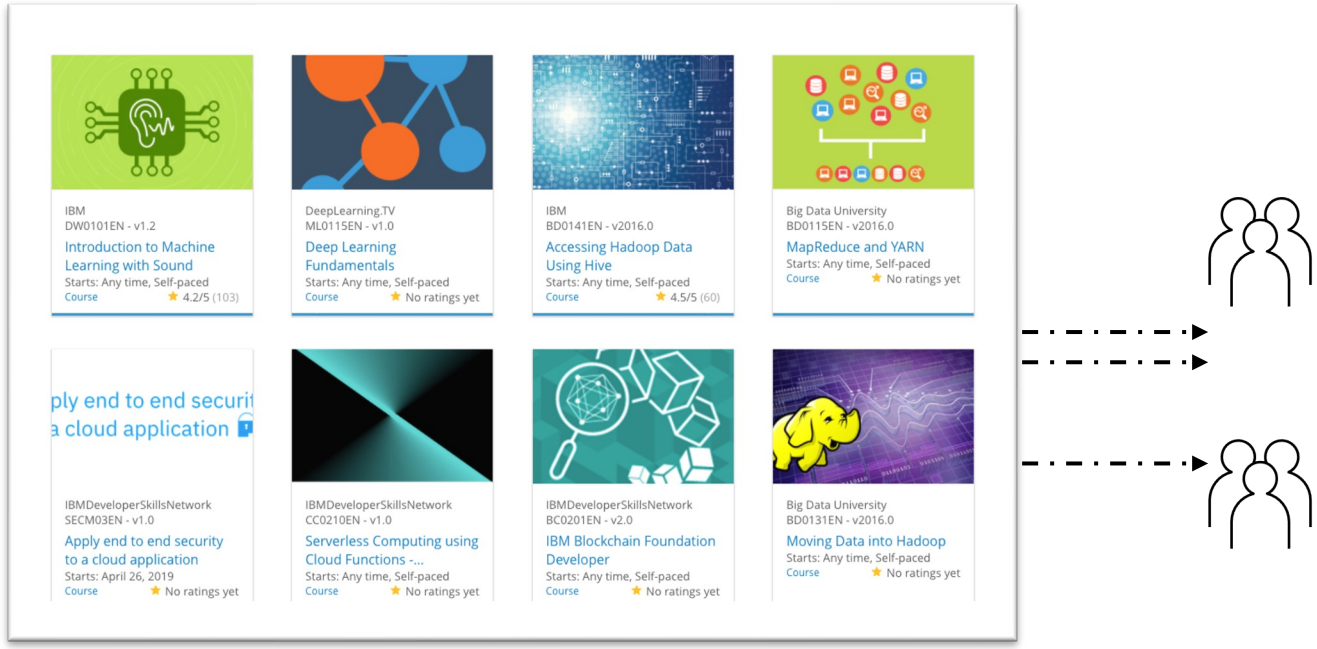
Build a Personalized Online Course Recommender System with Machine Learning

Venkata Tarun
Kumar Mavillapalli
11/09/23



Outline

- Introduction and Background
- Exploratory Data Analysis
- Content-based Recommender System using Unsupervised Learning
- Collaborative-filtering based Recommender System using Supervised learning
- Conclusion
- Appendix

Introduction

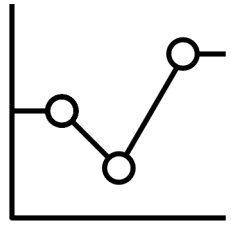
Project background and context

In AI Training Room, learners from all over the world can learn various technologies such as Machine Learning, AI, Data Science, Cloud, and App development. The company's rapid growth has increased the number of courses and learners, which makes it harder for learners to find new interesting courses and create a personalized learning path. Thus, this project aims to build a recommender system that helps learners discover new courses that match their interests and better pave their learning paths.

Problem states and hypotheses

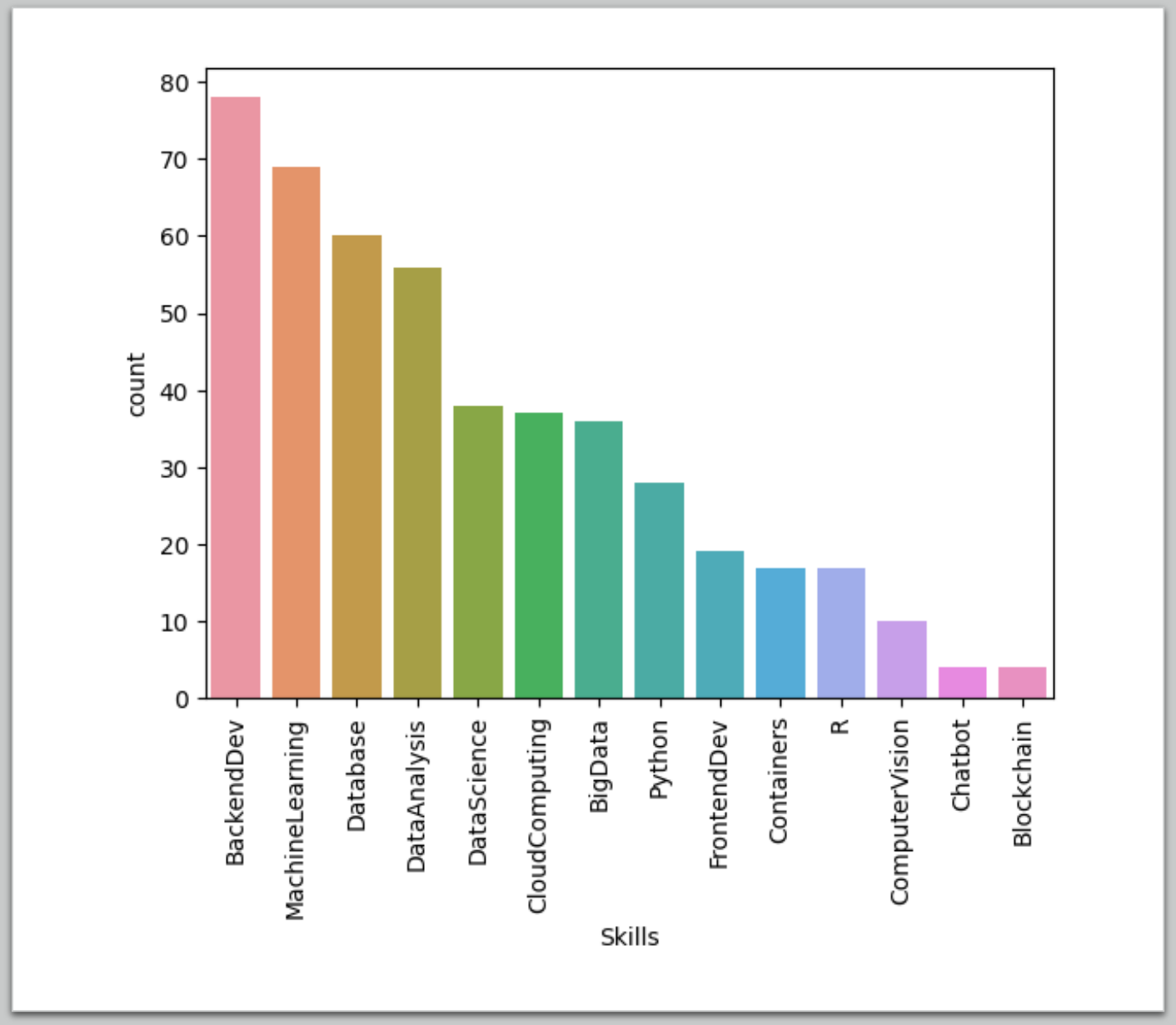
The main problem addressed in this project is the difficulty for learners to discover new courses that fit their interests and to create a personalized learning path due to the increasing number of courses and learners. Our hypothesis is that building a personalized recommender system based on course content and learners' previous interactions with courses can help learners discover new courses of interest and facilitate their learning paths. We will explore and compare various unsupervised and supervised machine learning models to find the best performing model for this task.

Exploratory Data Analysis



Course counts per genre

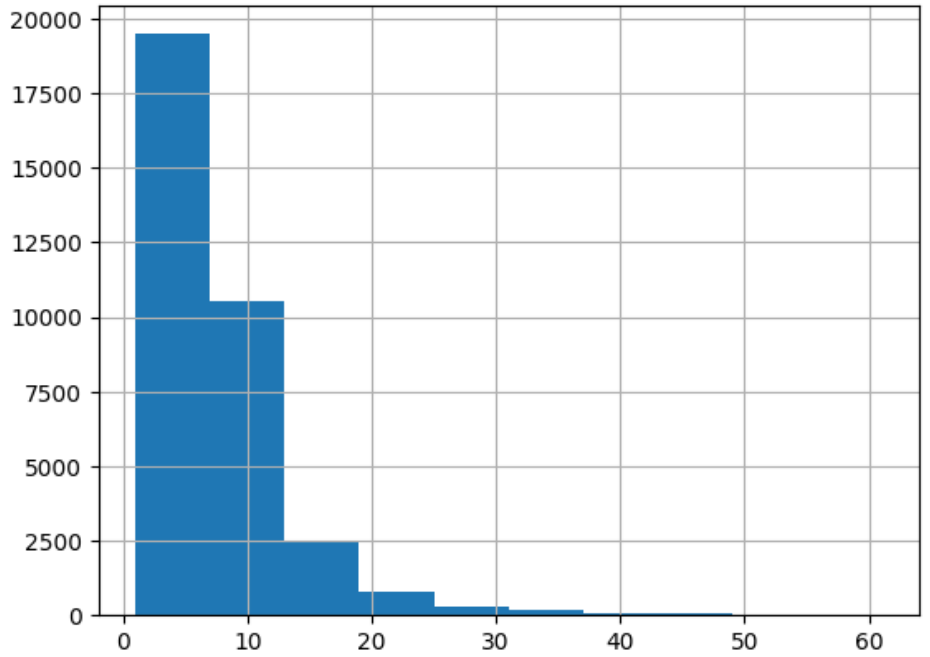
Genre	Count
BackendDev	78
MachineLearning	69
Database	60
DataAnalysis	56
DataScience	38
CloudComputing	37
BigData	36
Python	28
FrontendDev	19
Containers	17
R	17
ComputerVision	10
Chatbot	4
Blockchain	4



Course enrollment distribution

- We have 233306 enrollments

- The histogram shows the enrolment distributions, e.g., how many users rated just 1 item or how many rated 10 items, etc.

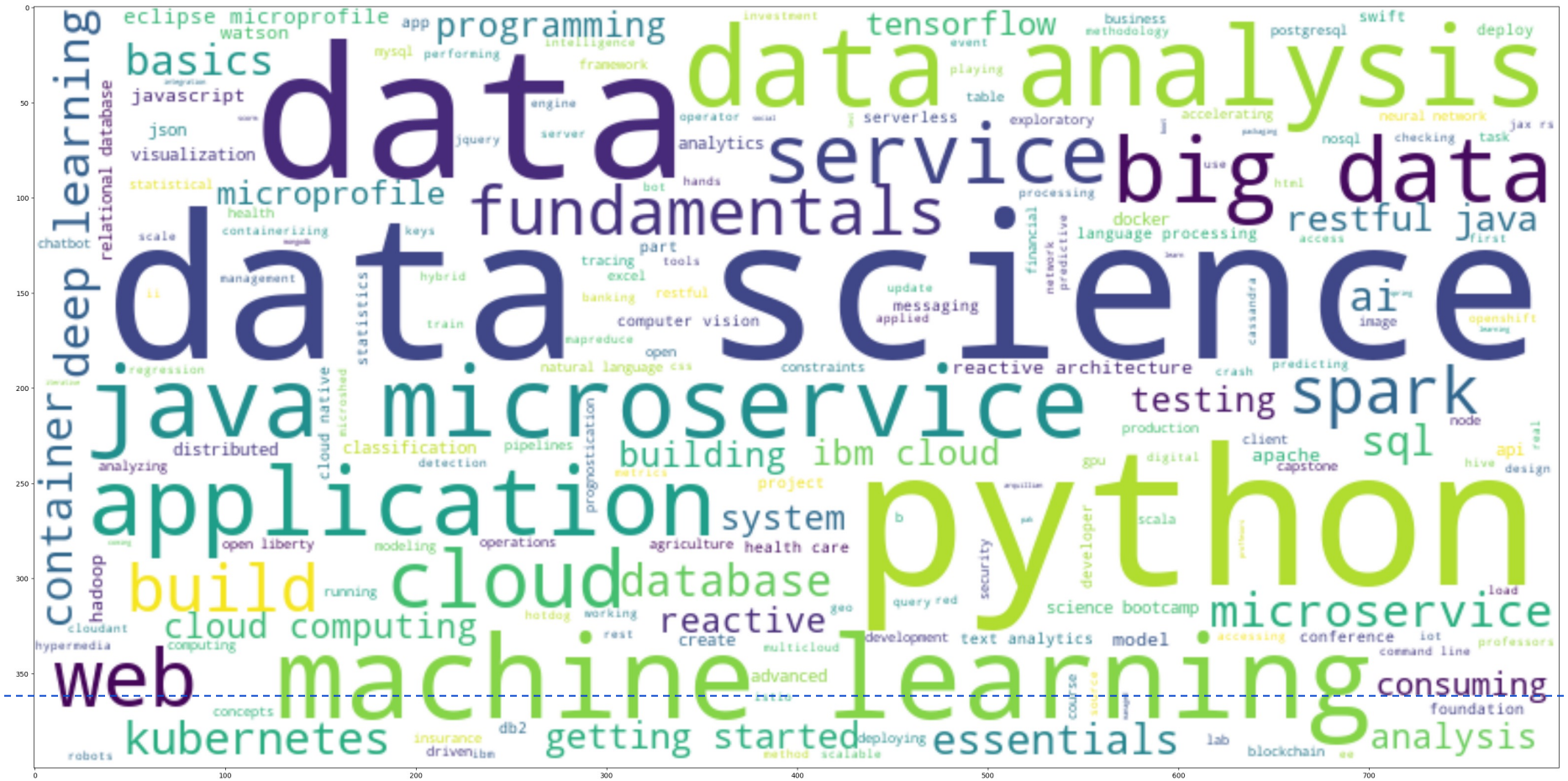


20 most popular courses

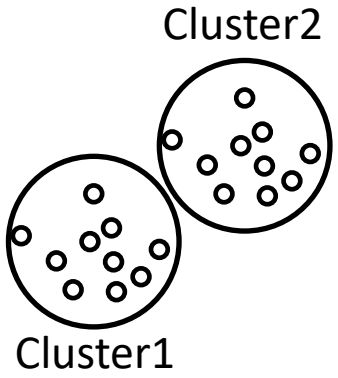
Used Pandas groupby() and size() methods on the item column to aggregate the rating count for each item, then use the sort_values() method to sort the course enrollment count, and use the slice method to get the top 20 courses.

ID	ENROLLS	TITLE
DS0301EN	3624	data privacy fundamentals
BD0115EN	3670	mapreduce and yarn
DB0101EN	3697	sql and relational databases 101
CO0101EN	4480	docker essentials a developer introduction
CC0101EN	4983	introduction to cloud
ST0101EN	5015	statistics 101
RP0101EN	5237	r for data science
CB0103EN	5512	build your own chatbot
ML0115EN	6323	deep learning 101
DV0101EN	6709	data visualization with python
BC0101EN	6719	blockchain essentials
DS0105EN	7199	data science hands on with open source tools
BD0211EN	7551	spark fundamentals i
ML0101ENv3	7644	machine learning with python
DS0103EN	7719	data science methodology
DA0101EN	8303	data analysis with python
BD0111EN	10599	hadoop 101
BD0101EN	13291	big data 101
DS0101EN	14477	introduction to data science
PY0101EN	14936	python for data science

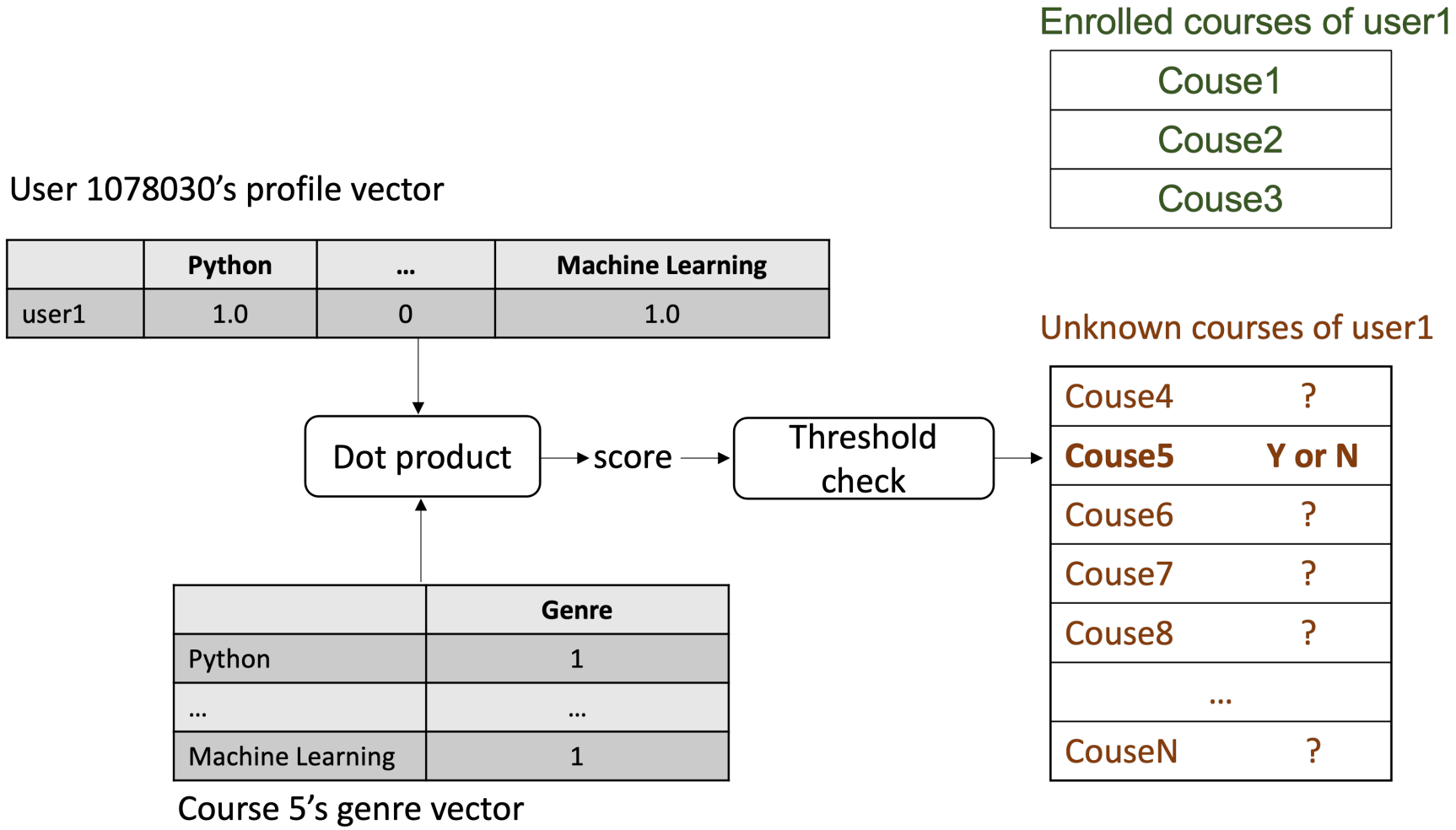
Word cloud of course titles



Content-based Recommender System using Unsupervised Learning



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



Evaluation results of user profile-based recommender system

```
# The threshold can be fine-tuned to adjust the size of generated recommendations
score_threshold = 10.0
```

On average, 61 courses have been recommended per user

Top-10 commonly recommended courses across all users

Course	Times recommended
TA0106EN	608
GPXX0IBEN	548
excourse22	547
excourse21	547
ML0122EN	544
excourse04	533
GPXX0TY1EN	533
excourse06	533
excourse31	524
excourse73	516

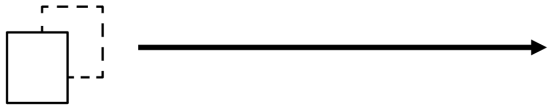
Flowchart of content-based recommender system using course similarity

Course 1: “Machine Learning for Everyone”

	machine	learning	for	everyone	beginners
course1	1	1	1	1	0

Course 2: “Machine Learning for Beginners”

	machine	learning	for	everyone	beginners
course2	1	1	1	0	1



Similarity Calculation:
Cosine, Euclidean, Jaccard index, …

75%

Evaluation results of course similarity based recommender system

In [27]:
```
res_dict = {}
users, courses, sim_scores = generate_recommendations_for_all()
res_dict['USER'] = users
res_dict['COURSE_ID'] = courses
res_dict['SCORE'] = sim_scores
res_df = pd.DataFrame(res_dict, columns=['USER', 'COURSE_ID', 'SCORE'])
# Save the dataframe
res_df.to_csv("profile_rs_results_course_similarities.csv", index=False)
```

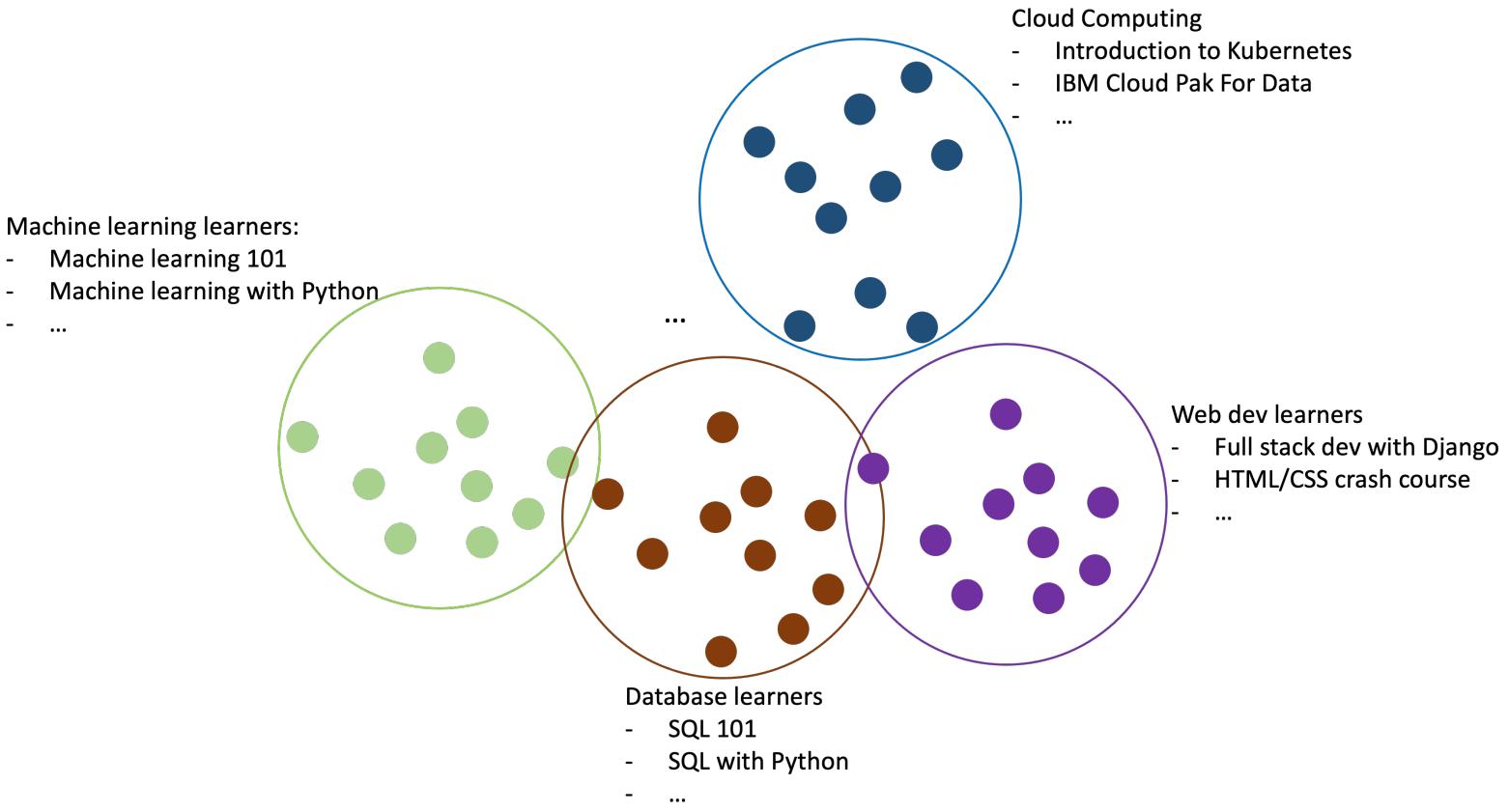
In [28]:
```
res_df.head()
```

Out[28]:

	USER	COURSE_ID	SCORE
0	37465	excourse67	0.708214
1	37465	excourse72	0.652535
2	37465	excourse74	0.650071
3	37465	BD0145EN	0.623544
4	37465	excourse68	0.616759

Flowchart of clustering-based recommender system

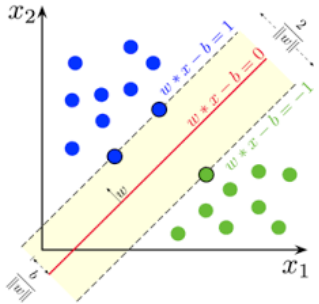
Clustering on User Profiles



Evaluation results of clustering-based recommender system

user in cluster 0 will be sugessted 3 courses as ['PY0101EN' 'CB0103EN' 'DA0101EN']
user in cluster 1 will be sugessted 3 courses as ['DS0101EN' 'BD0101EN' 'PY0101EN']
user in cluster 2 will be sugessted 3 courses as ['CO0301EN' 'CO0201EN' 'BC0101EN']
user in cluster 3 will be sugessted 3 courses as ['PY0101EN' 'ML0101ENv3' 'ML0115EN']
user in cluster 4 will be sugessted 3 courses as ['BD0111EN' 'BD0141EN' 'BD0115EN']
user in cluster 5 will be sugessted 3 courses as ['CB0103EN' 'DS0101EN' 'BD0101EN']
user in cluster 6 will be sugessted 3 courses as ['CO0101EN' 'CO0201EN' 'CO0301EN']
user in cluster 7 will be sugessted 3 courses as []
user in cluster 8 will be sugessted 3 courses as ['CO0101EN' 'PY0101EN' 'CC0101EN']
user in cluster 9 will be sugessted 3 courses as ['DS0101EN' 'RP0101EN' 'DS0103EN']
user in cluster 10 will be sugessted 3 courses as ['CO0101EN' 'LB0101ENv1' 'CO0401EN']
user in cluster 11 will be sugessted 3 courses as ['RP0101EN' 'DS0101EN' 'DS0103EN']

Collaborative-filtering Recommender System using Supervised Learning



Flowchart of KNN based recommender system

User-Item interaction matrix

	Machine Learning With Python	Machine Learning 101	Machine Learning Capstone	SQL with Python	Python 101
...	...	...	...	...	
user2	3.0	3.0	3.0	3.0	3.0
user3	2.0	3.0	3.0	2.0	
user4	3.0	3.0	2.0	2.0	3.0
user5	2.0	3.0	3.0		
user6	3.0	3.0	?		3.0
...	...	...	...	...	

Similar users

Predict the rating of user *user6* to item *Machine Learning Capstone*

Flowchart of NMF based recommender system

Non-negative Matrix Factorization

User-item interaction matrix: **A** 10000 x 100

	item1	...	item100
user1	...	...	
user2	3.0	3.0	3.0
user3	2.0	2.0	-
user4	3.0	2.0	3.0
user5	2.0	-	-
user6	3.0	-	3.0
...	...	...	

≈

User matrix: **U** 10000 x 16

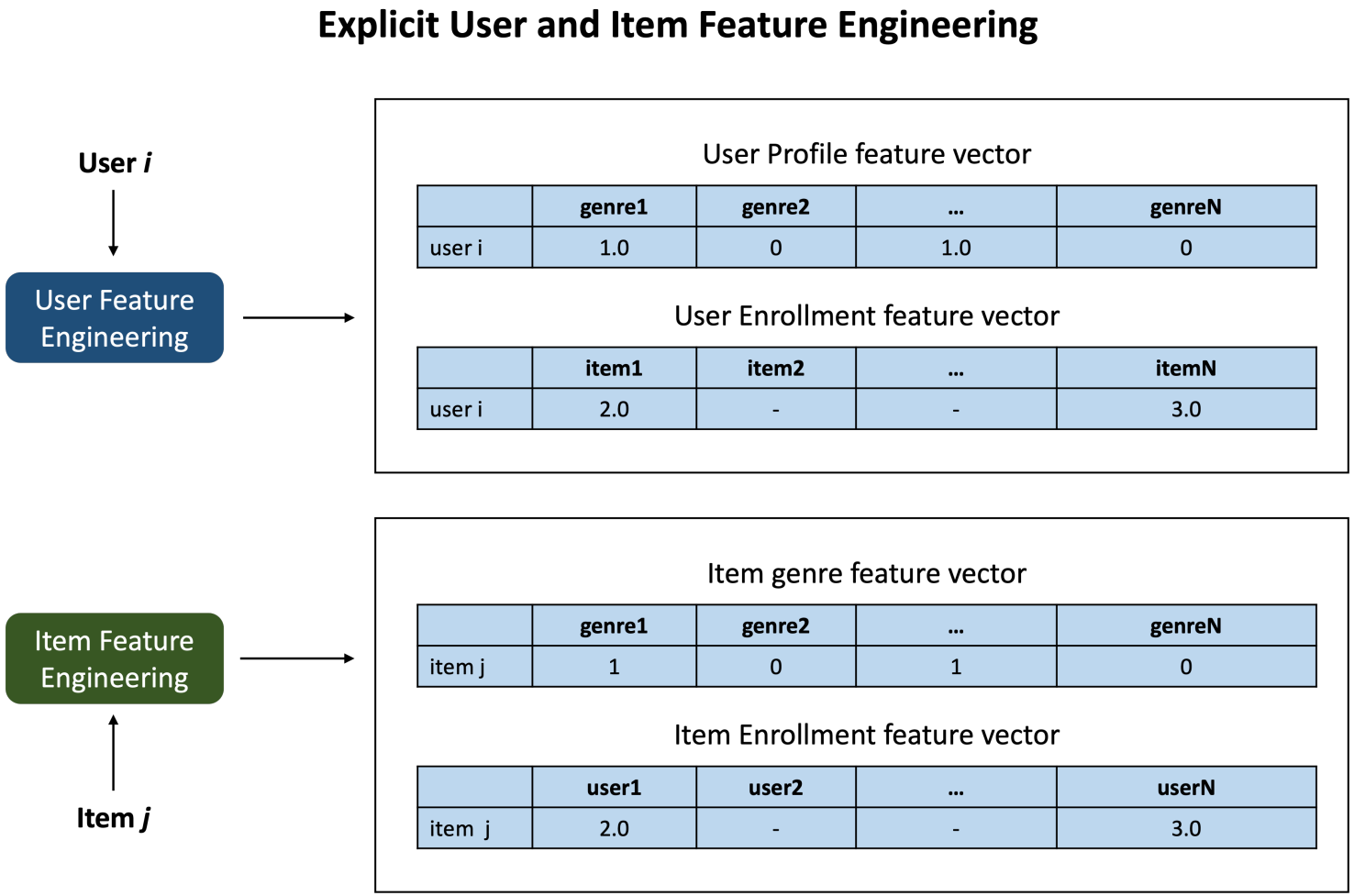
	feature1	...	feature16
user1	...	...	...
user2	...	...	...
user3	...	...	...
user4	...	...	...
...	...	...	...
...	...	...	...
user6	...	...	...

X

Item matrix: **I** 16 x 100

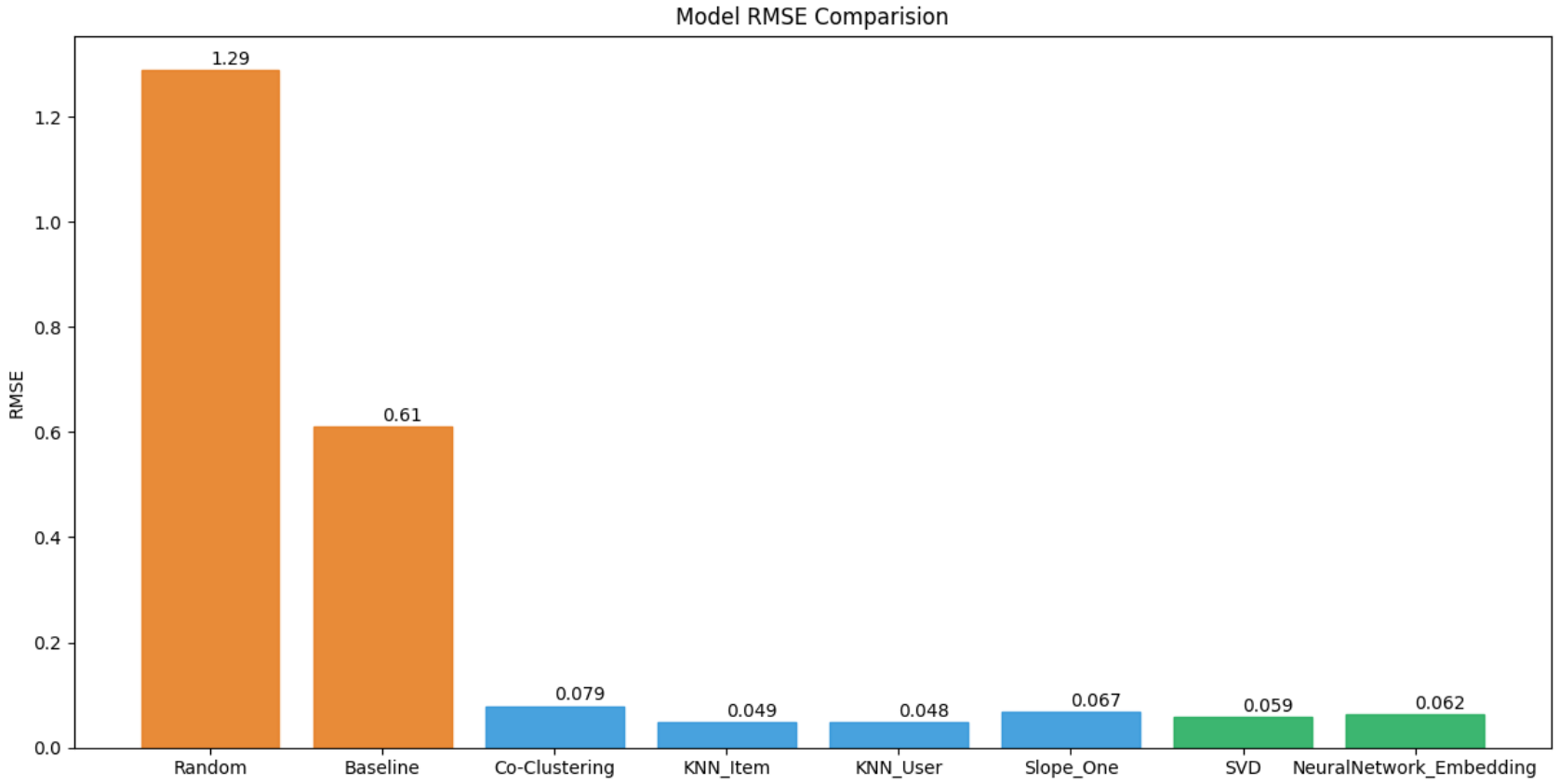
	item1	...	item100
feature1	...	...	...
feature2	...	...	...
...	...	...	...
feature16	...	...	...

Flowchart of Neural Network Embedding based recommender system

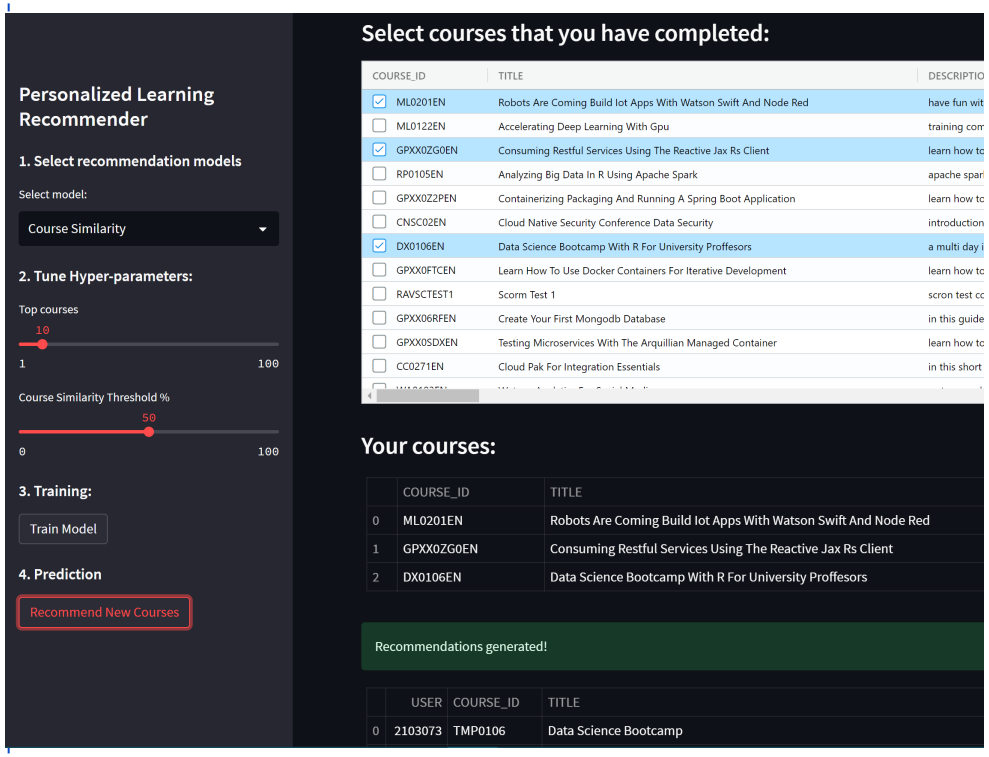


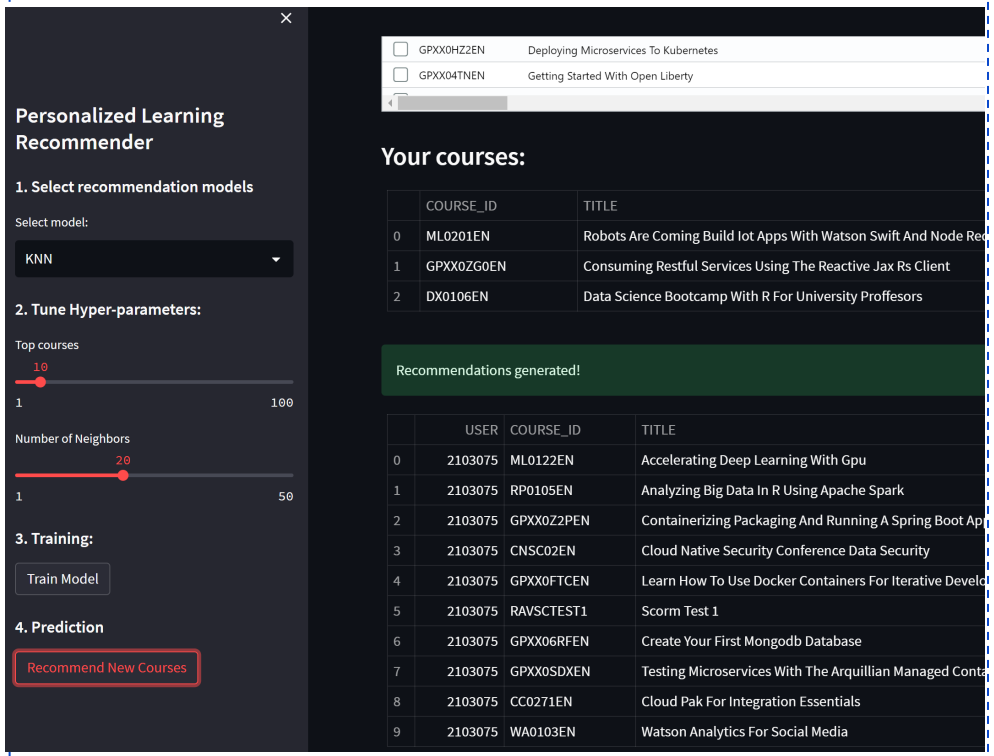
Compare the performance of collaborative-filtering models

Visualization of the performance metric (such as RMSE) of different collaborative-filtering models built so far



Optional: Build a course recommender system app with Streamlit





Conclusions

ANALYZING EXPLORATORY DATA

THE CONSTRUCTION OF A RECOMMENDER SYSTEM THAT UTILIZES COLLABORATIVE FILTERING AND SUPERVISED LEARNING TECHNIQUES.

BUILDING RECOMMENDER SYSTEM BASED ON COLLABORATIVE FILTERING AND SUPERVISED LEARNING

WE CAN ATTRACT MORE STUDENTS AND IMPROVE THEIR LEARNING EXPERIENCE BY FACILITATING THE DISCOVERY OF NEW, CAPTIVATING COURSES AND OPTIMIZING THEIR LEARNING PATHS.

THE INCREASED USAGE OF MY RECOMMENDER SYSTEMS BY STUDENTS TO ENGAGE WITH A WIDER RANGE OF COURSES WILL RESULT IN A RISE IN REVENUE FOR COURSERA.

Appendix

- IBM Machine Learning Professional Certificate
  https://www.coursera.org/professional-certificates/ibm-machine-learning
- My Github
  https://github.com/mvtkop/IBM-Machine-Learning-Professional-Certificate-Capstone-Project