Building a Personalized Online Course Recommender System With Machine Learning

John Pauline Pineda

December 31, 2023

OUTLINE

- Introduction and Background
- Exploratory Data Analysis
- Content-Based Recommender System Using Unsupervised Learning
- Collaborative Filtering Recommender System Using Supervised Learning
- Conclusion
- Appendix

INTRODUCTION

- ## Problem Statement
  - The current landscape of online education is marked by an overwhelming abundance of course offerings, making it challenging for learners to navigate and identify the most relevant and engaging courses aligned with their individual preferences and learning objectives.
- ## Hypothesis
  - It is hypothesized that the development of an online course recommender system will create a user-centric, adaptive, and effective learning platform that maximizes the educational value for each user. Its implementation will provide personalized recommendations ensuring that learners receive contents that align with their specific interests, shared preferences with other learners, and similarities to the courses they're currently enrolled in – leading to a more engaging and effective learning experience.
- ## Study Objective
  - This project aims to conduct the following on gathered data containing user profiles and course preferences –
    - Data preprocessing to extract and transform relevant features suitable for analysis and model training
    - Exploratory data analysis to identify similarity patterns and correlations among the distribution of user and course data
    - Recommender system model building using machine learning algorithms including regression, classification, and clustering
- ## Tools
  - Python APIs: SciPy, Pandas, MatPlotLib, NumPy, NLTK, GenSim, WordCloud, Scikit-Learn, TensorFlow, Keras

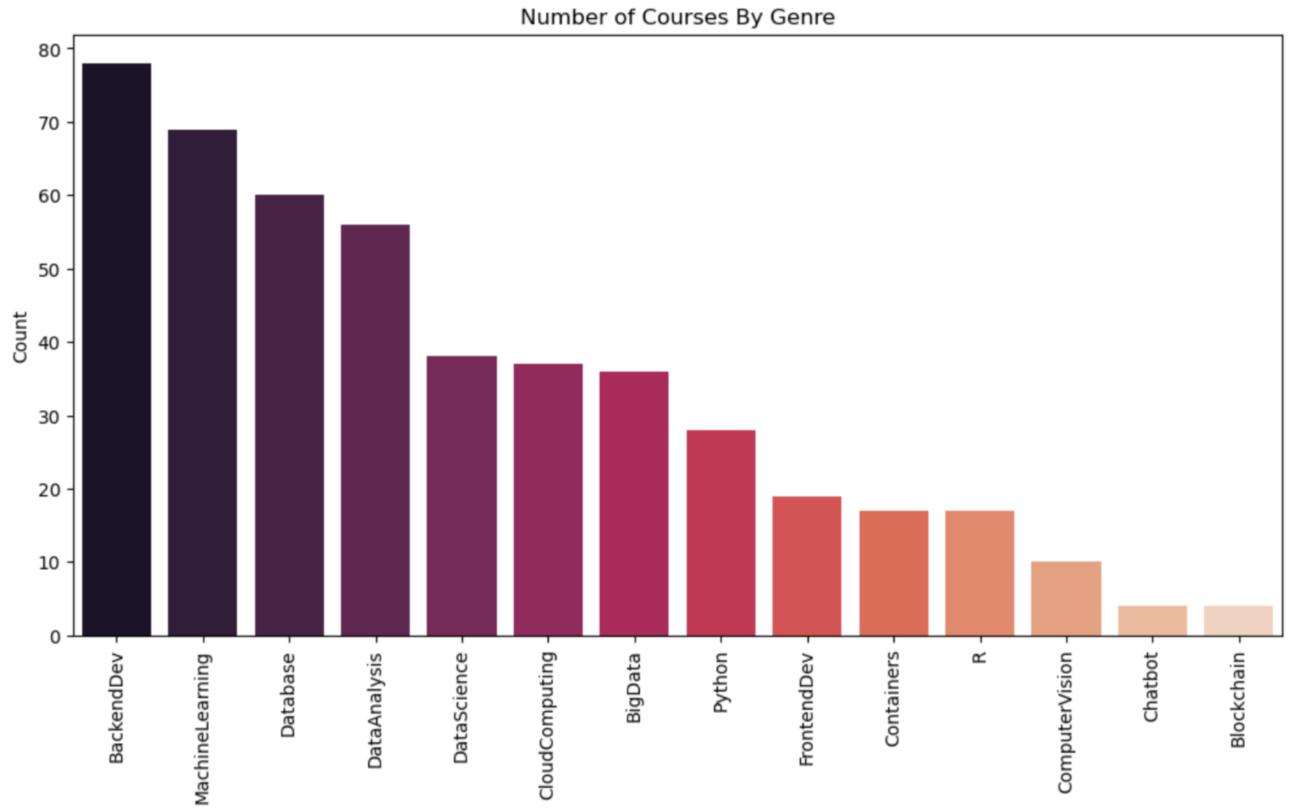
Section 1

Exploratory Data Analysis

EXPLORATORY DATA ANALYSIS

• Course Counts per Genre

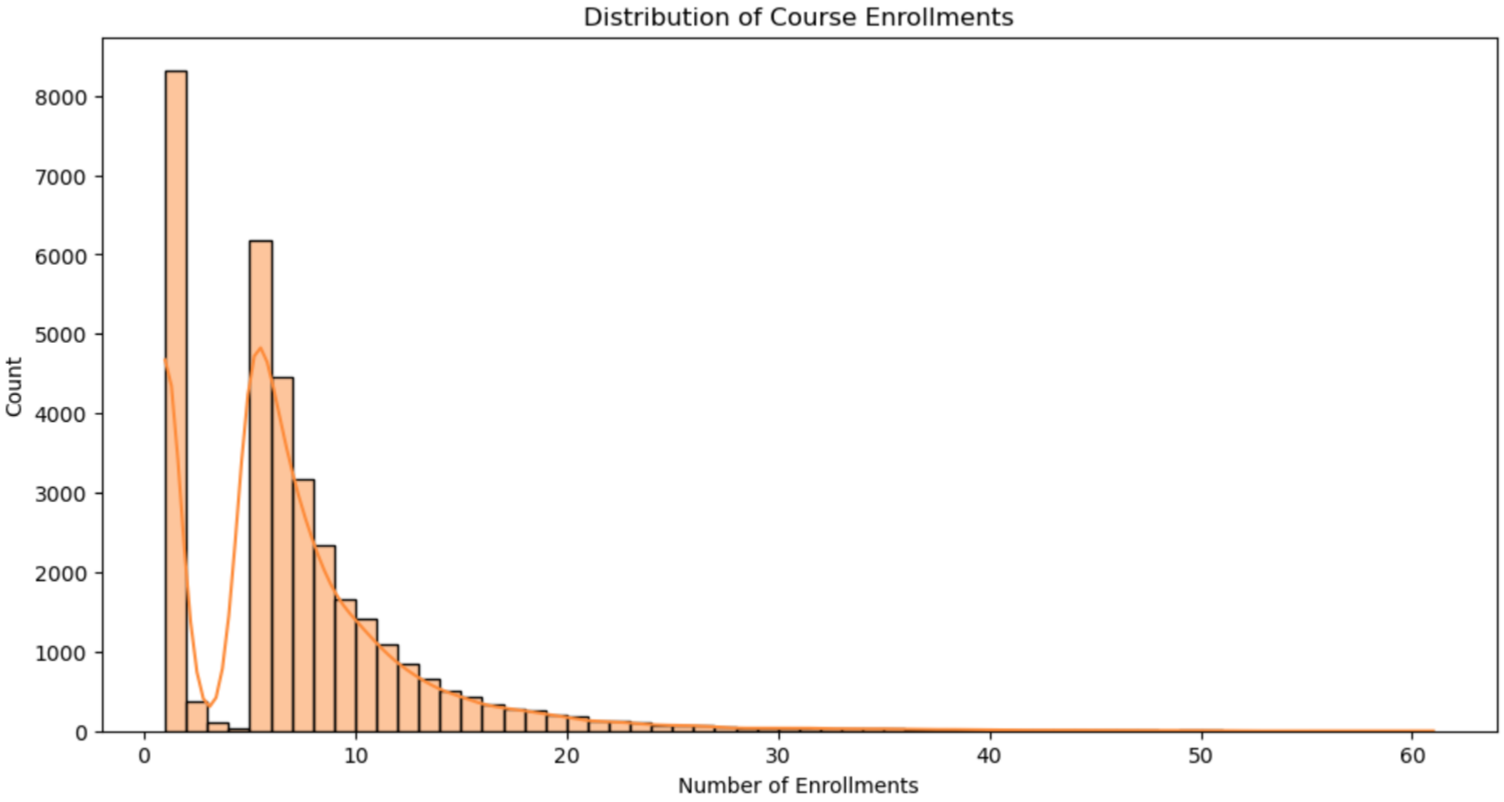
- More courses were noted for the **Backend Dev**, **Machine Learning**, and **Database** genres, among others.



	Count
BackendDev	78
MachineLearning	69
Database	60
DataAnalysis	56
DataScience	38
CloudComputing	37
BigData	36
Python	28
FrontendDev	19
Containers	17
R	17
ComputerVision	10
Chatbot	4
Blockchain	4

EXPLORATORY DATA ANALYSIS

- Course Enrollment Distribution
  - Course enrollment distribution is bimodal with most users enrolled in **1 course** or **5 to 6 courses**.



EXPLORATORY DATA ANALYSIS

• Top 20 Most Popular Courses

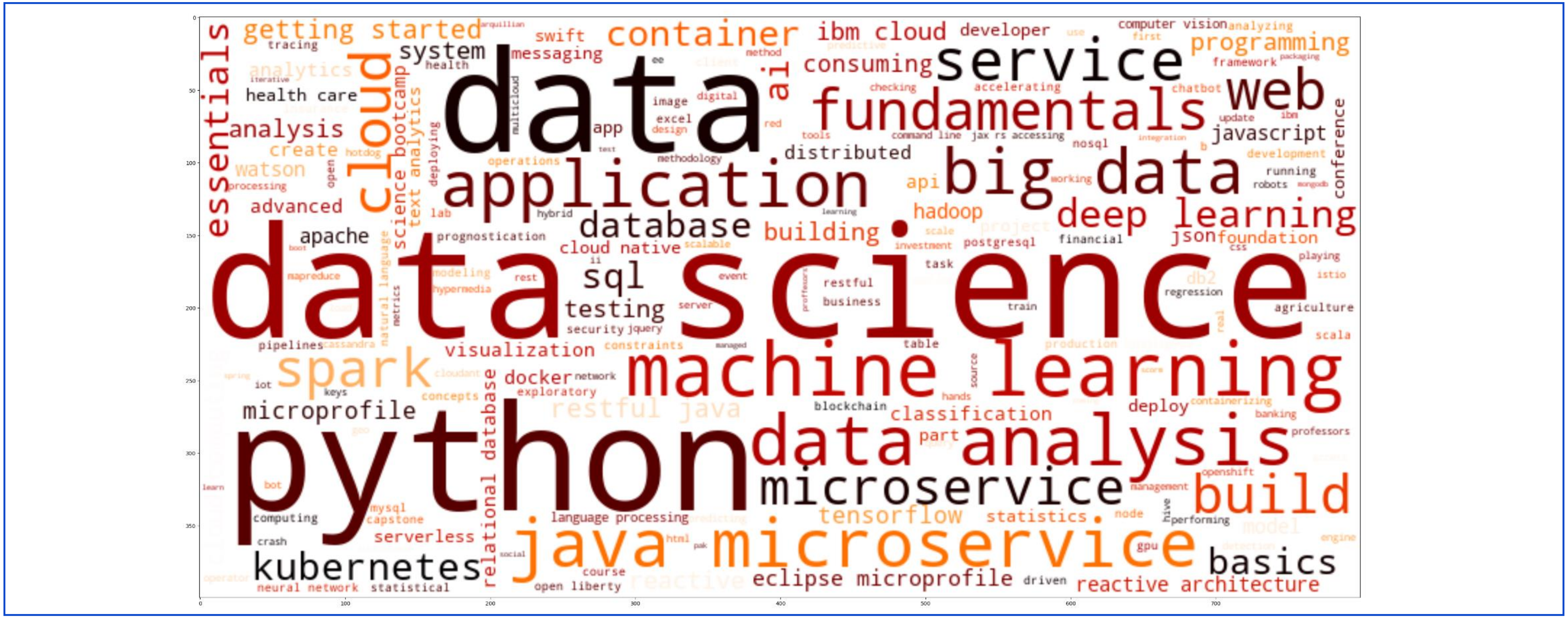
- Among the highly rated courses included **Data Science (DSxxxxxx)** and **Big Data (BDxxxxxx)**.

	COURSE_ID	TITLE	Ratings
0	PY0101EN	python for data science	14936
1	DS0101EN	introduction to data science	14477
2	BD0101EN	big data 101	13291
3	BD0111EN	hadoop 101	10599
4	DA0101EN	data analysis with python	8303
5	DS0103EN	data science methodology	7719
6	ML0101ENv3	machine learning with python	7644
7	BD0211EN	spark fundamentals i	7551
8	DS0105EN	data science hands on with open source tools	7199
9	BC0101EN	blockchain essentials	6719

	COURSE_ID	TITLE	Ratings
10	DV0101EN	data visualization with python	6709
11	ML0115EN	deep learning 101	6323
12	CB0103EN	build your own chatbot	5512
13	RP0101EN	r for data science	5237
14	ST0101EN	statistics 101	5015
15	CC0101EN	introduction to cloud	4983
16	CO0101EN	docker essentials a developer introduction	4480
17	DB0101EN	sql and relational databases 101	3697
18	BD0115EN	mapreduce and yarn	3670
19	DS0301EN	data privacy fundamentals	3624

EXPLORATORY DATA ANALYSIS

- Word Cloud of Course Titles
  - The most prominent words used in the course titles were **Python**, **Data Science**, and **Data**, among others.

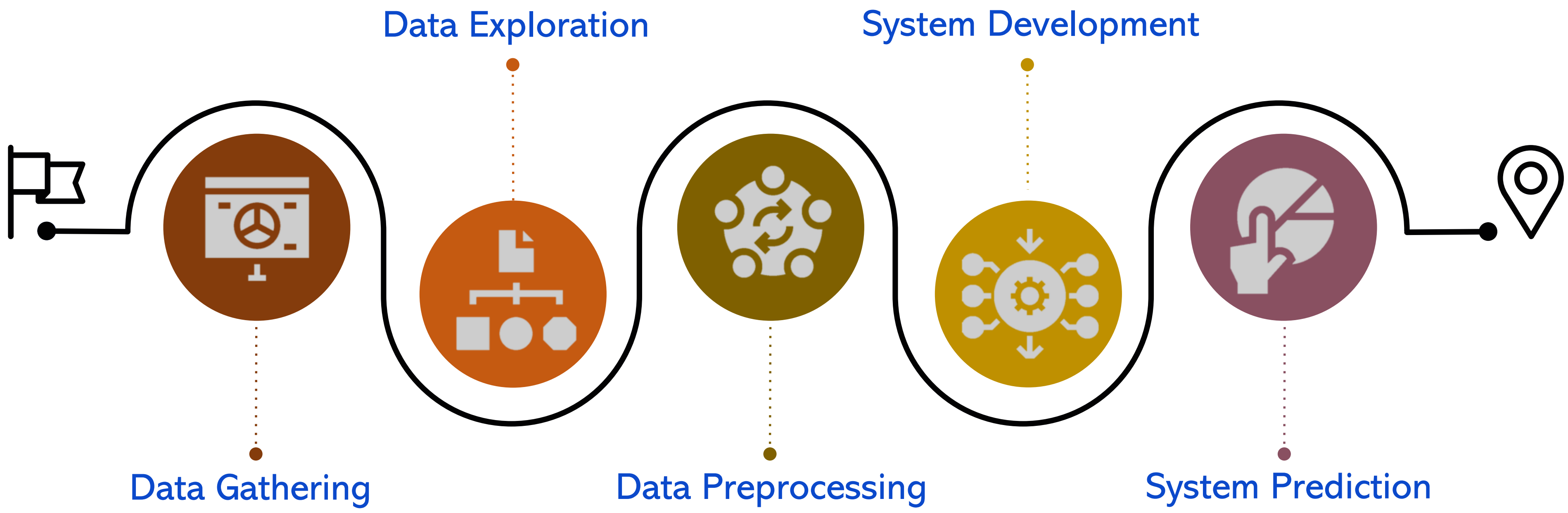


Section 2

Content-Based Recommender System Using Unsupervised Learning

CONTENT-BASED SYSTEM

- Process Flowchart Using User Profile and Course Genres



- User Information
- Course Information
- Course Rating
- Course Genre

- Course-Genre Table
- User Rating Distribution
- Course Rating Preferences
- Course Textual Frequency

- User Profile Vector Generation
- Course Genre Vector Generation

- Interest Score Calculation (Using the Dot Product of User Profile and Course Genre Vectors) and Threshold Setting

- Recommendation for Unenrolled Courses with Interest Scores Above Threshold

CONTENT-BASED SYSTEM

- Evaluation Results of User Profile-Based Recommender System
  - On average, 19 recommendations are provided with the top 10 recommendations given below.
  - **Hyperparameters:** Threshold=10

Average new | unseen course recommendations per user for the test dataset

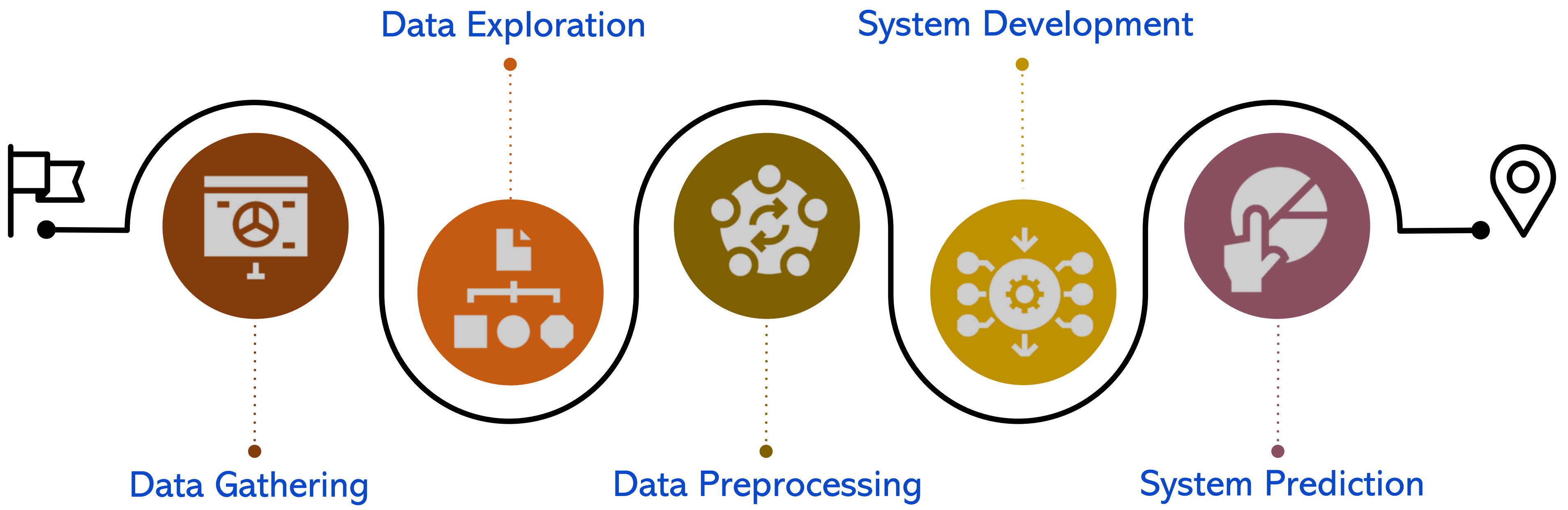
```
18.62679972290352
```

Top 10 most recommended courses across all users

COURSE_ID	Count
TA0106EN	608
GPXX0IBEN	548
excourse22	547
excourse21	547
ML0122EN	544
excourse06	533
excourse04	533
GPXX0TY1EN	533
excourse31	524
excourse73	516

CONTENT-BASED SYSTEM

- Process Flowchart Using Course Similarity



- User Information
- Course Information
- Course Rating
- Course Genre

- Course-Genre Table
- User Rating Distribution
- Course Rating Preferences
- Course Textual Frequency

- Tokenization
- Stop Word Removal
- Bag-of-Words Feature Extraction

- Course Similarity Score Calculation (Using the Bag-of-Words Features) and Threshold Setting

- Recommendation for Unenrolled Courses with Similarity Scores Above Threshold

CONTENT-BASED SYSTEM

- ## Evaluation Results of Course Similarity-Based Recommender System
  - On average, 12 recommendations are provided with the top 10 recommendations given below.
  - **Hyperparameters:** Threshold=0.60

Average new | unseen course recommendations per user for the test dataset

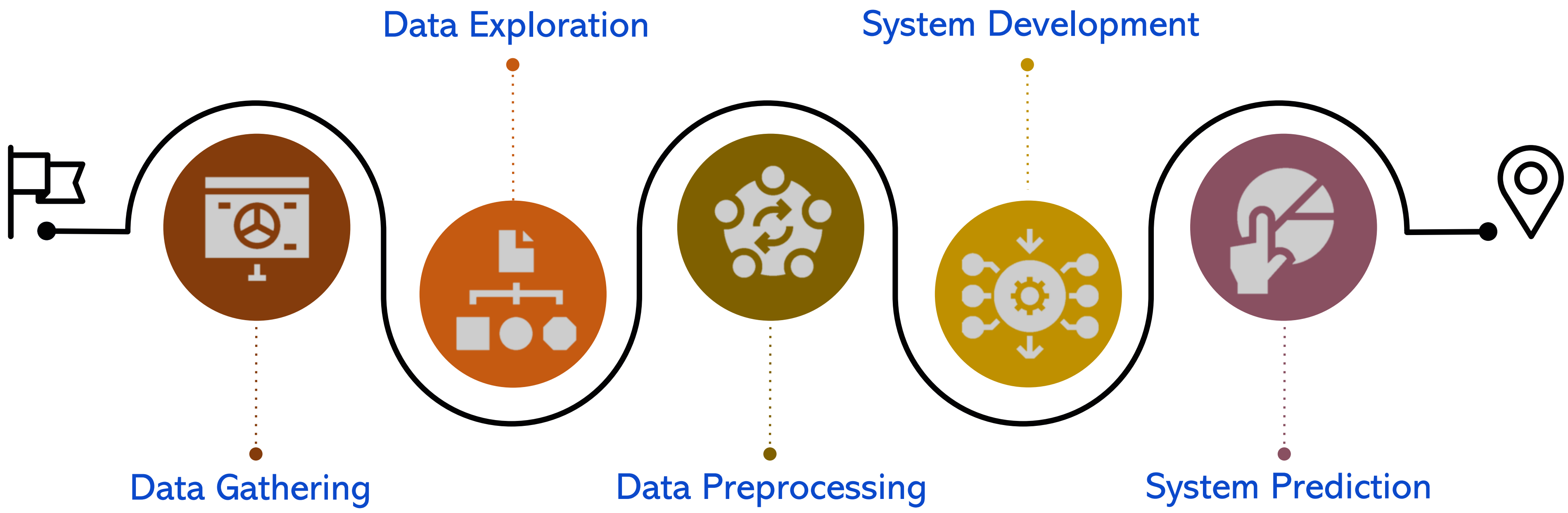
11.573753814852493

Top 10 most recommended courses across all users

COURSE_ID	Count
excourse62	579
excourse22	579
DS0110EN	562
excourse65	555
excourse63	555
excourse72	551
excourse68	550
excourse74	539
excourse67	539
BD0145EN	506

CONTENT-BASED SYSTEM

- Process Flowchart Using User-Profile Clusters



Data Gathering
- User Information
- Course Information
- Course Rating
- Course Genre

Data Exploration
- Course-Genre Table
- User Rating Distribution
- Course Rating Preferences
- Course Textual Frequency

Data Preprocessing
- User Profile Normalization
- Dimensionality Reduction of User Profile Vector by Principal Component Analysis

System Development
- K-Means Clustering Analysis for User Profile Vector Principal Components
- Recommendation Score Calculation and Threshold Setting by Cluster

System Prediction
- User Cluster Determination
- Cluster-Specific Recommendation for Unenrolled Courses with Scores Above Threshold

CONTENT-BASED SYSTEM

• Evaluation Results of Clustering-Based Recommender System

- On average, 17 recommendations are provided with the top 10 recommendations given below.
- **Hyperparameters:** Number of Clusters=15, Number of Principal Components=9, Threshold=10

Average new | unseen course recommendations per user for the test dataset

```
16.720858895705522
```

Top 10 most recommended courses across all users

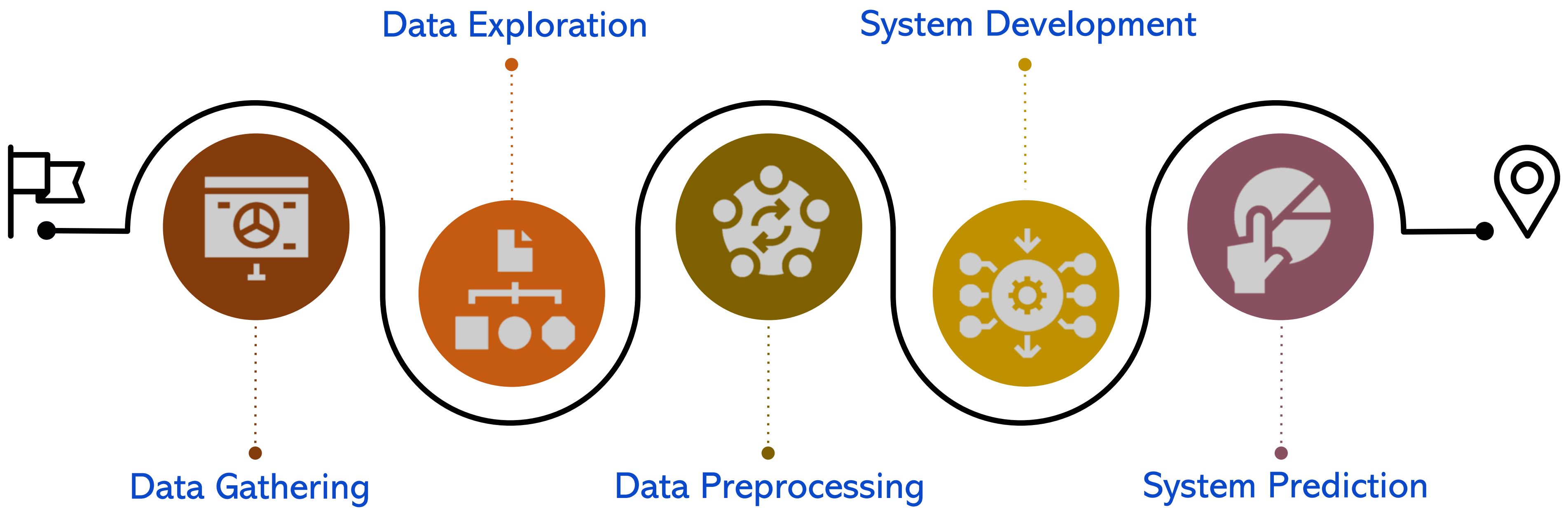
COURSE_ID	Count
ML0115EN	678
ST0101EN	636
DS0105EN	602
DB0101EN	582
CL0101EN	570
DS0103EN	569
BD0111EN	562
DS0301EN	554
CC0101EN	516
BD0211EN	510

Section 3

Collaborative-Filtering Recommender System Using Supervised Learning

COLLABORATIVE-FILTERING SYSTEM

- Process Flowchart Using K-Nearest Neighbors-Based Approach



- User Information
- Course Rating

- User Rating Distribution
- Course Rating Preferences

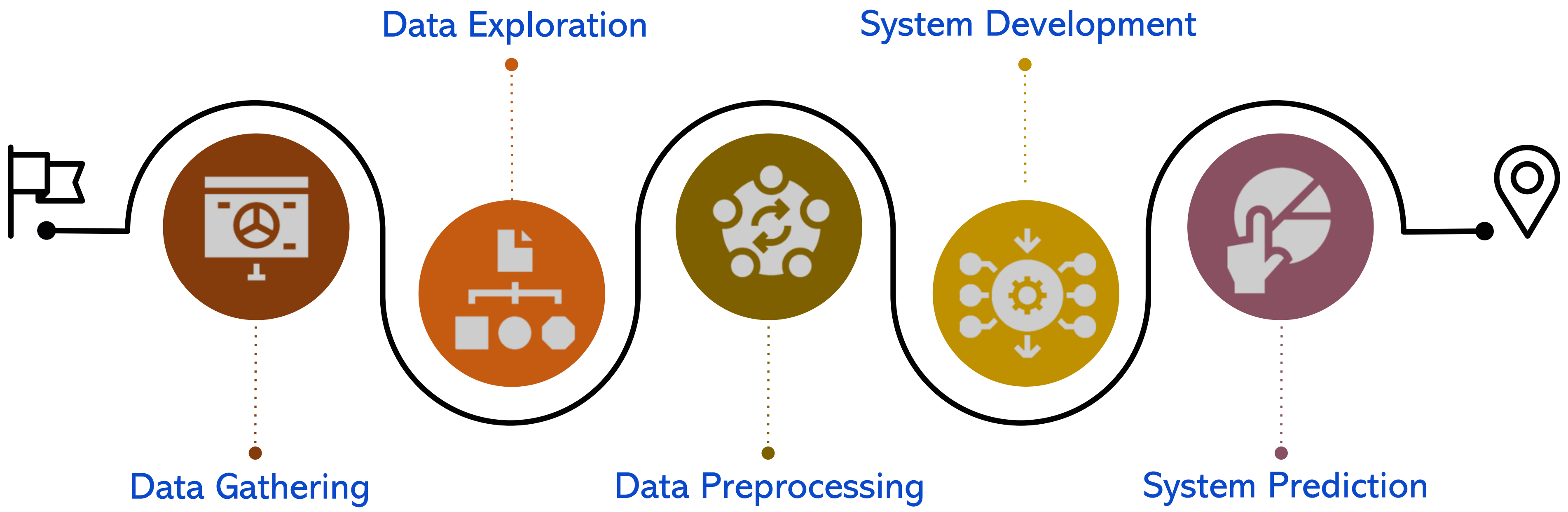
- User-Course Interaction Matrix Generation

- User Similarity Score Calculation (Using the Rating History) and Threshold Setting
- Rating Prediction Model Development using KNN

- Recommendation for Unenrolled Courses with Higher Ratings Based on the Aggregation of the User's K- Nearest Neighbor's Ratings

COLLABORATIVE-FILTERING SYSTEM

- Process Flowchart Using Non-Matrix Factorization-Based Approach



- User Information
- Course Rating

- User Rating Distribution
- Course Rating Preferences

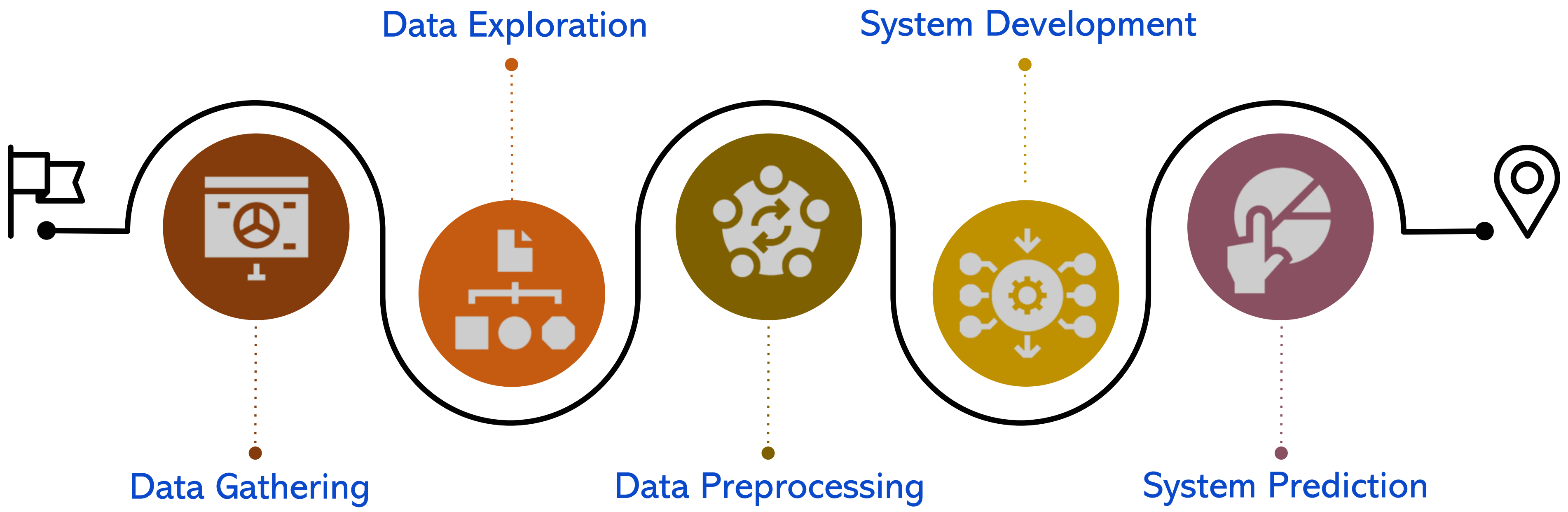
- User-Course Interaction Matrix Generation
- Matrix Decomposition to Transformed User-Features and Course-Features

- Rating Prediction Model Development using NMF (Using the Dot Product of Transformed User-Features and Course-Features)

- Recommendation for Unenrolled Courses with Higher Ratings Based on the Latent Features of Users and Courses Extracted from NMF

COLLABORATIVE-FILTERING SYSTEM

- Process Flowchart Using Neural Network Embedding-Based Approach



- User Information
- Course Rating

- User Rating Distribution
- Course Rating Preferences

- One-Hot Encoding of User Vector
- One-Hot Encoding of Course Vector

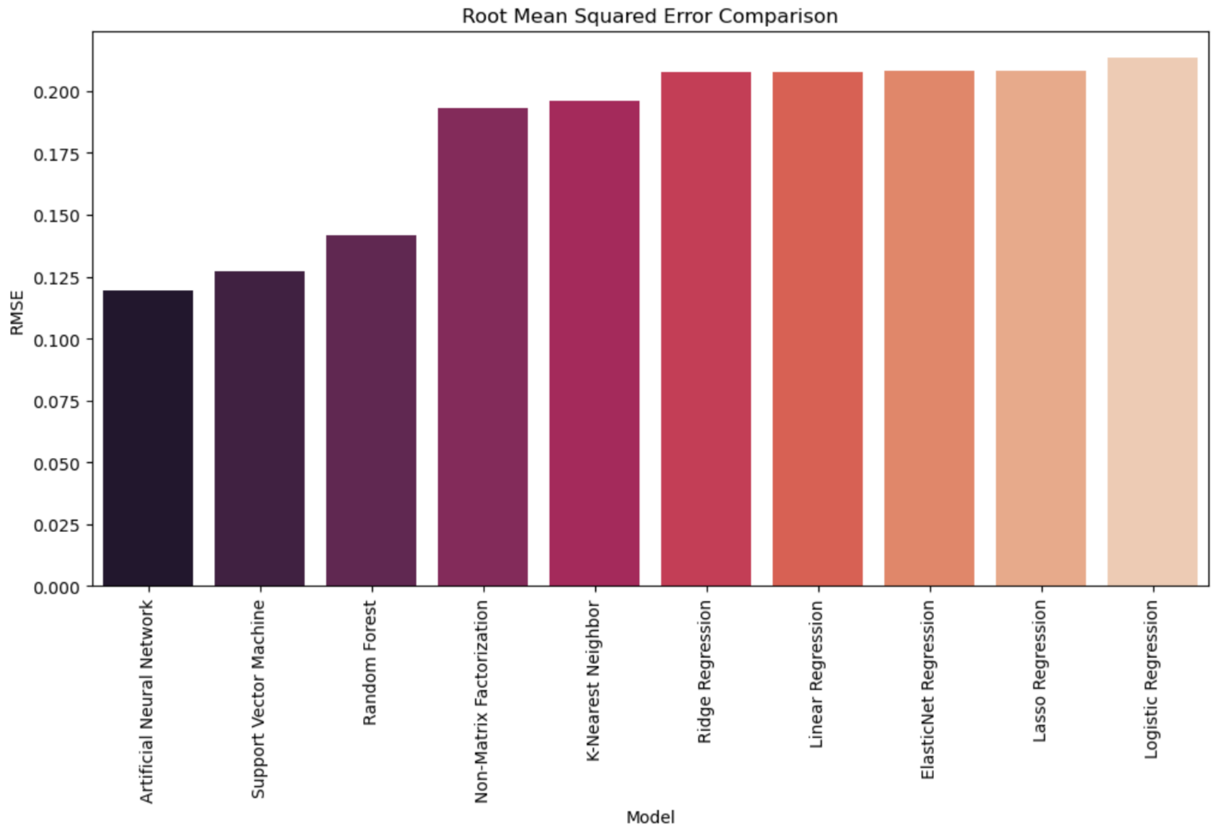
- Rating Prediction Model Development using Neural Network (Using the Dot Product of User-Embedded and Course-Embedded Layers)

- Recommendation for Unenrolled Courses with Higher Ratings Based on the Latent Features of Users and Courses Extracted from Neural Network

COLLABORATIVE-FILTERING SYSTEM

• Performance Comparison of Collaborative-Filtering Models

- The Neural Network Embedding-Based Approach demonstrated the lowest root mean squared error among all the collaborative-filtering model candidates evaluated.



Model	RMSE
Artificial Neural Network	0.119362
Support Vector Machine	0.127033
Random Forest	0.141933
Non-Matrix Factorization	0.193257
K-Nearest Neighbor	0.196312
Ridge Regression	0.207695
Linear Regression	0.207695
ElasticNet Regression	0.208401
Lasso Regression	0.208401
Logistic Regression	0.213301

Section 4

Conclusion

CONCLUSION

- ## Key Findings
  - 10 candidate recommender systems were developed based on two techniques – a **Content-Based Approach Using Unsupervised Learning** (3 models) and a **Collaborative Filtering-Based Approach Using Supervised Learning** (7 models).
  - Among all candidates, the **Rating Prediction Model Based on the Latent Features of Users and Courses Extracted from Neural Network** provided the lowest error based on RMSE at 0.119362.
- ## Next Steps
  - The current study can be further extended to improve the performance of the content-based and collaborative filtering methods in recommendation systems by addressing their respective limitations and exploring hybrid or enhanced techniques:
    - Content-Based Recommendation System
      - Deep learning architectures can be explored to automatically learn intricate patterns and representations from item features such as recurrent neural networks (RNNs) or transformers, to capture complex relationships within item content.
      - Ensemble methods can be explored by training multiple recommendation models with different feature representations and aggregating their predictions to improve robustness and generalization.
    - Collaborative Filtering-Based Recommendation System
      - Other matrix factorization techniques can be explored such as Singular Value Decomposition (SVD), Alternating Least Squares (ALS), or matrix factorization with deep learning approaches to effectively decompose the user-item interaction matrix into lower-dimensional matrices to better capture latent factors
      - Hybrid collaborative-content models can be explored which combine collaborative and content-based filtering methods to leverage the strengths of both approaches. Collaborative filtering can be used to capture user preferences while content-based recommendation can be employed to enhance recommendations with item features.

Section 5

Appendix

APPENDIX

- Source Data
  - Raw Data: Course Genre | Ratings | Course Description
  - Processed Data: Bag-of-Words Features | User Profile | Course Similarity Calculations
  - Embedded Data: User Embeddings | Course Embeddings
- Python Notebooks | Code Repository
  - GitHub URL: Exploratory Data Analysis
  - GitHub URL: Bag-of-Words Feature Extraction
  - GitHub URL: Similarity Computation Using Bag-Of-Words Features
  - GitHub URL: Content-Based Approach Using User Profiles and Course Genres
  - GitHub URL: Content-Based Approach Using Course Similarities
  - GitHub URL: Clustering-Based Approach
  - GitHub URL: Collaborative Filtering Using K-Nearest Neighbors
  - GitHub URL: Collaborative Filtering Using Non-Negative Matrix Factorization
  - GitHub URL: Course Rating Prediction Using Neural Network
  - GitHub URL: Regression-Based Rating Score Prediction Using Embedding Features
  - GitHub URL: Classification-Based Rating Mode Prediction Using Embedding Features

Thank You!