (BoW) method to compute the frequency of each unique word in the courses. Stopwords were removed to reduce dimensionality since they don't significantly impact the system.
- Using this BoW representation, we calculated course similarities by applying cosine distance to obtain similarity scores for each course, which were then used as a feature.
- Finally, we looped through each user's enrolled courses to identify similar ones and recommended them based on the calculated similarity scores.

Fig4: Flowchart of content-based recommender system using user profile and course genres

Results of user profile-based recommender system

- We set our threshold score for recommendation is 10
- On average, there are around 19 courses are recommended to users from this recommender system.(avg_course_recommended = 19.11)
- The below are the 10 most recommended courses across users:

COURSE ID	USER
TA0106EN	17390
excourse21	15656
excourse22	15656
GPXX0IBEN	15644
ML0122EN	15603
excourse04	15062
excourse06	15062
GPXX0TY1EN	14689
excourse73	14464
excourse72	14464

2. Content-based Recommender System using Course Similarity

- Next, we utilized the course similarity metric to recommend new courses that closely resemble a user's currently enrolled courses.
- Initially, we generated the Bag of Words (BoW) representation and indexed the courses for easy querying. Using course IDs, we can employ the id_idx_dict dictionary to locate the corresponding row and column indices in the similarity matrix.
- Then, we identified courses that were sufficiently like the ones users were already enrolled in.

Fig5: Flowchart of Content-based Recommender System using Course Similarity

Results of course similarity based recommender system

- If the similarity is larger than a threshold such as 0.5 or 0.6, then it is added to the user course recommendation list
- On average, there are around 8 to 9 courses are recommended to users from this recommender system. (average_course_recommended = 8.5465)
- The below are the 10 most recommended courses across users:

COURSE ID	USER
DS0110EN	15003
excourse22	14937
excourse62	14937
excourse63	14641
excourse65	14641
excourse68	13551
excourse72	13512
excourse74	13291
excourse67	13291
BD0145EN	12497

3. Clustering-based Recommender System

- With the generated user profile vectors, we can easily calculate similarities between users based on their shared interests. We applied clustering algorithms, like K-means, to group users with similar learning preferences. For each group, we curated a list of popular courses to recommend to users with similar interests.
- The dataset utilized consists of user profile vectors, which contain the genre scores for each user. We normalized these values and applied PCA to reduce dimensionality, generating new principal components that represent the features. These components were then input into the K-means algorithm to determine the number of clusters that could be formed.

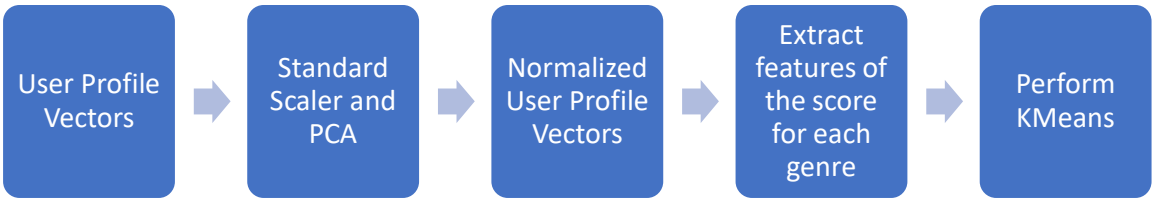

Fig6: Flowchart of clustering-based recommender system

Result of clustering-based recommender system

- We found that the 8 clusters and 9 principal components performed the best for clustering.

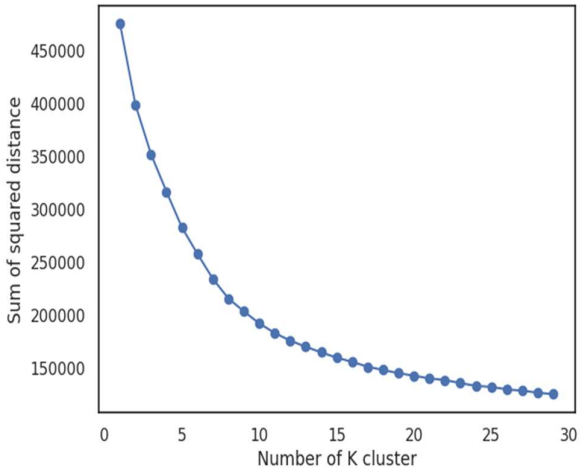

Fig7: Number of K Clusters vs Sum of Squared Distance

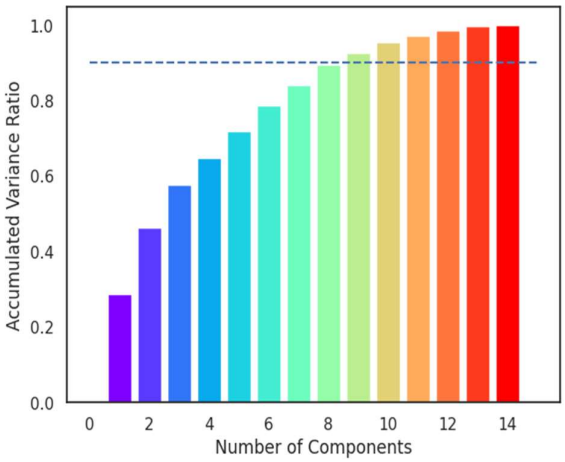

Fig8: Number of Components vs Accumulated Variance Ratio

- On average, there are around 41 courses are recommended to users. (average_recommended_courses = 40.889)

- The below are the 10 most recommended courses across users:

COURSE ID	USER
DS0321EN	32108
SC0101EN	31162
WA0101EN	30990
ML0120ENv2	30705
CC0103EN	30425
CL0101EN	30266
DS0301EN	29644
BD0115EN	29610
DB0101EN	29551
CO0101EN	29408

4. Collaborative Filtering based Recommender System using K Nearest Neighbor

- We implemented KNN-based collaborative filtering on the user-item interaction matrix.
- Using the Surprise library, we loaded the rating_df dataset and trained the KNN model. The root mean squared error (RMSE) was then calculated based on the model's predictions on the test data.
- After splitting the user-course rating data, the KNN classification model was trained on the training set. Predictions were made on the test set, and the RMSE was evaluated to assess the model's performance.

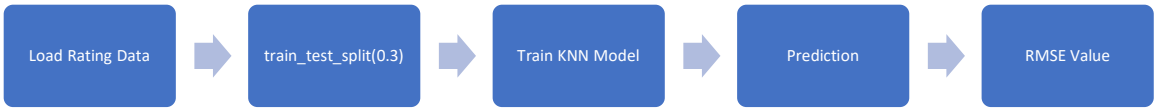

Fig9: Flowchart of KNN based recommender system

Result of K Nearest Neighbor Recommender Systems

- Root Mean Square Error of the testing(RMSE): 1.2889

5. Non negative Matrix based Recommender System

- We applied NMF-based collaborative filtering to the user-item matrix, which breaks down a large sparse matrix into two smaller, denser matrices.
- After splitting the user-course rating data with a 70/30 train-test ratio, the NMF model was trained on the training set. Predictions were then made on the test set, and the RMSE was calculated to evaluate the model's accuracy.

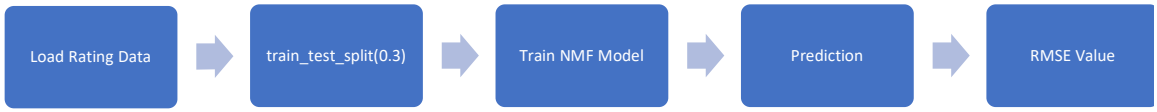

Fig10: Flowchart of NMF based recommender system

Result of NMF Recommender Systems

- Root Mean Square Error of the testing(RMSE): 1.300

6. Neural Network Embedding based Recommender System

- Next, we used TensorFlow to train neural networks, allowing us to extract latent features for users and items from the hidden layers.
- The trained neural network was used to predict course ratings.
- Neural networks can also effectively capture latent user and item features. The ANN predicts ratings by performing a dot product of user and course one-hot encodings, followed by activation using ReLU.

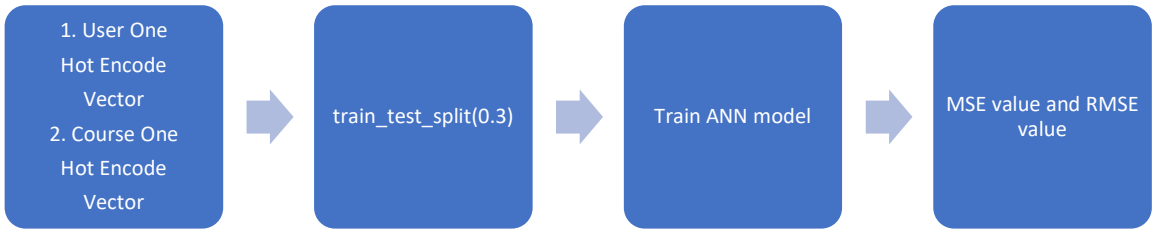

Fig11: Flowchart of Neural Network Embedding based recommender system

7. Performance of the Collaborative-Filtering Models

- Additionally, we extended the ANN model by building both regression and classification methods, using two embedding vectors as input to a neural network for predicting ratings.
- We developed regression and classification models to predict ratings based on the combined embedding vectors.
- For the regression model, we employed L1, L2, and ElasticNet regularization, while for classification, we used Logistic Regression, Random Forest and Bagging models.
- The combined embedding vectors were trained and tested to predict the rating scores.

- **Regression Models:**

Model	MSE	RMSE
Ridge	0.662327	0.813835
Lasso	0.662299	0.813818
ElasticNet	0.662299	0.813818

- **Classification Model:**

Model	Accuracy	Precision	Recall	F-Score	MSE	RMSE
Logistic Regression	0.3341	0.3349	0.3341	0.3248	1.3943	1.1808
Random Forest	0.3366	0.3379	0.3366	0.3287	1.3734	1.1719
Bagging	0.3372	0.3373	0.3372	0.3357	1.3219	1.1497

Conclusion

In this project, we implemented various machine learning techniques to enhance course recommendation systems. Starting with preprocessing techniques like Bag of Words (BoW) and stopword removal, we transformed raw textual data into numerical features for analysis. Using cosine similarity, we recommended similar courses based on user enrollment data. We further enhanced the recommendations by applying clustering algorithms like K-means to group users with similar learning interests, enabling personalized suggestions.

Collaborative filtering methods, such as KNN and NMF, were used to predict user-course interactions. These models were evaluated using root mean squared error (RMSE) to assess prediction accuracy. Neural networks, trained using TensorFlow, extracted latent features of users and courses, further refining the rating predictions.

Moreover, we extended our models by building regression and classification methods, leveraging embedding vectors to predict course ratings. The regression models used regularization techniques (L1, L2, and ElasticNet), while classification models incorporated methods like Logistic Regression, Random Forest and Bagging.

Overall, the combination of traditional collaborative filtering techniques, deep learning models, and advanced regression/classification methods provided an effective, multi-layered approach to recommending personalized courses based on user preferences.

4. Real time Applications of Key Takeaways

- E-Learning Platforms: The recommendation system can be integrated into online education platforms like Coursera, Udemy, or edX to suggest personalized courses to learners based on their past enrollments and learning patterns. This enhances user engagement and helps learners discover relevant content.
- Corporate Training Programs: Companies providing employee training programs can use similar systems to recommend skill development courses tailored to an employee's role, past training, and learning preferences. This ensures employees are exposed to the most relevant content for career growth.
- Entertainment Services: Streaming platforms like Netflix, Hulu, or Spotify can use these models to recommend personalized shows, movies, or music based on user preferences and behavior. Collaborative filtering and neural networks can improve content discovery.
- E-Commerce: In online retail, these techniques can recommend products based on user purchase history, preferences, and browsing behavior. This could be highly beneficial for platforms like Amazon, helping users find products aligned with their interests.
- Job Portals: Job search platforms like LinkedIn and Indeed can utilize these models to recommend jobs based on a candidate's profile, skills, and past applications, improving the relevance of job suggestions and career opportunities.
- Social Media Platforms: These recommendation models can be applied to suggest friends, groups, or content on platforms like Facebook, Instagram, or Twitter based on users' activities and interests, enhancing the user experience by curating relevant connections and content.
- Online Advertising: Personalized advertisements can be served to users based on their interaction history, preferences, and behavioral data. This improves ad targeting and user engagement, benefiting platforms like Google Ads or Facebook Ads.

These real-world applications demonstrate how personalized recommendations can enhance user experiences across various industries by providing relevant, tailored content.

5. Coursera Capstone Project Completion Certificate