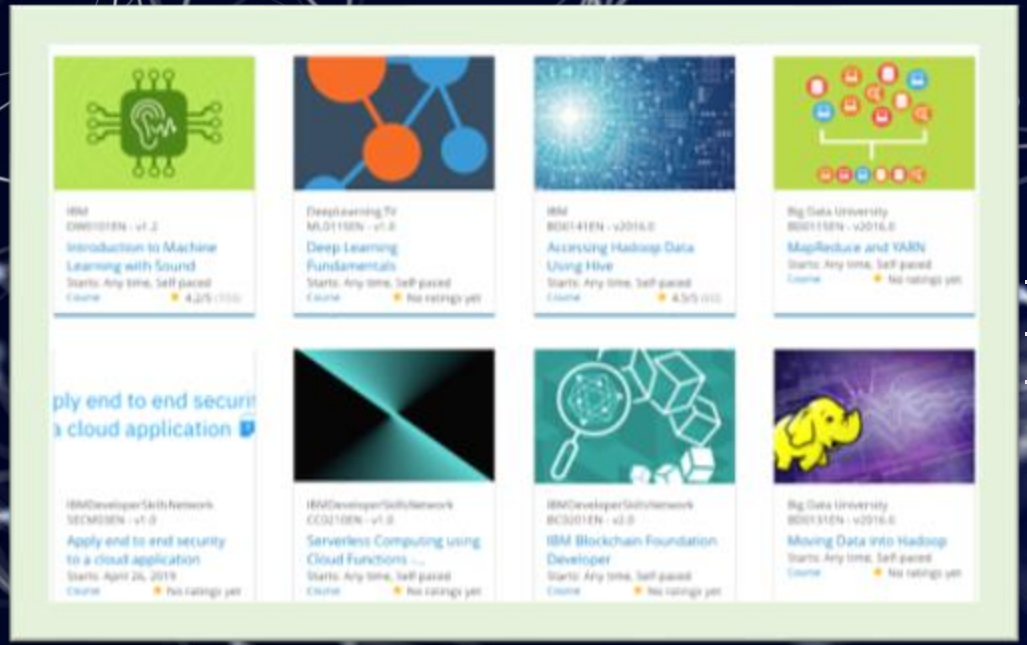
Build a Personalized Online Course Recommender System with Machine Learning

Aung Hein
16-Jun-2024



Outline

Introduction and Background

Exploratory Data Analysis

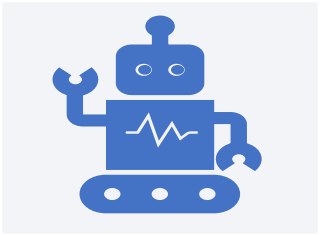
Content-based Recommender System using Unsupervised Learning

Collaborative-filtering based Recommender System using Supervised learning

Conclusion

Appendix

Introduction



Project background and context

In AI Training Room, learners across the world can learn leading technologies such as Machine Learning, AI, Data Science, Cloud, App development, etc. It grows rapidly and reaches millions of learners in a very short period.

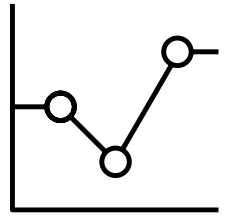
The main goal of this project is to improve learners' learning experience
via helping them quickly find new interested courses and better paving their learning paths. Meanwhile, with more learners interacting with more courses via your recommender systems, your company's revenue may also be increased.

Problem states and hypotheses

Do content-based or collaborative-filtering recommender systems are best suited for us ?

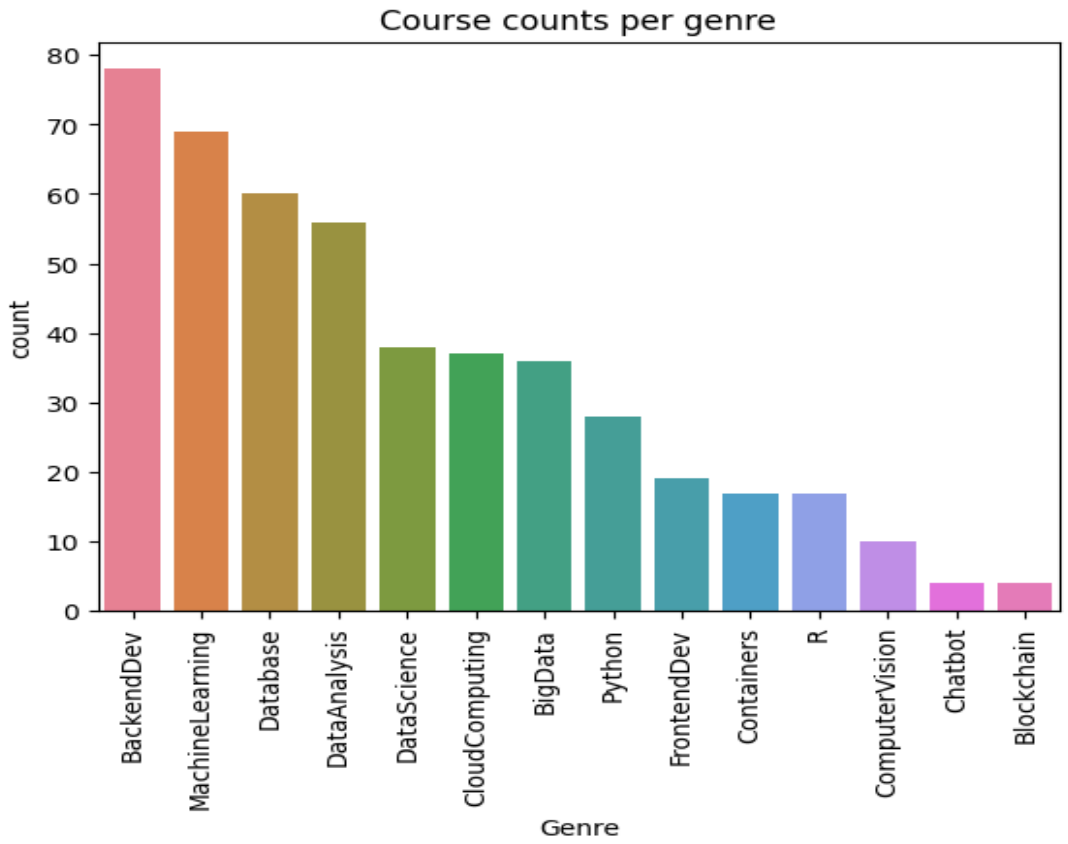
What is the best model for these recommender system?

Exploratory Data Analysis



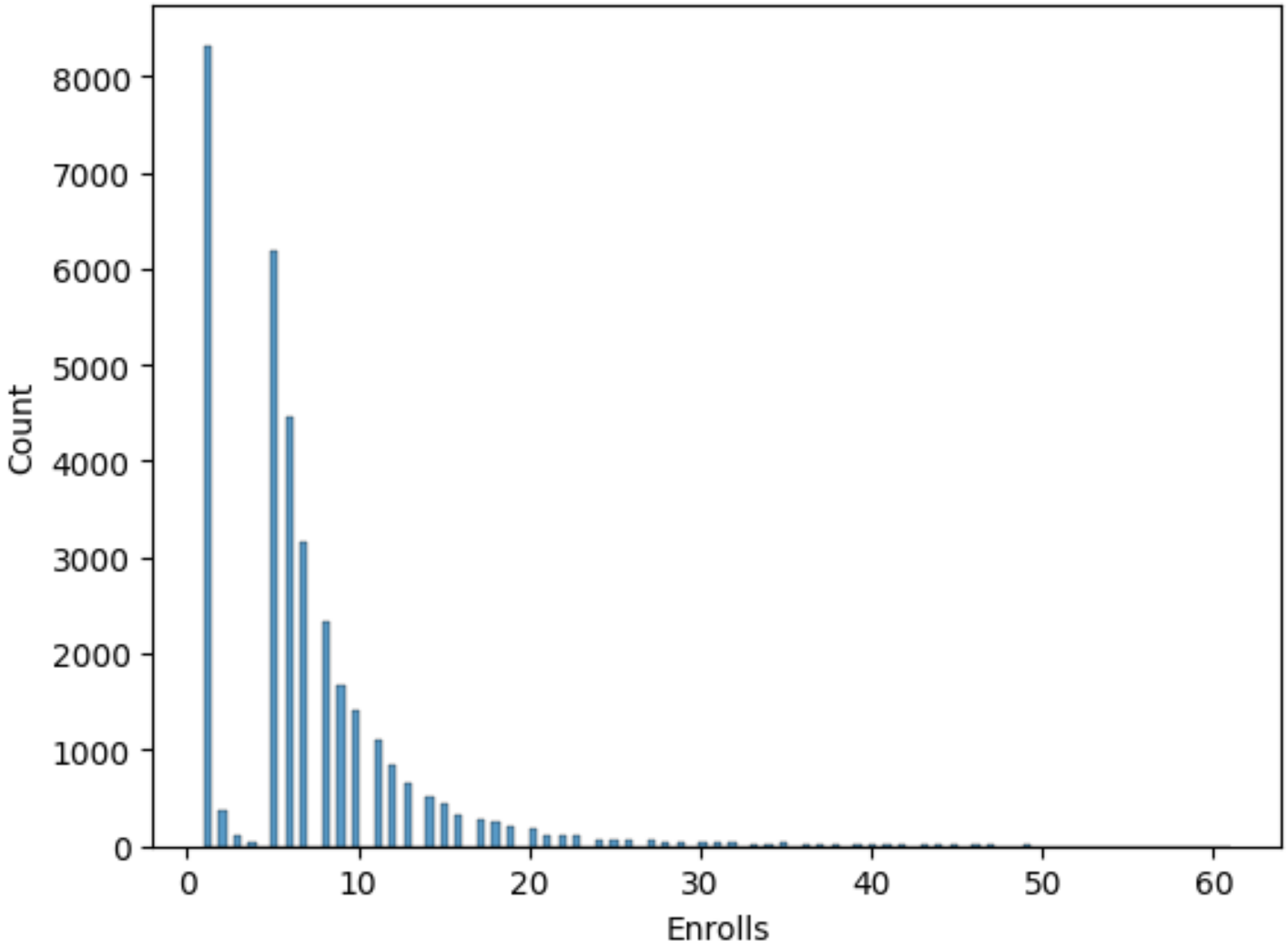
Course counts per genre

- Course Genres Overview
  - Backend Development and Machine Learning seen the most interested genres among the course
  - Chatbot and Blockchain seen now less popular



Course enrollment distribution

- Around 8000 learns seen learned only one courses
- May be they are new-users or we need to improve user's experiences
- On average learners learn about 5-6 course



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

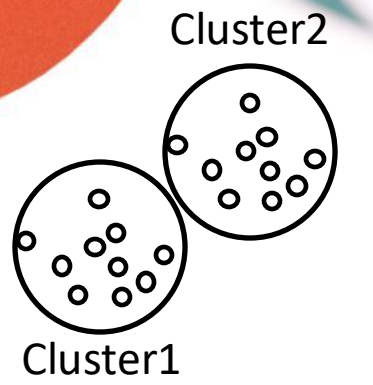
20 most popular courses

- Data Science courses seen quite popular hit top 1 and 2
- Big Data also follow after Data Science

Word cloud of course titles

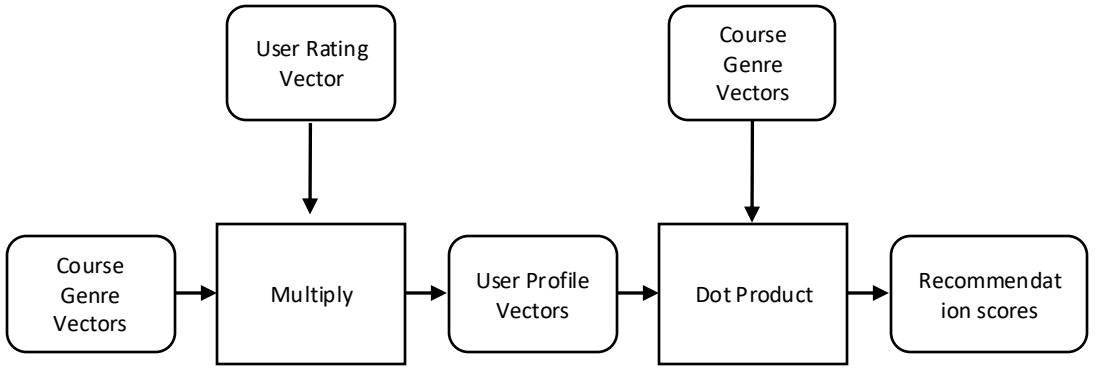
- Data Science, Python and Machine learning seen most demanding skills these days.

Content-based Recommender System using Unsupervised Learning



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

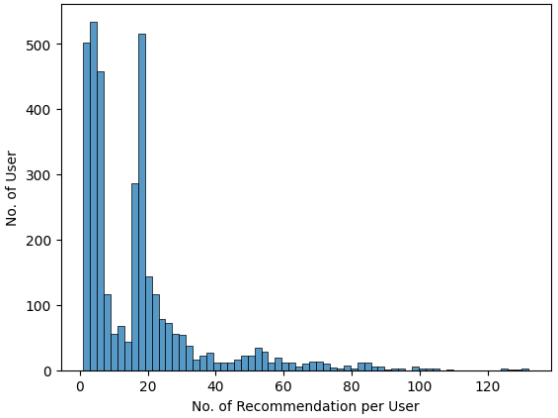
- User Profile was created by multiply course rating to it genre vector.
- So we can know how much learner is interested on each genres.
- Since we know how much learner is interested in each genre, we can calculate course score by dot product of course genre vectors and user profile vector.
- It mean how much learner will interest (recommendation score) on the course base on his interest on genre (profile vector).

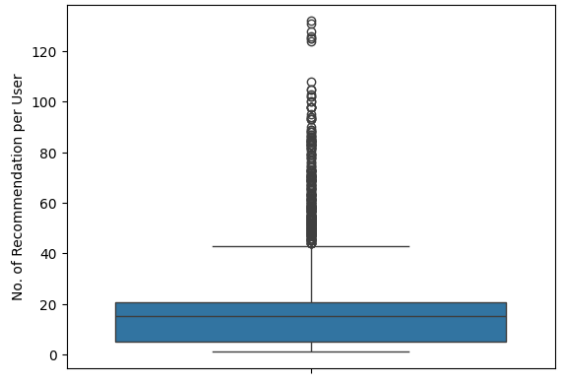


Evaluation results of user profile-based recommender system

Recommendation score = 40

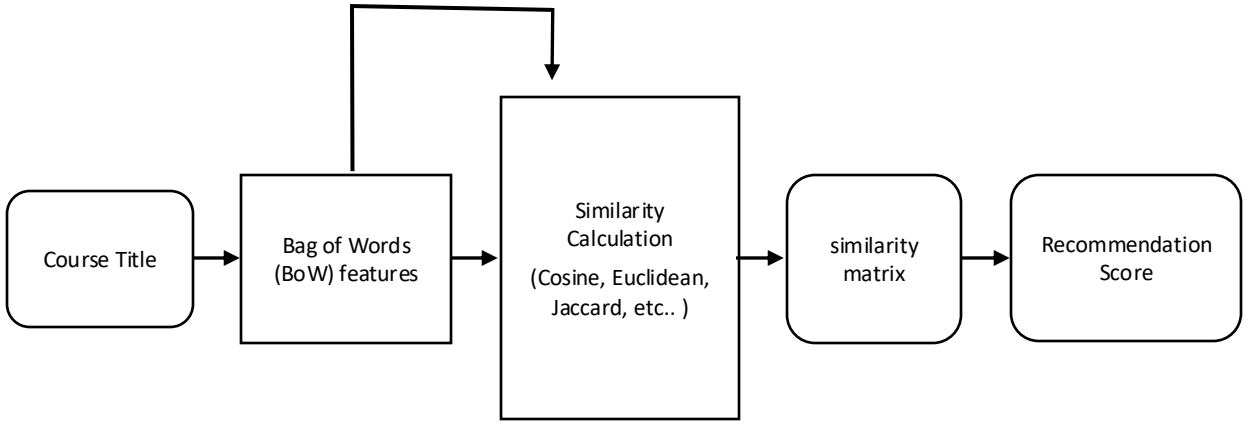
	COURSE_ID	TITLE	No. Recommended
0	excourse72	foundations for big data analysis with sql	2555
1	excourse73	analyzing big data with sql	2555
2	TMP0105EN	getting started with the data apache spark ma...	2406
3	RP0105EN	analyzing big data in r using apache spark	2015
4	excourse31	cloud computing applications part 2 big data...	1912
5	SC0103EN	spark overview for scala analytics	1782
6	excourse70	big data capstone project	1634
7	GPXX097UEN	performing table and crud operations with cass...	1634
8	excourse05	\r\ndistributed computing with spark sql	1634
9	excourse42	big data analysis hive spark sql dataframes...	1634





Flowchart of content-based recommender system using course similarity

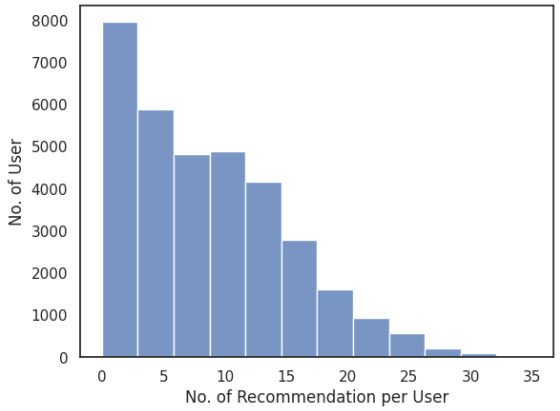
- We will use Bag of Words features to extract contents of course
- Then we can calculate similarity matrix by using similarity calculation such as Cosine, Euclidean, Jaccard index and etc..
- And then we can lookup recommendation score each courses

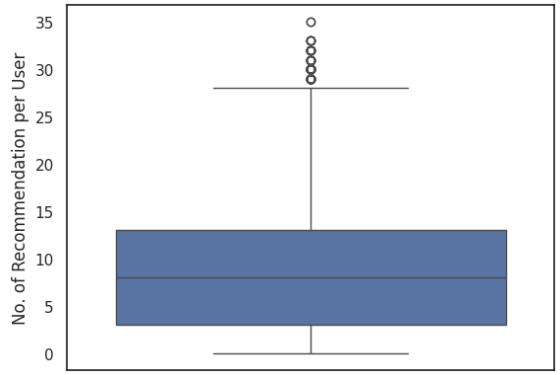


Evaluation results of course similarity based recommender system

similarity threshold = 0.6

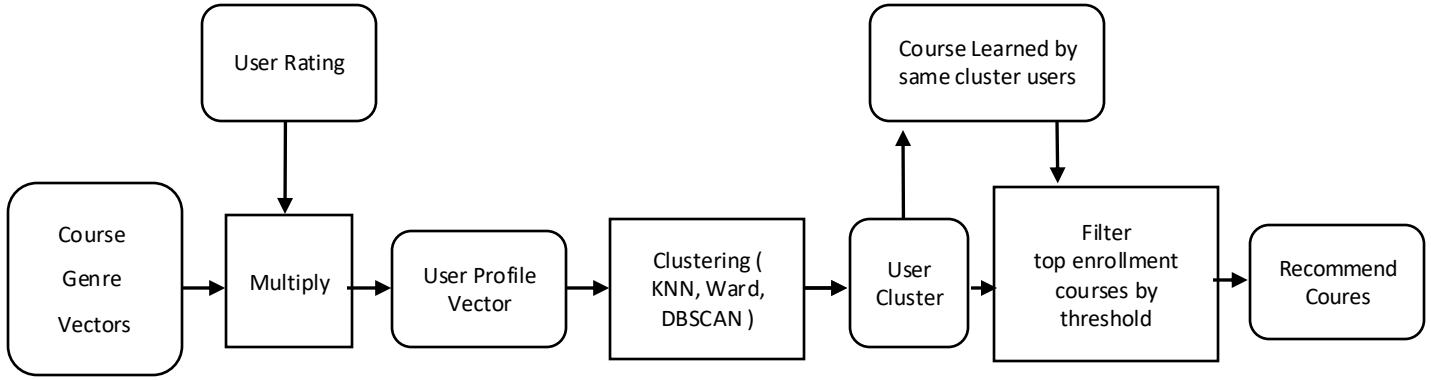
	COURSE_ID	TITLE	No. Recommended
0	DS0110EN	data science with open data	15003
1	excourse62	introduction to data science in python	14937
2	excourse22	introduction to data science in python	14937
3	excourse65	data science fundamentals for data analysts	14641
4	excourse63	a crash course in data science	14641
5	excourse68	big data modeling and management systems	13551
6	excourse72	foundations for big data analysis with sql	13512
7	excourse74	fundamentals of big data	13291
8	excourse67	introduction to big data	13291
9	BD0145EN	sql access for hadoop	12497





Flowchart of clustering-based recommender system

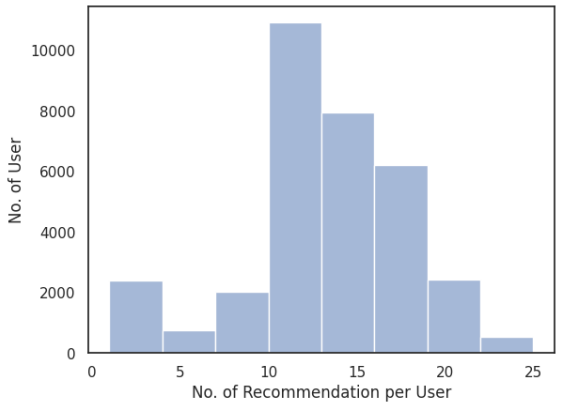
- User Profile Vector was calculated by multiplying course genre vector and user rating
- cluster users by machine learning algorithm such as KNN, DBSCAN etc..
- Course are recommended base on similar courses learned by users in same cluster

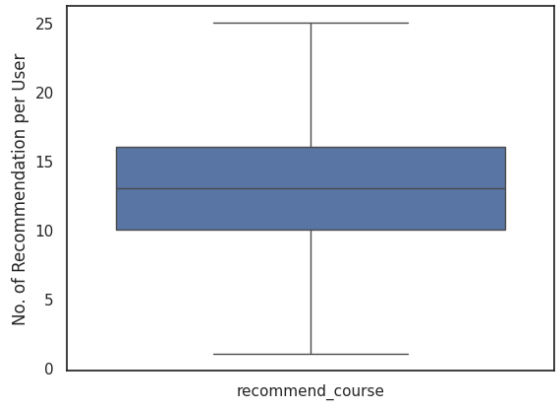


Evaluation results of clustering-based recommender system

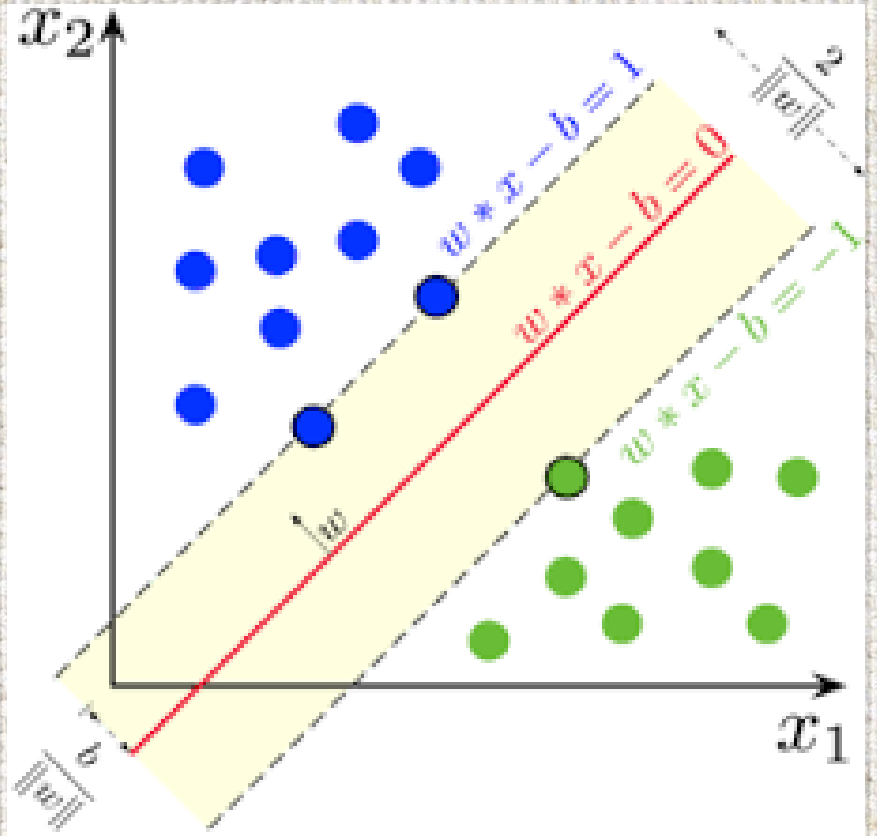
enrollment threshold = 200

	COURSE_ID	TITLE	No. Recommended
181	BD0111EN	hadoop 101	22824
106	ST0101EN	statistics 101	21280
147	BD0101EN	big data 101	20135
176	DS0101EN	introduction to data science	18903
142	ML0115EN	deep learning 101	15909
135	DS0103EN	data science methodology	15255
134	DS0105EN	data science hands on with open source tools	14843
191	DB0101EN	sql and relational databases 101	14463
161	DA0101EN	data analysis with python	14411
102	BD0211EN	spark fundamentals i	14391



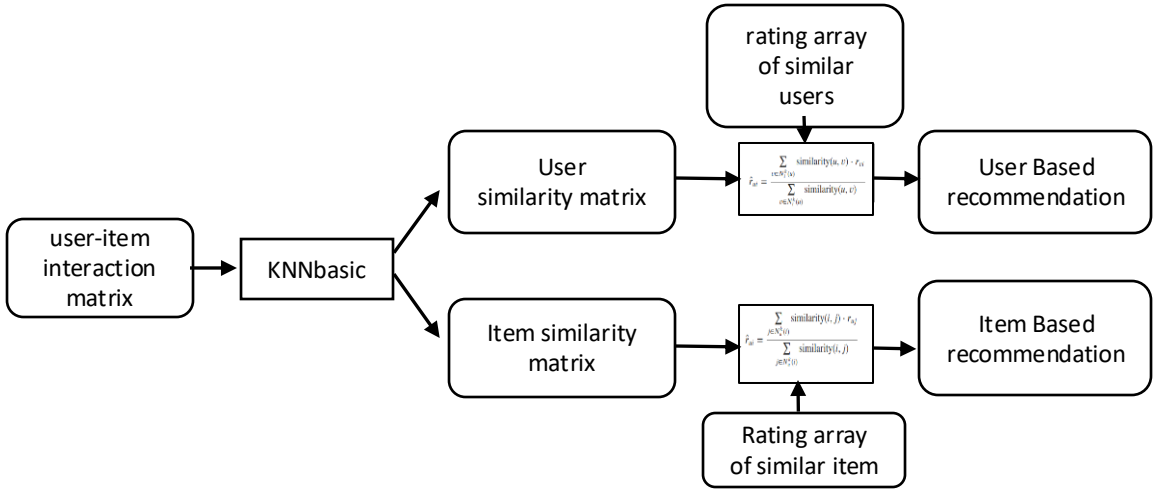


Collaborative-filtering Recommender System using Supervised Learning



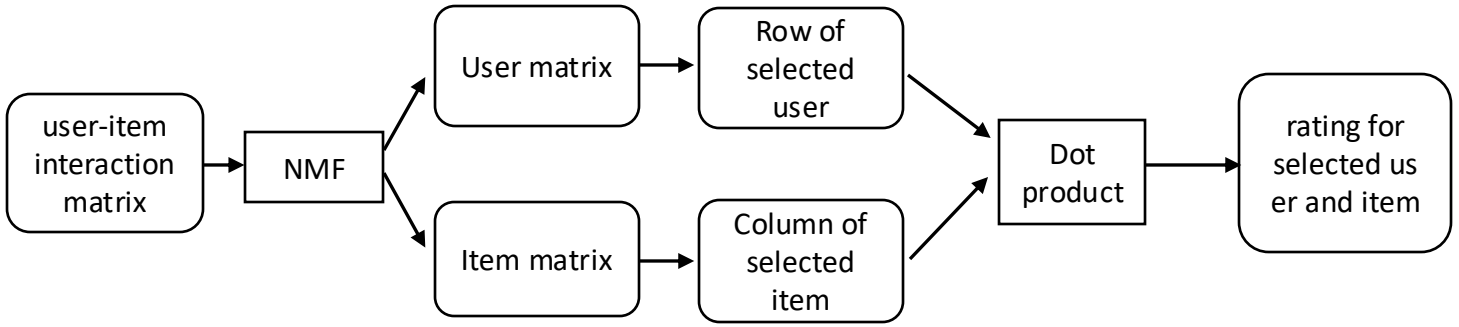
Flowchart of KNN based recommender system

- Similarity was measured by K-NN basic algorithm and can measure using different distance such as (pearson, cosine , and euclidean)
- User base recommendation was calculated by dot product of user similarity matrix and rating array by similar user and divided by total of similarity
- Item base recommendation was calculated by dot product of item similarity matrix and rating array of similar item and divied by total of similarity



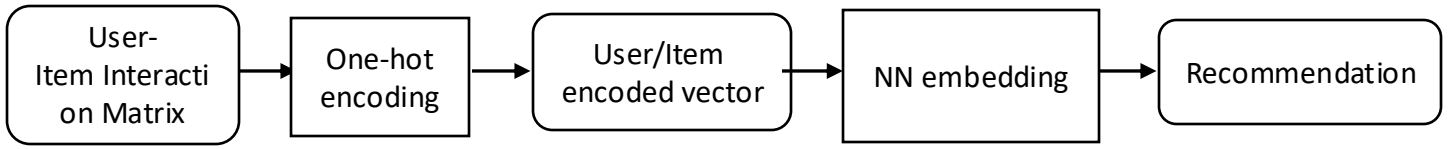
Flowchart of NMF based recommender system

- by using NMF learning on user item interaction matrix, we will get user matrix and item matrix
- And then we can get recommendation by dot product of row of selected user from user matrix and columns of selected item from item matrix



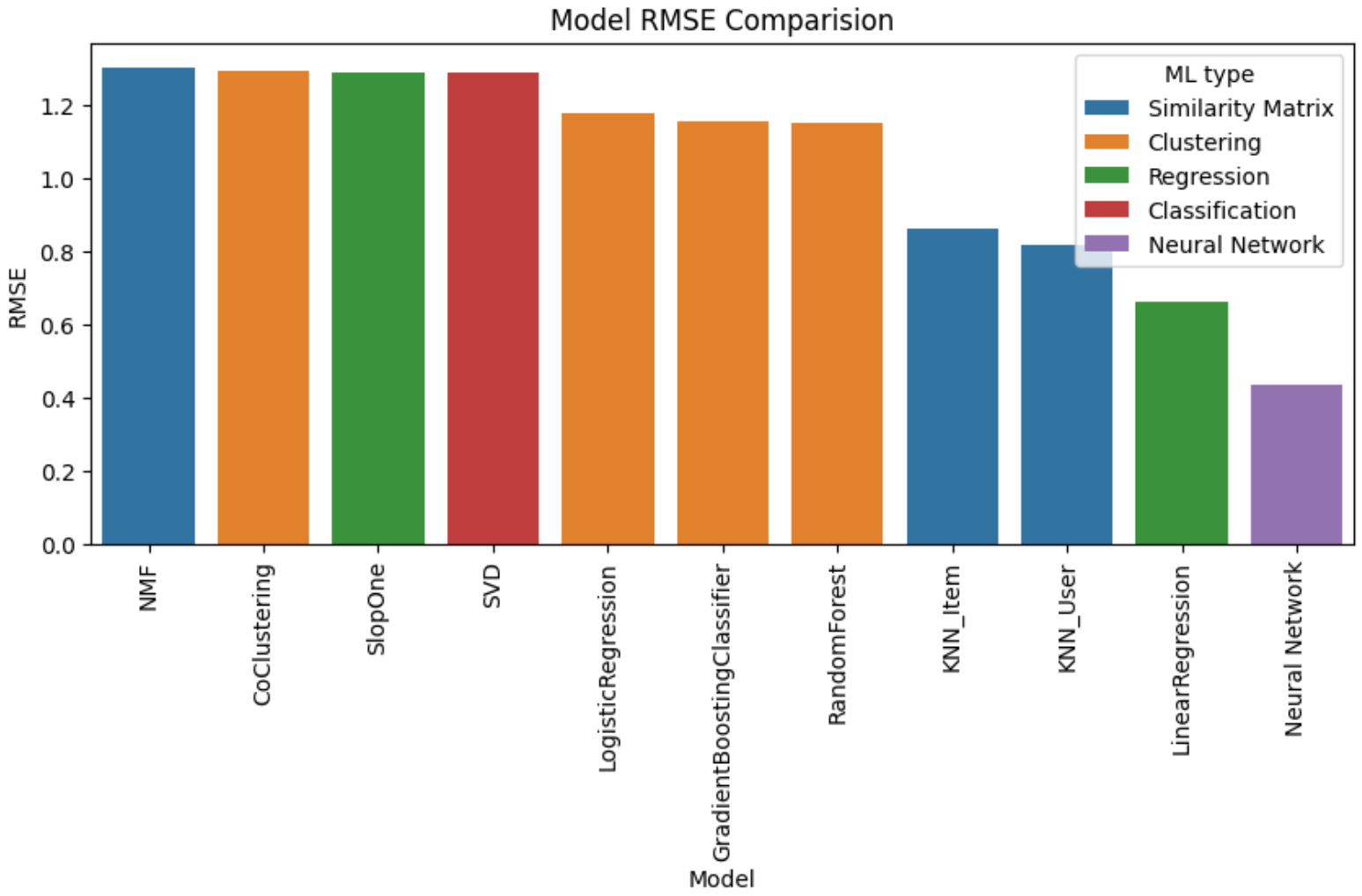
Flowchart of Neural Network Embedding based recommender system

- User vector from user-item matrix are one-hot encoded and also item vector from user-item matrix are one-hot encoded
- User vector are trained to user-embedding layer and item are trained to item-embedding layer
- Recommendation can obtain by dot product of user-embedded vector and item-embedded vector



Compare the performance of collaborative-filtering models

- Models from Suprise library are among highest RMSE score and maybe we need further tuning
- KNN are the best among Suprise Models but it is memory intensive
- Linear Regression also got better rmse but further tuning require for overfit
- Neural Network are the best scored model



Conclusions

- Content base recommendation will have a lot of outlier (number of recommend courses) due to multiple similarity to user or course
- Cluster base recommendation system don't have outlier if we have higher threshold
- KNN got better RMSE than NMF,SVD but it is memory intensive and so only can train with smaller dataset
- Linear Regression also got better RMSE but it seen overfitted to training due to negative r2 score, further tuning may require
- Neural Network are the best scored and by further tuning and testing, we should deploy NN model as recommender system

Appendix

- GitHub Repo : https://github.com/thonenyangal/Recommender-System.git