TEST Engineering & Management

May - June 2020
ISSN: 0193-4120 Page No. 4108-4114

Towards Recommending Courses in a Learner-Centered System based on Trend, Faculty and Student Course Preferences

[1]Abdulrahman Mohsen Zeyad, [2]Shantala Devi Patil

[1,2]School of Computing and Information Technology, REVA University, Bengaluru, India
[1]amamzeyad@gmail.com, [2]shantaladevipatil@reva.edu.in

Article Info
Volume 83
Page Number: 4108-4114
Publication Issue:
May - June 2020

Article History
Article Received:19 November 2019
Revised: *27 January 2020*
Accepted: *24 February 2020*
Publication: *12 May 2020*

Abstract

Recommendation systems have become essential today in different areas of life, they help learners to find content in large sets. Also, the recommendation engines can display the elements that users may not have thought of searching on their own and users get never-expected results. People today use search engines to look for products, tomorrow they will just explore the proposals submitted. This system aims to enhance student`s skills and provide them with training courses to raise their opportunities for good careers. Students assessments are traditional methods to predict student`s performance such as failing or passing or forecasting successful completion of the course, in this continuation, predicting the classification of degree or achievement. This paper discusses the course of their interests and proposes course selection assistance through a recommendation system that may help students make the right choices through experienced support.

Keywords: *Pearson correlation coefficient, Prediction function, Content based, Collaborative filtering, Hybrid.*

1. Introduction

During the last few decades, with the rise of Youtube, Amazon, Netflix[1][2][3] and many other such web services, recommender systems have taken more and more places in our lives. From e-commerce (suggest to buyers' articles that could interest them) to online advertisement (suggest to users the right contents[2], [4], [5], matching their preferences), recommender systems are today unavoidable in our daily online journeys.

In a very general way, recommender systems are algorithms aimed at suggesting relevant items to users (items being movies to watch, text to read, products to buy or anything else depending on industries)[2]–[4], [6]–[9].

Through the various research papers, the concept of recommendation system and its applications was understood. I got to know the various kinds of recommendation systems[10] content-based[2], [4], [5], [7]–[9], [11], [12], collaborative[1], [2], [12]–[15], [3]–[9], [11] and hybrid[1], [3], [5]–[9], [11], [13], [15], in those papers, various methods were discussed, and experiments were conducted with real datasets to assess the overall performance of the proposed approac[2]–[5], [8]–[10], [13]–[15].Also, they have calculated the faculty expertise because faculty also plays a great role as they teach those subjects. Jeff Hale[16] proposed a recommendation system for LinkedIn.com which recommends courses to users, the system works on network analysis of rating constructed on LinkedIn data using rating prediction. The Pearson correlation coefficient (also referred to Pearson's r)[1], [3], [6]–[8], [11], [12], [17] r is the most common measure of correlation and has been widely used in the sciences as a measure to determine the relationship between two quantitative[10] variables (interval/ratio) and the degree to which the two variables coincide with one another—that is, the extent to which two variables are linearly[6], [9], [17] related: changes in one variable correspond to changes in another variable or measure of the degree of linear dependence between two paired data. Have been developed over the years for measuring[8] relationships between sets of data, this project uses the Pearson correlation coefficient, and specifically the types and assumptions of a correlation. It also explains correlation computation[2], [3], [7], [8], [10] and interpretation.

Educational needs vary from student to student and from level to level based on the profession objectives and skills gap. Success begins with a plan and support from those with previous experience, with what students bear

TEST Engineering & Management

May - June 2020
ISSN: 0193-4120 Page No. 4108-4114

in terms of their academic plans, and they are more likely to graduate on time and find success in the labor market in which competition and demand for skills to cover market needs are increasing, Skills-based recruitment is an employment [8], [13] management approach that empowers employers to match employment [8], [13] around business outcomes and begins with company owners who identify the special skills required for the role, and then examine and evaluate the competencies of candidates with those requirements. However, there is no one-size-fits-all solution for student success [5]. So that people behave intelligently [9], [10] and make the right choice. Students assessments are traditional methods to predict student performance such as failing or passing or forecasting successful completion of the course, in this continuation predicting the classification of degree or achievement.Students can use the system recommendation functions to determine the best related courses that will be of great benefit for theirfuture in the competitve job market.

The "Pearson Correlation Coefficient [1], [3], [6]–[8], [11], [12], [17]" was used to find students who rated common subjects, and fifty common highest rated subjects have been chosen and arranged in descent order, out of which five subjects will be recommended for the user through the "Prediction Function [3], [10]". Both the subjects rated by students and the five subjects recommended to the user come from external csv data file, and the same file is also used to repeat the recommendation process using the Pearson Correlation Coefficient [1], [3], [6]–[8], [11], [12], [17] and the Prediction Function [3], [10], but this time depending on the teacher`s ratings instead of the student`s ratings and depending on the current trends as well.

2. Methodology

2.1 We Used

Software:

- OS: Windows 10 pro 64bit
- Python:(Pandas and Numpy python library)
- Anaconda3 Spyder (It is nothing necessary, just a personal preference)
- Microsoft Excel is used to review and identify data

Hardware:

- Processor: Intel i5-6440HQ 2.6GHz
- RAM: 12 GB
- SSD: PCI-E NVMe M.2

2.2 Data

Datasets [18]:

- Kaggle: is an online community of data scientists and machine learners, owned by Google.

Files:

- Ratings.csv (course_id, user_id, ratings)
- Corsus.csv (course_id, books_count, work_id, author, title, etc.)
- Ratings of 37 courses from 53418 user+ from above csv file.

2.3 Data Filtering

We kept

- Courses that have at least 30 ratings from users
- Users that have rated at least 5 courses

Removed Duplicates

Initial Data

- 37 courses
- 53418 Users

2.4 Algorithms:

- Pearson correlation coefficient[1], [3], [6]–[8], [11], [12], [17]
- Prediction function[3], [10]

The methods to be used in this paper include the following, building a recommender system, trying to solve the problem of a correct user model and how to link the fitting curriculum content to it. In this system, the user goes through six different stages:

Phase 1: User Registration

Phase 2: User Rating

Phase 3: Mixed Recommendation approach

Phase 4: Forecasting as similarity

Phase 5: List of a recommendation of courses from similar users

Phase 6: List of a recommendation of courses from faculty and Trend.

Course rating is also must during the registration process.

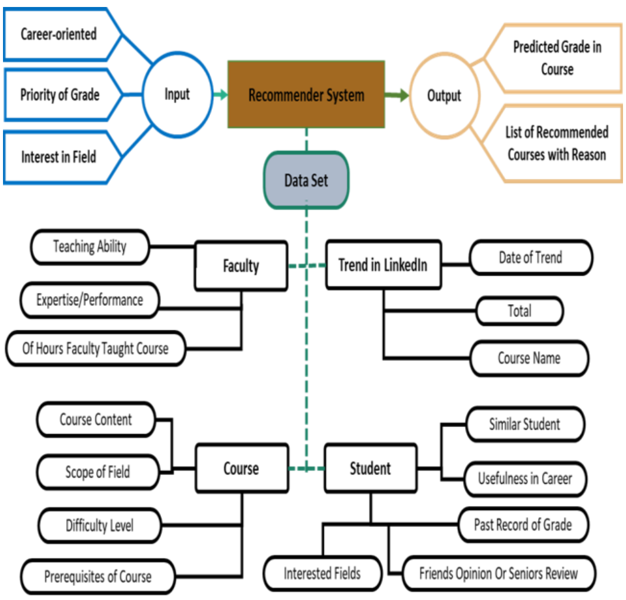

Figure 1: Block Diagram.

May - June 2020
ISSN: 0193-4120 Page No. 4108-4114

3. Related Works

Correlation Analysis

Pearson's method, popularly known as a Pearsonian Coefficient of Correlation, is the most extensively used quantitative[10] method in practice. Its value ranges from [1, -1]. When the correlation coefficient 1, it becomes the link is entirely positive; when equal to -1, It becomes linked to the negative completely. That means the higher the absolute value of the correlation coefficient, the stronger the correlation and vice versa[4], [14]–[16]. The coefficient of correlation is denoted by "r". Degree of association > measured by the correlation Coefficient, Coefficient> measure of linear association.

- a : Current user
- b :Target user
- r : Rated items
- $\bar{r}$: Unrated items

1. Find users that are eligible to be a similar user. A user can be a similar user if both of the followings are true (we have not considered recursive collaborative filtering[1], [2], [12]–[15], [3]–[9], [11] in our system):

a. A user can be a similar user if he has rated at least three item that have been rated by the target user as well.

b. A user can be a similar user if he has also rated the unrated item of the target user.

2. If a user can be a similar user, similarity is calculated using Pearson correlation:

- For a user with some items without ratings
- Find 30 most similar users by traversing the content matrix[10].
- Pearson correlation
- Calculate rating for a missing rating
- Prediction function[3], [10]

$$pred(a,p) = \bar{r}_a + \frac{\sum_{b \in N} sim(a,b) * (r_{b,p} - \bar{r}_b)}{\sum_{b \in N} sim(a,b)}$$

$$sim(a,b) = \frac{\sum_{p \in P}(r_{a,p} - \bar{r}_a)(r_{b,p} - \bar{r}_b)}{\sqrt{\sum_{p \in P}\left(r_{a,p} - \bar{r}_a\right)^2}\sqrt{\sum_{p \in P}\left(r_{b,p} - \bar{r}_b\right)^2}}$$

3. Once similarities have been calculated between the target user and all possible similar users, from those available similar users top 50 similar users are selected to calculate rating for that particular unrated item. Rating has been calculated using the following prediction function[3], [10]:

a. Repeat for each item without rating

b. Recommend top 5 items

In our recommender system, for a user, for each unrated item of that user steps 1 to 3 described above are repeated and a rating is generated for each unrated item. However, if for an unrated item there is no similar user found against target user (meaning no one has rated at least one item same as the target user or has rated at least one item as the target user, but has not rated that particular unrated item), then that unrated item will not get any rating on that run of the system. If content matrix gets updated enough after some recommendation's generation, then the a, b checks in step 1 will be true at some point and rating will be generated for that unrated item of that user as well later on. The worst-case scenario is for a target user no recommendations is generated. This will happen if for that target user for none of the unrated items there is a similar user.

We have numerical ratings for courses provided by users.
ratings.csv contains ratings, course id and user id. *courses.csv* contains the original title of the course and multiple other attributes (authors, frequency of rating 1, 2, 3 and so on).

We didn't need to use the other attributes.

Data contains 37 courses and 53418 users.

However, we removed duplicates and filtered out some data. We didn't change our data file.

Every time the project is run these filtering steps will be taken and we will use 37 courses and 53418 users to demonstrate the recommendation system.

We filtered users to the following:

Divided into several batches as below table.

In the first batch only, we reduced the number of users and evaluations as follows:

1. Maintain training sessions with at least 15 users
2. Maintaining users with a minimum of 10 courses

For the remaining batches:

1. Keep courses for at least 30 users
2. Maintaining users with at least 10 courses

Table 1: Divisions of batches

Batch	Number of rows	Number of users
1	1K	107
2	10K	828
3	35K	2549
4	70K	53418

A sample of users was chosen with their numbers (7, 10, 82, and 105). Their expected courses were recorded based on the batches of the above table, and the sample was added to user number 992 for batch 3 and 4 because it did not appear in batches 1 and 2.

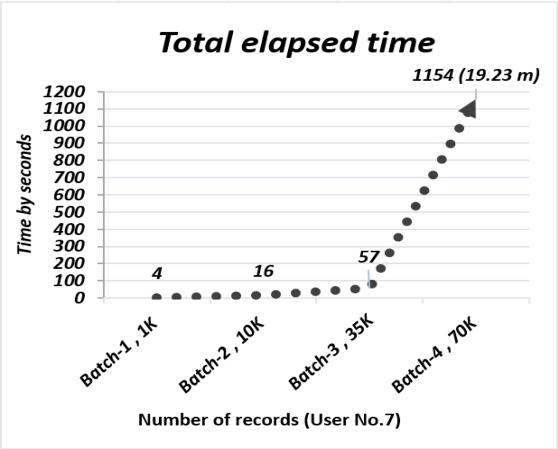

Figure 2: Total time of user No.7.

The elapsed time it takes for every batch varies, depending on the number of records that depend on user ratings, so there is a relationship between the time taken and the number of records filtered, and matrix created from the filtered data. Implement collaborative filtering[1], [2], [12]–[15], [3]–[9], [11] based on the data in the content matrix[10].

Here only the user data No. seven has been viewed and the expected courses are tracked for him in all batches based on the user's own assessments. It is clear in the picture below that the expectation remains constant if the batch number is fixed with no data updates will show the same results. The results will change based on the size of the records and on updating other users' ratings, meaning that user NO, seven ratings will be updated, and new results expected, based on updates to his file or files of other users.

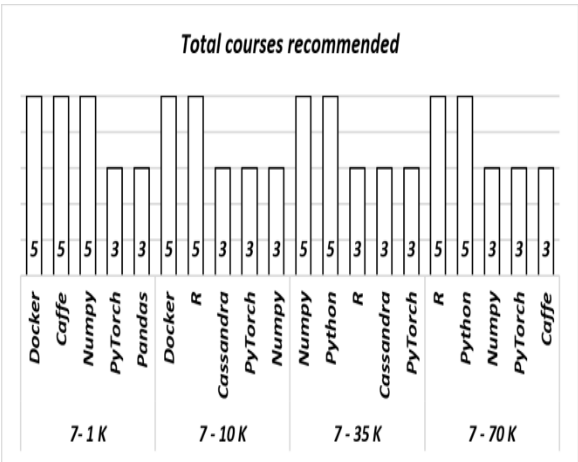

Figure 3: Clustered column user No. 7 course rating.

Here only user No. seven data has been viewed and expected courses are tracked for him in all batches based on trend rating of courses assessed from Linkedin.com the annual LinkedIn U.S. Emerging Jobs Report website shining as a spotlight on jobs experiencing tremendous growth and examining what these trends mean for the courses pictured below[16].

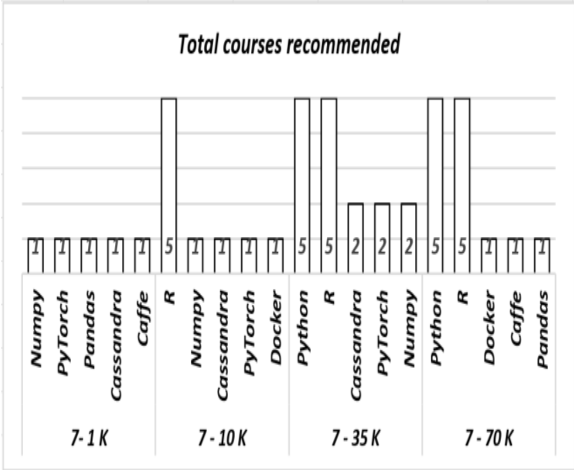

Figure 4: Clustered column user No. 7 course trend

Here only user data No. seven has been viewed and expected courses for him are tracked in all batches based on the ratings of faculty members for the valuable input data in the picture below, where it was approved using random numbers for assessments from user and trend, since faculty's actual ratings are not available they have been created and assumed for the purpose of completing this paper.

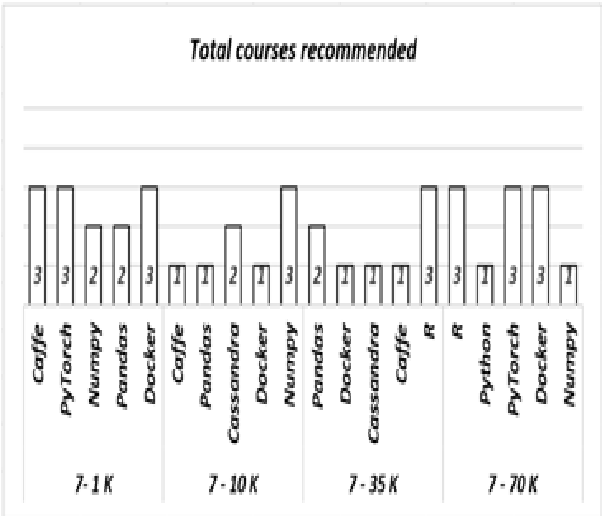

Figure 5: Clustered column user No. 7 course faculty

Only here all user data No. seven as (course rating, course trend, and course faculty) as a one grouped figure has been viewed and expected courses are shown in the image below.

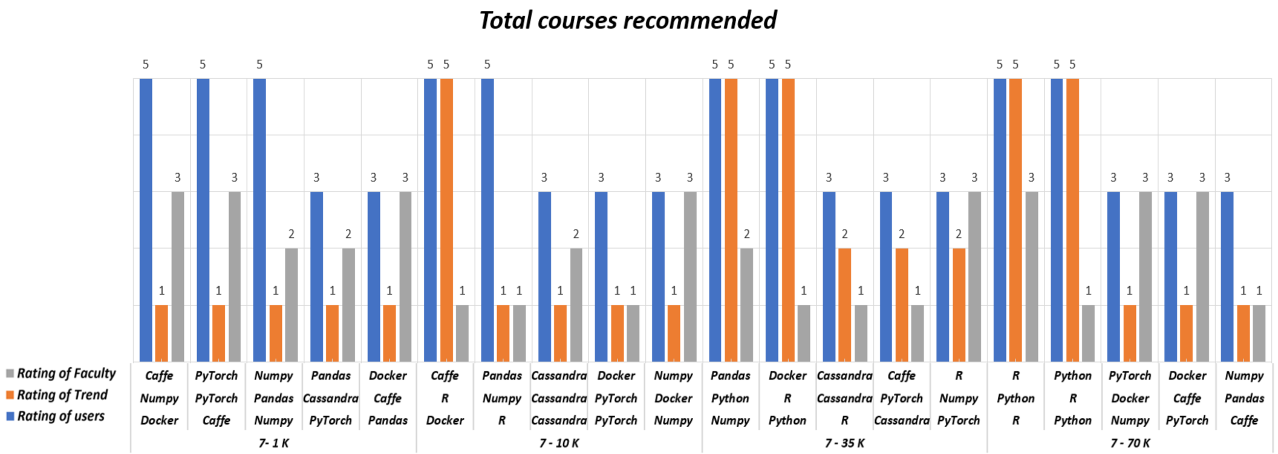

Figure 6: Clustered column user No. 7 course of (rating, trend and faculty)

4. Result and Discussion

- There is a directly proportional relation between the number of records involved in the prediction and the time needed to output the prediction.
- There is also a directly proportional relationship between the number of students who rated same subjects and the time taken to output the prediction too.
- The content matrix [10]. is made of all the subjects rated and not rated, and this matrix is used to recognize the students who rated subjects and the subjects which they rated.
- No recommendations can be made if users did not finish their profile data, which mandates that each user must rate minimum of three subjects initially.
- Also, no recommendations can be made if there is no match between the subjects rated by students and the subjects chosen by the user asking for recommendation. Hence, different subjects and different ratings must be periodically updated based on predefined intervals.

Figure 7: Block Diagram

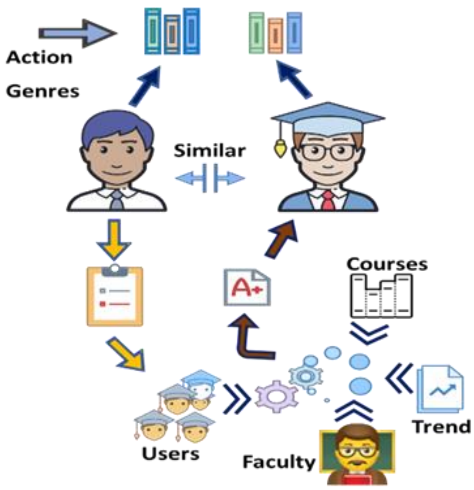

Figure 8: Block diagram of a similar users

- Users can seek assistance for course recommendation through this system.
- The recommendations returned by the system are divided into three separate lists, each one is organized in descent order. The first list is for user`s ratings, the second list is for the current trends, and the last list is for the faculties’ ratings.

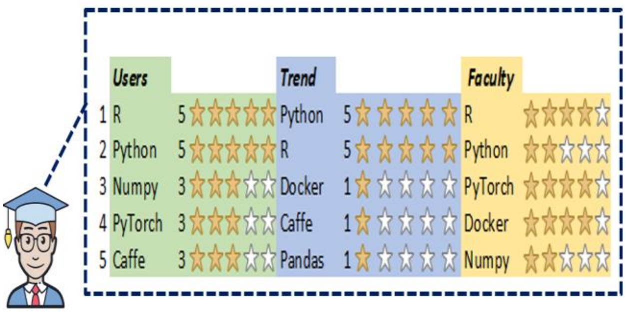

Figure 9: Recommendation for user No.7

5. Conclusion

To sum up, users can try to naturally anticipate and predict the outcomes of them taking specific courses based upon their knowledge of their own capabilities and tendencies such as persistence to complete courses, efforts put into studying, and chances of passing exams. The system suggested is a recommendation system, which will recommend elective courses[5], [12], [13], [15] to the users based on the recommendation variables such as student's ratings, teacher's ratings and the current trends, which will be provided as inputs to the system. Furthermore, we will mention some of our ideas that can be further developed from this point. Embedding prioritization in final recommendations before selecting the top five. The idea is to figure out which course trend in the market is more well rated by a user for example "emerging jobs report India" from LinkedIn. Then sort the generated recommendations based on their course preferences. Another is to filter the recommended books by doing content-based[2], [4], [5], [7]–[9], [11], [12], filtering against already highest rated books by the user, then recommend the top 5 books, also with showing the top five comments about the course from students who have commentedto closer to rated courses.Such information can be further collected and gather through trends site in auto or manual update, like the site (monster.com, Indeed.com, simplyhired.co.in, etc.[16]).

Acknowledgment

I would like to express my deep gratitude to Dr. Shantala Devi Patil , my research supervisors, for their patient guidance, enthusiastic encouragement and useful critiques of this research work. I would also like to thank Dr. Ashok K Patil, for her advice and assistance in keeping my progress on schedule.

References

[1] N. Dharaneeshwaran, S. Nithya, A. Srinivasan, and M. Senthilkumar, "Calculating the user-item similarity using Pearson's and cosine correlation," Proc. - Int. Conf. Trends Electron. Informatics, ICEI 2017, vol. 2017-Janua, pp. 1000–1004, 2017.

[2] G. Linden, B. Smith, and J. York, "Amazon.com recommendations: Item-to-item collaborative filtering," IEEE Internet Comput., vol. 7, no. 1, pp. 76–80, 2003.

[3] K. K. Fletcher and X. F. Liu, "A Collaborative Filtering Method for Personalized Preference-Based Service Recommendation," Proc. - 2015 IEEE Int. Conf. Web Serv. ICWS 2015, pp. 400–407, 2015.

[4] G. Koutrika, B. Bercovitz, R. Ikeda, F. Kaliszan, H. Liou, and H. Garcia-Molina, "Flexible recommendations for course planning," Proc. - Int. Conf. Data Eng., pp. 1467–1470, 2009.

[5] R. Obeidat, R. Duwairi, and A. Al-Aiad, "A Collaborative Recommendation System for Online Courses Recommendations," Proc. - 2019 Int. Conf. Deep Learn. Mach. Learn. Emerg. Appl. Deep. 2019, pp. 49–54, 2019.

[6] Y. Wen, F. Sheng, R. Li, B. Zhang, G. Feng, and X. Sun, "Item recommendation based on heterogeneous information networks with feedback information," 2019 Int. Conf. High Perform. Big Data Intell. Syst. HPBD IS 2019, pp. 61–67, 2019.

[7] D. Margaris, D. Vasilopoulos, C. Vassilakis, and D. Spiliotopoulos, "Improving Collaborative Filtering's Rating Prediction Coverage in Sparse Datasets through the Introduction of Virtual near Neighbors," 10th Int. Conf. Information, Intell. Syst. Appl. IISA 2019, pp. 1–8, 2019.

[8] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," IEEE Trans. Knowl. Data Eng., vol. 17, no. 6, pp. 734–749, 2005.

[9] S. Nath and E. Selvam, "A Pragmatic Review on Different Approaches Used in E-Learning Recommender Systems," 2018 Int. Conf. Circuits Syst. Digit. Enterp. Technol. ICCSDET 2018, pp. 1–4, 2018.

[10] C. Aggarwal, Recommender Systems. 2016.

[11] S. M. Nafea, F. Siewe, and Y. He, "A Novel Algorithm for Course Learning Object Recommendation Based on Student Learning Styles," Proc. 2019 Int. Conf. Innov. Trends Comput. Eng. ITCE 2019, no. February, pp. 192–201, 2019.

[12] K. Bhumichitr, S. Channarukul, N. Saejiem, R. Jiamthapthaksin, and K. Nongpong, "Recommender Systems for university elective course recommendation," Proc. 2017 14th Int. Jt. Conf. Comput. Sci. Softw. Eng. JCSSE 2017, 2017.

[13] E. S. Khorasani, Z. Zhenge, and J. Champaign, "A Markov chain collaborative filtering model for course enrollment recommendations," Proc. - 2016 IEEE Int. Conf. Big Data, Big Data 2016, pp. 3484–3490, 2016.

[14] A. Slim, D. Hush, T. Ojha, C. Abdallah, G. Heileman, and G. El-Howayek, "An automated framework to recommend a suitable academic program, course and instructor," Proc. - 5th IEEE Int. Conf. Big Data Serv. Appl. BigDataService 2019, Work. Big Data Water Resour. Environ. Hydraul. Eng. Work. Medical, Heal. Using Big Data Technol., pp. 145–150, 2019.

[15] R. Verma and Anika, "Applying Predictive Analytics in Elective Course Recommender System while preserving Student Course Preferences," Proc. 6th IEEE Int. Conf. MOOCS Innov. Technol. Educ. MITE 2018, pp. 52–59, 2018.

May - June 2020
ISSN: 0193-4120 Page No. 4108-4114

[16] H. Jeff, “The Most in Demand Skills for Data Scientists,” 2018. [Online]. Available: https://www.kaggle.com/discdiver/the-most-in-demand-skills-for-data-scientists.
[17] J. Liu, Y. Zhang, and Q. Zhao, “Adaptive ViBe Algorithm Based on Pearson Correlation Coefficient,” Proc. - 2019 Chinese Autom. Congr. CAC 2019, pp. 4885–4889, 2019.
[18] K. Jakub, “Open University Learning Analytics dataset Data Set,” 2015. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/Open+University+Learning+Analytics+dataset.
[19] H. Zhu, X. You, and S. Liu, “Multiple Ant Colony Optimization Based on Pearson Correlation Coefficient,” IEEE Access, vol. 7, pp. 61628–61638, 2019.
[20] P. Socha, V. Miskovsky, H. Kubatova, and M. Novotny, “Optimization of Pearson correlation coefficient calculation for DPA and comparison of different approaches,” Proc. - 2017 IEEE 20th Int. Symp. Des. Diagnostics Electron. Circuit Syst. DDECS 2017, no. 1, pp. 184–189, 2017.