Retraction

Retracted: IoT Applications for Recommended Methods of Physical Education Online Course Resources Based on Collaborative Filtering Technology

Wireless Communications and Mobile Computing

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

(1) Discrepancies in scope

(2) Discrepancies in the description of the research reported

(3) Discrepancies between the availability of data and the research described

(4) Inappropriate citations

(5) Incoherent, meaningless and/or irrelevant content included in the article

(6) Peer-review manipulation

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

[1] Z. Zheng, "IoT Applications for Recommended Methods of Physical Education Online Course Resources Based on Collaborative Filtering Technology," *Wireless Communications and Mobile Computing*, vol. 2022, Article ID 2928559, 9 pages, 2022.

1. Introduction

1.1. Recommendation System. The World Wide Web is moving into a fast-growing new era. It is shifted from read-only web to the read-write-execution-concurrency web. This is the motivating force for the development of recommendation systems. A recommender system (RS) is a software application based on machine learning algorithms. It predicts the end users' interest and provides suggestions perfectly interesting for them. Majorly, RS has three types of components, namely, items, users, and transactions, represented in a matrix format which is illustrated in Figure 1. There are four items, four users, and ten transactions presented in it.

In general, an item is an object that is recommended by the system. The items are indicated on the top of the matrix row. In the same way, a user is a person in general with distinct characteristics. They are expecting the recommendation from the system. Formally, the users are represented in the adjacent matrix column. The transaction is an entity representing the relationship between the items and users. The format of the transaction may be numeric, ordinal, binary, or unary value. It may be collected from the user explicitly or implicitly. Based on the type of a recommended item, the design and technique of an RS application are determined. The designed RS produces the recommendations with the help of user data stored in the databases and the knowledge it gets from the data.

The suggestions given by the RS are various from one domain to another domain. For example, a suitable crop and fertilizers for the field are recommended in an agriculture domain, assets are suggested to a large panel of investors in a financial domain, a suitable e-government service is recommended to the citizens in smart cities, music is

Items \ Users	I1	I2	I3	I4
U1	4.5		2.5	3.5
U2	2.5	4	4	
U3	3		4.5	
U4			5	4

FIGURE 1: Components of the recommendation system.

recommended to people in the social media domain, restaurant information is recommended nearby the tourist place in the tourism domain, and many more. The development of the Internet, communication technologies, and network telecommunications is the foundation of the Internet of Things (IoT). Through a wireless network, it can connect related devices and connect electronic and communication equipment that is connected. The controller can send commands to the machinery remotely, carrying out the instructions. In the Internet of Things, physical things are categorized into electronic tags for online connectivity, and the Internet of Things manages the precise position of each object.

2. Review of Literature

According to Kundu et al. [1], because of the ease with which we can now access the world's information, the Internet has become a critical tool in modern business. All walks of life and age groups have been able to benefit from this association factor; education is no exception [2]. As a result of this, the term "e-learning" has been coined. E-learning incorporates many Internet tools, such as Integrated Classroom Teaching, to reach the general public and educate them on a variety of subjects. Teaching quality is improved, delivered, and aided thanks to today's high-tech educational tools. E-learning includes the participation of educators, learners, and mentors in the use of digital technology to enhance their job. Anyone in the modern period may quickly and easily get any information they need by using search engines like Google, Yahoo, and YouTube. With the rise of voice assistants, e-learning is becoming increasingly popular. E-learning, like anything else in the world, has its

advantages and disadvantages. Nevertheless, it has the potential to be a boon to human existence if properly employed. Data innovation has become a common platform for providing e-administration and e-learning administration because of the unstable growth of the World Wide Web (WWW).

According to Juan et al. [3], data sparsity, cold start, and accuracy of the similarity measure are three of the algorithm's primary flaws, and they are all addressed in this study, which provides an overview of current domestic and international research on collaborative filtering. Deep learning for recommender systems is then discussed as a potential research and development trend. The recommendation system's performance could be improved by incorporating vast multisource heterogeneous data into deep learning.

According to Estrela et al. [4], nowadays, students have a wide variety of courses to choose from, and it might be difficult for them to go through the material and choose the right one. An effort to construct a system that recommends online courses based on the user's profile and similarity to other users is the goal of this project. Content-based, collaborative filtering, and hybrid methods were all used to gather data and generate course recommendations for this project. For this reason, the system can provide more accurate recommendations by integrating all three strategies. Consequently, users will not get bored while viewing material that they are interested in and will continue to utilize the system since they are engaged and interested.

According to Khan et al. [5], there has been a massive increase in the amount of data available online in the last few years. Most of this data pertains to online shopping platforms. The bulk of this data makes it tough to analyze and

extract useful information. Getting the outcomes you want in a timely manner is difficult for everyone, regardless of their role in the business. An automated and efficient answer to this problem is provided by recommender systems (RS). In e-commerce, collaborative filtering (CF) is widely recognised as one of the best approaches for recommending products to buyers. This study examines collaborative filtering-based recommendation systems. Item-based and user-based approaches are both used in recommendation systems based on collaborative filtering. When these approaches are used in today's society, they are used in a worldwide internet setting to provide the user with precise results. This paper gives a study of current strategies for recommendation systems and highlights the best techniques for generating accurate results.

According to Sharma et al. [6], RWARS, a new recommender system for recommending research areas, is introduced in this paper. E-commerce, e-services, e-libraries, entertainment, tourism, and social networking sites all have recommender systems in place. However, when it comes to education, there has been little progress made. Because of this, we have created RWARS, a recommender system for educational purposes. During the development of our system, we used the Tanimoto coefficient and cosine similarity. The purpose of this study is to compare the outcomes of several approaches in order to determine which is the most effective. As far as measurements go, the mean square error, root mean square error, and coverage all came into play. RWARS is still in its infancy, but transforming it into an online system will make it even more useful to young researchers.

According to Onah and Sinclair [7], in today's technologically driven educational environment, massive open online courses (MOOCs) are exploding in popularity. MOOCs must evolve from a one-size-fits-all approach. The collaborative filtering method will be used to implement an algorithm-based recommendation system in this framework (CFM). Participant rating options are used to evaluate several items using the collaborative filtering technique (CFM). We are interested in seeing if the recommendation system can help students learn and get more out of their participation in online study sessions. Using Python programming, the goal was to recommend courses to distinct users in a text editor mode. As a result, collaborative filtering will recommend courses to different learners based on comparable rating patterns.

According to Karampiperis et al. [8], it is common practice in social networking platforms to recommend user relationships or interesting shared resources via collaborative filtering. Qualitative information, such as the opinion of the user who actually utilized the resource and whether or not he would recommend it to other users, is not taken into consideration by metrics based on access patterns and user behavior. In educational repositories, where there is a wide range of aims, demands, interests, and expertise levels, this is particularly relevant.

According to Schafer et al. [9], clients' inclinations are considered naturally through recommender frameworks, which are programming devices and philosophies. Ideas

are made to help clients in settling on different choices. IR, text order, AI, and choice emotionally supportive networks all assume a part in the advancement of recommender frameworks (DSS). With the assistance of proposal frameworks, users can avoid becoming overwhelmed with too much information, which is a common symptom of today's Information Overload (IO) crisis. A popular and effective technique in e-commerce has shown to be a worthy instrument for online users to deal with IO. Many well-known e-commerce companies have found them to be quite effective when using their advanced strategies. A summary of the current generation of recommendation methods is presented in this article, as well as CF systems and their algorithms in detail.

According to Ray and Sharma [10], students in management education programmes today have a tough time deciding which electives to take because there are so many to choose from. Course recommendation systems that assist students in choosing course selections have become more relevant as the number and variety of elective courses available to choose from have increased. Collaborative filtering is used in this paper to build a course recommendation system. As a student considers elective courses, grades are an important consideration.

According to Bobadilla et al. [11], there should be more weight given to users with more knowledge in e-learning recommender systems than to users with less knowledge, according to our research. New equations have been created in the core of the memory-based cooperative sifting to accumulate and deal with data in view of the scores got by every client in a variable number of level tests in order to meet this goal.

According to Schafer et al. [12], when it comes to personalization, collaborative filtering is a powerful tool. Cooperative filtering (CF) is a way to filter or evaluate an item based on other people's thoughts and opinions. The filtering of enormous amounts of data is made possible by CF technology, which pulls together the views of many networked groups on the web. CF algorithms and design decisions involving rating systems and rating acquisition are discussed in detail in this chapter, which covers the fundamentals of collaborative filtering for adaptive web users. Rich interaction interfaces and how to evaluate them are also covered in our discussion. We wrap off the chapter with a look at the privacy concerns unique to a CF recommendation service and some of the field's most pressing unanswered topics.

3. Recommendation Systems and Their Types

In general, recommendation system has categorized into six different types in particular content-based, collaborative filtering, demographic, knowledge-based, community-based, and hybrid recommendation system. The types are pictorially represented in Figure 2.

3.1. Content-Based (CB). The input for the CB recommendation system is the user profile or item profile. The user profile is based on user taste. Here, the referred content is an
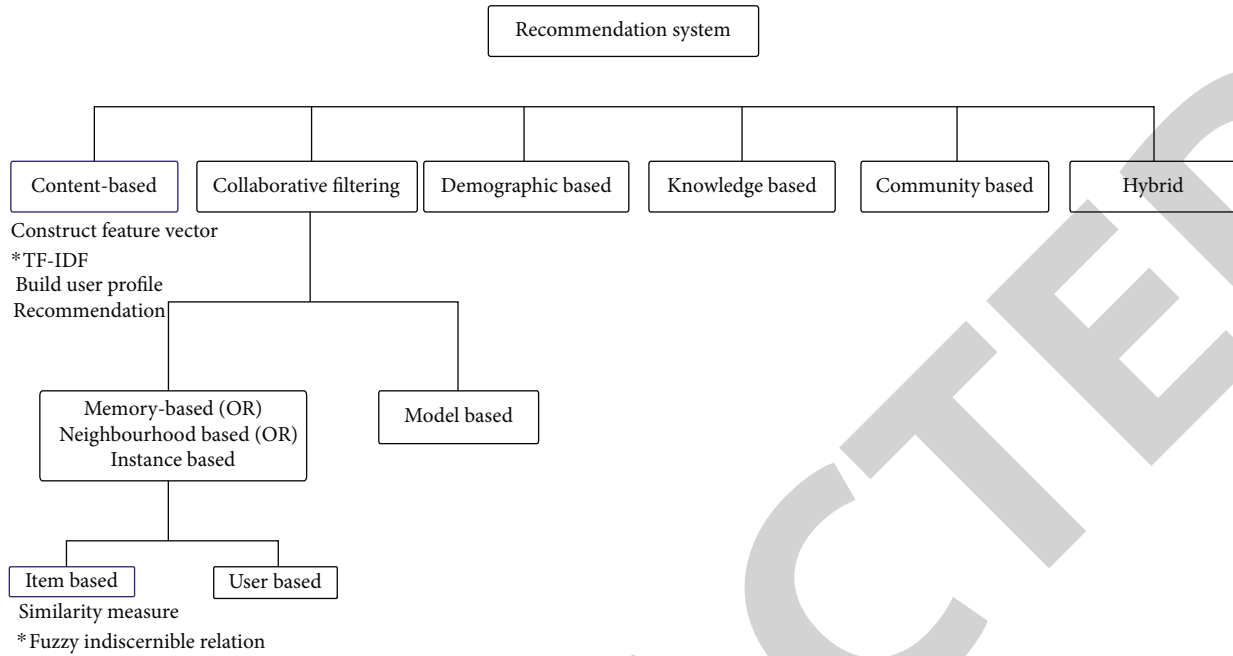
Figure 2: Recommendation system categories.

attribute of the item the user may like. For instance, assuming a client unequivocally gives a positive rating to a book that has a place with the specific writer, then, at that point, the framework can figure out how to suggest different books composed by something similar author to the user.

3.2. Collaborative Filtering (CF). The CF kind of recommendation system works with the rating data or buying behavior data or user-item transaction data. It does not pertain the attributes of the user or item. The similarity between the users' taste is calculated based on the similarity in the past user rating history.

For example, if users U1, U2, and U3 gave a maximum rating to books B1 and B2, then user U4 supposes to buy book B2, they also get recommendations to purchase book B1 because the system identifies book B1 and B2 as similar ratings of users U1, U2, and U3. There are two types of approaches commonly used in the CF-based recommendation system. It is named as memory-based CF and model-based CF.

3.2.1. Memory-Based CF. Memory-based CF determines the correlations between the users or items by loading the entire user and item data into the memory, then predicts a new rating for a user or an item by finding the weighted average of ratings from a similar group. It confides significantly on straightforward similitude estimates like cosine likeness, Pearson relationship, and Jaccard coefficient which are essentially used to match comparative individuals or things.

This type of CF is also named as neighbourhood-based CF or instance-based CF. The correlation between the user to user or item to item is considered in this category. It is further subdivided into two categories like user-based CF (UBCF) and item-based CF (IBCF). In UBCF, the correla-

tion between the users are calculated; then, the ratings of those users are averaged, and the top rank item is recommended. In IBCF, the rated items are taken into account and find the similarities between the items. If comparable things are found, then, at that point, rating for another thing is anticipated by utilizing the weighted normal of the client's evaluations on these comparative things. The approach followed by the UBCF is a lazy learning method. It needs the data to do the prediction. But data is not needed in the case of IBCF. Hence, it is called the eager learning method. IBCF is more efficient than the UBCF method.

3.2.2. Model-Based CF. Model-based CF uses supervised or unsupervised machine learning and data mining methods. First, it creates a model with partial data not for entire users and item data. The technique dimension reduction is used to select the partial data from the entire data. Then, the transaction between the users and items is approximated to make the new prediction [13].

3.3. Demographic-Based Recommendation System. In this type of system, the demographic profile of the user plays a key role to get the recommendation. Due to that, the recommendations are different based on the demographic places. For example, the users are categorized on the basis of their demographic information, and then, the book is recommended based on the demographic classes. The history of user ratings is not needed for this kind of recommendation system.

3.4. Knowledge-Based Recommendation System. This type of RS considered the knowledge about the items, user preferences, and recommendation criteria. The user profile and item rating history details are not sufficient for this

implementation example. The algorithms used in this type of recommendation system know about the way that a specific book meets a specific client need. There are two subdivisions in this class to be specific limitation-based and case-based. The case-based RS decide suggestions as per the closeness measurements. The requirement-based RS transcendently utilizes the predefined information bases that contain unequivocal principles about the relation between the customer requirements with the item features.

3.5. Community-Based Recommendation System. The items are recommended in light of the inclinations of the client companions. This strategy follows the quip "Let me know who your companions are, and I will let you know what your identity is". This sort of RS is coordinated with online media. It obtains the client data about social relations, and the inclinations of the client's companions are considered in this category of RS.

3.6. Hybrid Recommendation System. Each and every recommendation system has its own pros and cons. One type of RS combines with another type of RS to strengthen the recommendation given to the user. CF methods are not able to recommend the new items to the users. But the content-based approaches predict the new items based on their attribute. Lots more ways are available to merge the techniques and create a new hybrid system. So, combining these two recommendation systems for recommending new and existing items is called a hybrid system. The available hybrid recommender systems with the four types of recommendation algorithms: content-based, standard collaborative, heuristic collaborative, and knowledge-based are discussed in a study. Additionally, it discussed the six different hybridization strategies and the 41 hybrid system implementations in detail.

4. Collaborative Filtering Recommendations

The collaborative filtering (CF) system predicts the rating of different items based on the user's previous tastes and the opinions of other like-minded users. The authors proved that CF is very successful in both research and practice. Most of the existing systems use CF as its base algorithm for providing recommendations. The milestones of the CF recommender systems are shown in Table 1.

4.1. Algorithmic Approach of Collaborative Filtering Systems. The first step in CF-based recommendation is to find a group of users who have similarly rated items. This group of users is called a neighbourhood. Once a neighbourhood is formed, the unknown ratings are predicted for the items based on how the user's neighbours have rated that item. After prediction, the top predicted rating items are recommended to the user.

4.2. Types of Collaborative Filtering Algorithms. The authors gathered the CF algorithm into two general classes: memory-based and model-based. Memory-based algorithms basically make rating expectations in light of the whole assortment of recently evaluated things. Model-

TABLE 1: Milestones of some CF systems.

Year	Name of the systems	Items
1992	Tapestry	Users
1994	Netnews	News
1995	Ringo	Music
	Bellcore Video	Movies
	Amazon	Items
1996	Syskill&Webert	Websites
	Entrée	Restaurant
	PHOAKS	Web sites
1997	Fab	Web sites
	Usenet news	News
	Ripper	Movies
1998	Filter bots	News
1999	Advogato	Software
	PTV	TV shows
2000	NewsDude	News
	Netflix	DVD
2001	Jester	Jokes
	ExpertClerk	Items
	NutKing	Travel
2002	Last.fm	Music
	iGoogle	Web pages
	Barnes & Noble	Books
2003	Movie lens	Movies
	INTRIGUE	Travel
	Facebook	Friends
2004	Orkut	Friends
	Flickr	Photos
	Smart Radio	Music
2005	Yahoo! Shopping	Items
	CiteSeer.IST	Papers
	Twitter	Friends
2006	Trip@divce	Travel
	Film Trust	Movies

based calculations utilize the assortment of appraisals to gain proficiency with a model, which is then used to make rating expectations.

4.2.1. Memory-Based Algorithms. The authors discussed the basic memory-based CF algorithm. The authors divided the algorithm into three main phases: neighbourhood formation, pair-wise prediction, and prediction aggregation. Consider an example of the user-item rating matrix with 4 users and 5 items as given in Table 2. The ratings are in the range 1 (poor) to 5 (good), and "?" indicates that the item has not been rated.

The rating prediction of User 1 for "computer network" is as follows: The first step is to find neighbours for User 1. There are several ways to formalize the neighbours, for example, as cosine similarity or Pearson correlation.

TABLE 2: User-item rating matrix.

	Data structures	Compiler design	Computer networks	Computer architecture	Software engineering
User 1	5	4	?	4	4
User 2	1	2	3	1	1
User 3	5	4	2	?	5
User 4	2	?	3	2	1

4.2.2. Model-Based Algorithms. The memory-based algorithm described in the previous section has two main limitations:

Scalability: as the quantity of clients and things develops, the method involved with figuring out neighbours turns out to be very opportunity-consuming. The calculation is roughly direct with the quantity of clients. This is particularly dangerous for enormous, high-volume sites that need to do a ton of personalization.

Sparsity: one more issue with the memory-based CF frameworks is that as the quantity of things develops, clients rate a more modest level of the thing populace. Memory-based calculations necessitate that clients ought to have something like two things to be evaluated to associate them. In a huge informational collection, numerous clients might have no connection by any means. The result of this issue is the absence of neighbours, which prompts helpless suggestions.

Numerous analysts have taken a gander at the issue of diminishing the sparsity of the client thing rating framework utilizing an assortment of strategies including the following:

Filter bots: The researchers proposed content-based software agents to automatically generate ratings. It reduces sparsity and increases the density of the rating matrix, which helps in finding the correlation between users.

Clustering: The researchers group the users into clusters and then use the memory-based algorithm on the clusters for prediction.

The authors proposed item-based technique, which clusters similar items in a rating matrix and estimates the missing values.

Classifiers: The study changed a CF as a classification problem using a machine learning algorithm. Based on a set of ratings from users for items, the authors tried to induce a model for each user that allows classifying unseen items into two or more classes.

Dimensionality reduction technique: The researchers utilized Singular Value Decomposition (SVD), to catch dormant connections among clients and things that permit registering the anticipated likeliness of a specific thing by a client. Likewise, it is utilized to deliver a low-layered portrayal of the first client thing rating network and afterward figure area in the diminished matrix.

The entire model-based CF algorithm has two phases. The first phase is offline, where a model is built which captures the relationships between users and items. The second phase is online, where a model is used to compute recommendations.

5. Evaluation of Collaborative Filtering Systems

Great analyses are fundamental to truly know the advantages and impediments of the proposed suggestion procedures. A study expressed that the presentation assessment of proposal calculations is normally done as far as exactness measure.

Exactness measures can be either factual or choice help. Measurable precision measurements basically analyze the anticipated evaluations against the genuine appraisals in the client thing network and incorporate Mean Absolute Error (MAE). The MAE measure is characterized.

Choice help measures to decide how well a recommender framework can make forecasts of high-pertinence things. They incorporate old style data recovery proportions of accuracy (proportion of significant things chosen by the recommender to the quantity of things) and review (proportion of pertinent things chosen to the quantity of important).

These actions are regularly clashing in nature. For example, expanding the number suggested things N, which will in general build review but diminish accuracy. F1-measure gives equivalent load to both.

6. Technical Issues in Collaborative Filtering Systems

CF recommender systems pose many technical challenges, and some of the technical challenges include the following:

(i) Need for dimensionality reduction techniques or methods to handle a large number of data in recommender system

(ii) Need for sparsity reduction technique, in order to provide high-quality recommendation

(iii) Requirement for performance measures to evaluate the CF recommender algorithms

(iv) Development of recommender systems, to assist the users relevant to education such as books and laptops

7. Basic Terminologies

(a) *Scalability Problem*. The memory-based algorithm requires calculations that develop with both the quantity of clients and things. With a large number of clients and things, an electronic recommender will experience genuine versatility issues. The most

pessimistic scenario intricacy is$O(m * n)$ (m clients and n things)

(b) *Sparsity Problem.* In the recommender framework, the quantity of appraisals previously got is typically tiny contrasted with the quantity of evaluations that should be anticipated. For instance, in online business applications, there are gigantic quantities of things, yet the client has evaluated or bought under 1% of the things (1% of 1 million things is 10,000 things). This prompts inadequate sections in the rating lattice. This is designated "sparsity issue." A study expressed that the outcome of this issue is the absence of neighbours, which prompts helpless suggestions

(c) *Cold Start (New User) Problem.* CF recommender frameworks give a solid suggestion to the client, in light of their rating inclination. Be that as it may, the new clients may not know about the accessible things and their attributes, and they are reluctant to evaluate these things. Thus, these clients jump to rate those things in the framework

This issue is named as "chilly beginning (new client) issue."

Cold start (new item) problem: new things are added consistently to recommender frameworks. CF frameworks depend entirely on the client's evaluations to make proposals. In this way, until another thing is evaluated by an impressive number of clients, the recommender framework would not have the option to suggest. This issue is named as "chilly beginning (new thing) issue."

8. Proposed Model

8.1. The Structure of Study and Other Notions. Each student enrolled has to study some compulsory and some optional subjects, and each subject is divided into different units. Each unit is comprised of many topics.

It proposes that the University has to maintain a server which can store the presentations prepared by the professors of affiliated colleges and University on various topics related to various degrees. These presentations may be in the form of theory (notes), slide shows, audio files, or video files. Naturally, every student joined in a college will be trained and taught by his/her professors. But if a student wants to learn more or wants to understand the topic in a better way, he/she may approach this system and get benefited.

The method of describing the characteristics for users and topics in e-courses is to be discussed. The general and specific characteristic of the user has to be denoted. For example, the use is denoted by the letter U and the general characteristics of the user is denoted by

$$U_i(D_i, S_i, P_i, C_i, \text{SEm}_i), \tag{1}$$

where D_i is the name of the degree that the student i is studying, S_i is the semester that the student i is studying in

D_i, P_i is the place where student i is living, C_i is the college where student i is studying, and SEm_i is the student email id. The specific characteristics of the user can be denoted as

$$U_i(\text{IL}_i, \text{KE}_i, \text{IM}_i), \tag{2}$$

where IL_i is the instruction language for user i and can take the values mother tongue or English or both and KE_i is user i knowledge in English language and can take the values read, understand, converse, and write. Here, the student has to specify a weight for each attribute, and it should be a value in the interval $(0, 1)$. IM_i is instruction methodology preferred by the user and can take the values theory, practical, slide show, and video.

Similarly, the characteristics of topics have to be defined in e-courses. Here, each course is a cluster of many subjects, and each subject is divided into many topics. The general and specific characters of these topics should be denoted. In a subject, the topic is denoted by the letter T, and the general characteristics of the courses are denoted by the letter D:

$$T(\text{tname}, D), \tag{3}$$

where $D = \{u_j, su_k, s_l, y_m\}$. Here, tname is the name of the topic in unit j in the subject k in semester l in the year m.

It is possible that many professors might have prepared the study material for the same course with their ability and hosted it on the web. Hence, it is necessary for us to store the details of the professors who are involved in this e-learning service. The professor is denoted by the letter P, and it can be defined

$$P(\text{Pno}, \text{Name}, \text{Col}, \text{Exp}, \text{Qua}, \text{Spe}, \text{PEm}), \tag{4}$$

where Pno is a number given to the professor, Name is the name of the professor, Col is college where the professor is working, Exp is the academic experience of the professor, Qua is the highest qualification of the professor, Spe is the professor's subject specialization, and PEm is the email id of the professor. It is possible that the same professor might have prepared e-learning material for many courses. The e-learning material prepared by professors is combined with the topic as follows:

$$C_i(\text{Pno}, \text{tname}, D, \text{IL}_i, \text{IM}_i, \text{PL}, R), \tag{5}$$

where PL is the presentation level specified by the professor (very easy to understand with moderate technical level, understandable with good technical level, tough, and highly technical) and R is the rating given by the user. Here, students will rank whether they found it easy or difficult to understand, and the votes are converted into percentages. $R = \{\text{easy, moderate, difficult, very difficult to understand}\}$.

When a student wants to access the system, he has to supply the following details to the system so that system will recommend him/her the best possible study material:

$$\text{TR}(\text{Uno}, \text{tname}, D, \text{IL}_i, \text{IM}_i, \text{SK}). \tag{6}$$

TABLE 3: Student general information table.

Sl no.	Field name	Type	Width	Description
1	User no.	Text	8	User number
2	Username	Text	40	User name
3	Degree	Text	40	Course of study
4	Semester	Number	2	Semester number
5	Place	Text	40	Place where user is using the system
6	College	Text	40	Name of the college where the student is studying
7	SEmail	Text	40	Student email
8	Phyedu	Number	1	Knowledge of physical education subject on the scale of 10

TABLE 4: Topic table.

Sl no.	Field name	Type	Width	Description
1	Topic no.	Text	10	Topic code
2	Topic name	Text	40	Name of the topic
3	Degree	Text	30	Degree to which this topic is included
4	Semester	Number	2	Semester in which this topic is included
5	Subject	Text	40	Subject name in which this topic is included
6	Unit	Number	2	Unit number in which this topic is included

TABLE 5: Professor's information table.

Sl no.	Field name	Type	Width	Description
1	Profidno	Text	10	Professor identification number
2	Prof name	Text	40	Professor name
3	College	Text	40	College where professor is associated
4	Experience	Number	2	Professor's experience
5	Qualification	Text	40	Qualification of professor
6	Specialization	Text	40	Professor's specialization
7	PEmail	Text	40	Professor's email

Here, Uno is the user number allotted by the system after filling the form available in Figure 2 and SK is the level of subject knowledge that the student is already having so that he can understand the topic further. Values SK can take low, moderate, and high.

8.2. Forms. Students who want to learn through this e-learning system have to enroll with this system. The information can be obtained about the student who wants to use this e-learning service with the form shown.

Student Enrollment Form

(1) Name of the student

(2) Degree

(3) Year

(4) Semester

(5) College

(6) Student email id

(7) Place where you live

(8) Instructional language you prefer

(9) Please rate your knowledge level in physical education course

    (a) Good:

    (b) Bad:

    (c) Average:

    (d) Excellent:

Note: please press the submit button after filling in the above details. You will be provided with a user number. You can use that user no. in future interactions.

Your user number:

Submit.

8.3. Method. In the first step, the student has to enroll himself/herself using the form given in Figure 2. After getting the user number, he can use the system for his e-learning purposes.

In the second step, the student has to use the form provided in Figure 2 to proceed to the e-learning system. The system has to compare the attributes specified by the students with the attributes assigned to the various topics and will recommend the list of best-suited topics to the student.

In the third step, the student has to learn with the material supplied by the system.

In the fourth step, the student has to give feedback to the system with the alternatives given (easy, moderate, difficult, and unable to understand). The system will convert to percentages.

8.4. Files and Algorithm. Stdgeninfo file will store information about the student (user), and its structure is as shown in Table 3.

Topicfile will store information about the topics available for e-learning and its structure is as shown in Table 4.

Proffile will store the information about the professors who are contributing to the e-learning system. The file structure is as shown in Table 5.

Proftopic file stores the information about professor number topic number and other attributes of the topic which are useful during recommendation.

9. Conclusion

The educational domain is based on a heterogeneous collection of information. The design of the recommender system approach depends on the information sources and interest in various objects used by the system. Some of the sources are easily available, and some of them are not easily available. All basic recommending approaches are applicable to the education domain, and heterogeneity of this domain naturally favors the use of hybrid recommenders. When a student enrolls in a degree, he has to study some of the subjects in each semester. Every educational institute will train the students in these subjects and make the students well for the examinations. When the student wants to learn more or needs much more information, he has to approach the e-learning system that is one of the ways to satisfy his educational needs. Researchers proposed a model for the physical education course based on the undergraduate. Design of data files and file structures is analyzed and developed. Algorithms for enrolling the students and methodology of recommending the topics are also developed. While designing, research scholars considered the various aspects which will help in the recommendation process. The above-said techniques may be fine-tuned in future work.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] S. S. Kundu, D. Sarkar, P. Jana, and D. K. Kole, "Personalization in education using recommendation system: an overview," in *Computational Intelligence in Digital Pedagogy*, A. Deyasi, S. Mukherjee, A. Mukherjee, A. K. Bhattacharjee, and A. Mondal, Eds., vol. 197 of Intelligent Systems Reference Library, Springer, Singapore, 2021.

[2] R. Wang, M. B. Alazzam, F. Alassery, A. Almulihi, and M. White, "Innovative research of trajectory prediction algorithm based on deep learning in car network collision detection and early warning system," *Mobile information systems*, vol. 2021, Article ID 3773688, 8 pages, 2021.

[3] W. Juan, L. Yue-Cin, and W. Chun-Ying, "Survey of recommendation based on collaborative filtering," *Journal of Physics: Conference Series*, vol. 1314, article 012078, 2019.

[4] D. Estrela, S. Batista, D. Martinho, and G. Marreiros, "A recommendation system for online courses," in *Recent Advances in Information Systems and Technologies. WorldCIST 2017*, Á. Rocha, A. Correia, H. Adeli, L. Reis, and S. Costanzo, Eds., vol. 569 of Advances in Intelligent Systems and Computing, pp. 195–204, Springer, Cham, 2017.

[5] B. M. Khan, A. Mansha, F. H. Khan, and S. Bashir, "Collaborative filtering based online recommendation systems: a survey," in *2017 International Conference on Information and Communication Technologies (ICICT)*, Karachi, Pakistan, 2017.

[6] R. Sharma, S. Vinayak, and R. Singh, "Research work area recommendation based on collaborative filtering," *International Journal of Education and Management Engineering*, vol. 7, pp. 50–62, 2017.

[7] D. Onah and J. Sinclair, *Collaborative Filtering Recommendation System: A Framework in Massive Open Online Courses*, 2015.

[8] P. Karampiperis, A. Koukourikos, and G. Stoitsis, "Collaborative filtering recommendation of educational content in social environments utilizing sentiment analysis techniques," in *Recommender Systems for Technology Enhanced Learning*, N. Manouselis, H. Drachsler, K. Verbert, and O. Santos, Eds., pp. 3–23, Springer, New York, 2014.

[9] J. B. Schafer, D. Frankowski, J. Herlocker, and S. Sen, "Collaborative filtering recommender systems," in *The Adaptive Web. Lecture Notes in Computer Science*, P. Brusilovsky, A. Kobsa, and W. Nejdl, Eds., vol. 4321, Springer, Berlin, Heidelberg, 2013.

[10] S. Ray and A. Sharma, "A collaborative filtering based approach for recommending elective courses," *Communications in Computer and Information Science*, vol. 141, pp. 330–339, 2011.

[11] J. Bobadilla, F. Serradilla, and A. Hernando, "Collaborative filtering adapted to recommender systems of E-learning," *Knowledge-Based Systems*, vol. 22, pp. 261–265, 2009.

[12] J. B. Schafer, D. Frankowski, J. Herlocker, and S. Sen, "Collaborative filtering recommender systems," in *The Adaptive Web. Lecture Notes in Computer Science*, P. Brusilovsky, A. Kobsa, and W. Nejdl, Eds., vol. 4321, Springer, Berlin, Heidelberg, 2007.

[13] M. Bader Alazzam, F. Alassery, and A. Almulihi, "Identification of diabetic retinopathy through machine learning," *Mobile Information Systems*, vol. 2021, Article ID 1155116, 8 pages, 2021.