International Journal on Advanced Science Engineering Information Technology

Vol.14 (2024) No. 1
ISSN: 2088-5334

An e-Learning Recommendation System Framework

Wan-Er Kong [a], Su-Cheng Haw [a,*], Naveen Palanichamy [a], Siti Husna Abdul Rahman [a]

[a] *Faculty of Computing and Informatics, Multimedia University, 63100, Cyberjaya, Malaysia*
*Corresponding author: *sucheng@mmu.edu.my*

***Abstract*—With the emergence of the digital era, the e-learning platform has become an effective tool for obtaining quality e-learning content. However, despite its potential, the true extent of its capabilities has yet to be fully explored. In order to attract users and maximize revenue, e-learning platforms are now expected to provide content tailored to their users' needs and preferences. These recommendations are generated by considering factors such as prior purchases, browsing history, demographic information, and more. By leveraging these advanced technologies, e-learning platforms can enhance the learning experience by providing users with content that is both engaging and relevant to their individual needs and interests. This paper explores the popular Machine Learning (ML) techniques employed in e-learning content recommender platforms. Two machine learning techniques, k-Nearest Neighbour Baseline (KNNBaseline) and Singular Value Decomposition (SVD), are selected and used to accurately forecast customer interests and preferences. By examining the data patterns and user behaviors, these ML techniques provide insights into the most relevant and personalized educational content for individual users, enhancing their learning experience. The item ratings predicted are generated based on the underlying pattern in past ratings of users. The performance of applied approaches was assessed using several evaluation metrics, which include root mean square error and mean absolute error.**

***Keywords*— Machine learning; e-learning system; recommendation system; recommendation technique; collaborative filtering**

Manuscript received 21 Jan. 2023; revised 29 Aug. 2023; accepted 14 Oct. 2023. Date of publication 29 Feb. 2024.
IJASEIT is licensed under a Creative Commons Attribution-Share Alike 4.0 International License.

I. INTRODUCTION

Recommendation systems have become increasingly prevalent in our daily lives with the advent of websites like YouTube, Amazon, and Netflix [1],[2]. A recommendation system is a program that utilizes Big Data technology to suggest suitable content to users. Businesses widely use recommender systems because they can forecast customer interests and wishes on a highly customized level [3]. They can direct customers to almost any item or service that piques their interest.

As digitalization steps into this era, humans create an incredible volume of data in daily life using various platforms. It is known that we can extract valuable information from this enormous size of data by leveraging the knowledge of data science, such as identifying a restaurant preference for a specific group of people, generating business strategic plans as well as calculating the next potential trending products of a company [4], [5]. Without this technology, the information overflowing at users' fingertips could be rather messy and less attractive to them, which contributes to a less satisfying experience. This would happen in any field; education is no exception. In the area of e-learning [6], recommender systems are essential for combating the issue of information overload.

Recommender systems are a subclass of information filtering systems that offer personalized suggestions according to user interest by utilizing the algorithm behind the system. The algorithm calculate and show the most preferable product or service from the overwhelming selection based on the information obtained from the user's collected data. In conjunction with the emergence of machine learning, humans manage to generate a variety of recommendation systems that assist users in making decisions and provide an excellent experience for users.

We can see recommendation systems actively applied by big companies such as Google and Meta in their products. For example, YouTube, a well-known content-sharing platform, adopts a recommendation system to help users search for videos with a high possibility of interest. As a global online shopping platform, Amazon also applies the recommendation system to suggest a series of products that users opt to purchase. The random advertisement appearing during web surfing is also one of the most common recommendation systems. A recommendation system is highly applied to increase user and platform engagement, further benefiting the company.

The produced recommendation can be generated by multiple factors, including prior purchases, browsing history, demographic information, and others. These recommendations are developed based on the data collected from customers. If the recommendation algorithm is well-written, this strategy might generate tremendous income and make a company stand out from its competitors [7].

Machine learning (ML) is an emerging approach that makes it feasible for computers to develop knowledge autonomously using historical data. Machine learning utilizes various techniques to build mathematical models and generate forecasting upon prior experience or data [8], [9]. This technology can be applied in many domains, including sales predicting, weather forecasting, and user interest predicting. For example, Khademizadeh et al. [10] use the decision tree technique and item-based collaborative filtering algorithm to build a recommender system that analyzes book circulation transactions and identifies user book lending trends. Jiang et al. [11] also apply a long short-term memory model and the composite sentiment index to help scholars analyze the energy sector more effectively.

This paper discusses an in-depth knowledge of ML, which includes the purpose of ML and the characteristics of different ML approaches as in the advantages, disadvantages, and suitability for different ML during application in an e-learning course recommender system. This knowledge can be obtained by studying a variety of research papers and combining their findings.

II. Materials and Method

A. *Stages of the Recommendation System*

Having sufficient data for recommendation systems to learn and explore a user's behavior and traits is crucial. According to [12], the volume of data is one of the crucial 5Vs essential in data science. The data volume has the potential to create a significant impact on the recommendation to users. Poor quality data only leads to faulty output, which is an unfavorable suggestion in this circumstance, regardless of how great and competent the recommendation system is. The recommendation system can be split into three primary phases: information collection, learning, and prediction. The prediction is also known as the recommendation phase. Figure 1 presents the three phases of the recommendation system.

At this level, the recommendation system gathers user data and information to build a forecast task model incorporating user characteristics, behaviors, and resource content. The system needs to gather a large amount of information about the user to provide accurate recommendations. Recommendation systems depend on many forms of inputs, like response that includes user interests in goods as input and indirect feedback that indirectly concludes consumer preferences by monitoring user behavior. Hybrid feedback may be obtained by combining indirect and direct feedback. A recommendation system cannot deliver proper and accurate results unless the user model is well-built.

Implicit feedback is a common way to collect information about users. The system automates the user's preferences by analyzing the user's history, purchases, links visited, time spent on various web pages, email content, and other data. This type of feedback does not involve user action; instead, it automatically delivers recommendations by examining the elements described above. In the explicit feedback, however, users are prompted by the system to provide feedback to generate recommendations. The user ratings are crucial to determine the effectiveness and quality of the recommendation system. Despite implicit feedback, this approach requires the user to put in more effort during feedback.

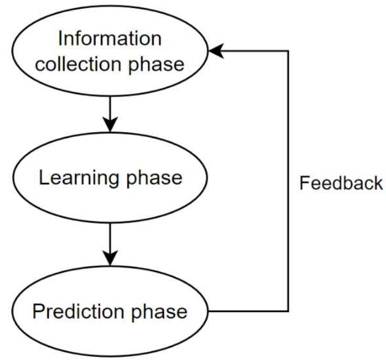

Fig. 1 Example of an unacceptable low-resolution image

In order to produce hybrid feedback, the drawbacks of implicit and explicit responses are eliminated while both of their benefits are merged, making hybrid feedback the best compared to implicit and explicit feedback. This type of feedback may be obtained by allowing users to submit direct feedback and ratings while using indirect data as a recommendation element.

In order to recognize and utilize the user's data, the response collected during the information collection stage is filtered by utilizing a learning algorithm in this step. The techniques that are useful in identifying patterns suitable for application in particular circumstances are learning algorithms.

During the last stage of the recommendation system, the system offers recommendations according to the provided data by utilizing the underlying patterns identified throughout the learning phase. The dataset obtained during the information-collecting step can be either memory-based, model-based, or system observations of the user's behaviors and can be used to make recommendations and predictions directly.

B. *Recommendation System Techniques*

For a recommendation system to offer accurate recommendations to each user, it is essential to deploy efficient and accurate recommendation techniques. This demonstrates how crucial it is to comprehend the characteristics and suitability of various recommendation techniques. Figure 2 presents the common recommendation approaches used in the industry.

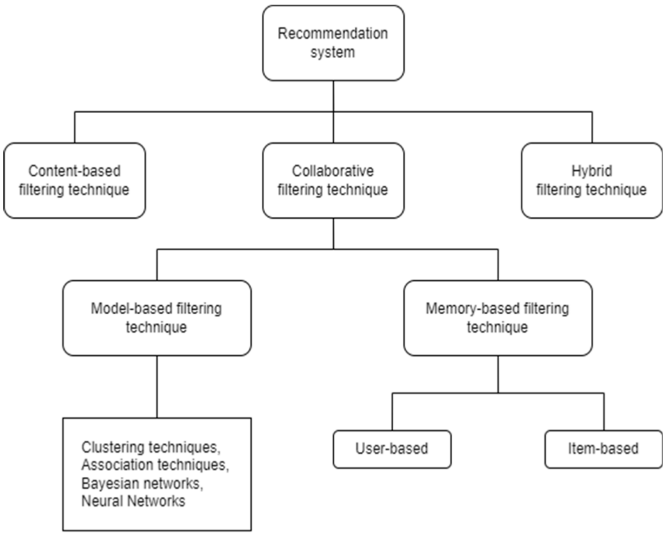

Fig. 2 Common Recommendation Techniques.

1) Content-based Filtering Technique: The content-based technique is an algorithm that takes account of the analysis of item attributes to produce accurate predictions. The content-based filtering strategy is widely used for recommending documents like websites, journals, and media. The content-based filtering technique generates recommendations based on user profiles and features retrieved from the evaluation history content [11], [13]. The user is given recommendations for things that show a strong positive relationship to the highly rated content. To achieve this technique, two methods are performed: vector spacing and classification model.

Vector spacing method: The vector spacing method was the first technique applied to the content-based recommendation system for generating item suggestions to users [14], [15], [16]. The suggestion ranking is customized based on the information provided from the user end. Under this technique, a user vector and an item vector are first created. Every item in the item vector is given a value using multiplication and obtaining the item's and user vector's dot product. The produced result could contribute to the recommendation for users. For example, a user is interested in horror movies and often gives high ratings for horror movies. According to the provided information from the user, the next recommended movie would probably be a horror movie. The dot products of all the available movies could be arranged in a ranking manner, and only the top ones could be included in the recommended movie list to the user.

Classification method: This classification method is used to decide whether an item should be recommended by creating a decision tree [17], [18]. Related statements or conditions compose a decision tree; the last node could indicate the final decision. For example, an algorithm is deciding whether it is suitable to recommend a specific movie to a user. The type of movie is the first condition checked by the system, and that movie is a horror movie, which suits the user's interest. Next, the rating of that movie is checked, and that particular movie has received a rather low recommendation from other consumers. These classifications result in the movie not being recommended to the user.

2) Collaborative Filtering Technique: Collaborative filtering is widely used and well-liked as the recommendation technique. For information that cannot be accurately represented using metadata, particularly videos or audio, collaborative filtering is the recommended approach that is most suitable in this situation [19], [20]. Building a database of user preferences for items is how the collaborative filtering technique always performs. The system could recommend users with similar interests and preferences by comparing users' profiles. Recommendations generated by this technique will be either predictions or recommendations. The collaborative filtering technique can be split into two sub-branches, which are memory-based and model-based.

Memory-based method: The memory-based method is one of the simplest techniques because no model is applied in this method. The predictions created using this method can be formed based on the information extracted from past data using a simple distance-measurement approach: nearest neighbor. This method will identify the user group with similar opinions to the user based on their past data and conclude that there is a high possibility that the user shows interest in the high-rating items given by this user group. This memory-based system can be categorized into two groups: user-based collaborative filtering and item-based collaborative filtering. User-based collaborative filtering focuses on identifying identical users based on given ratings on the same items.

In contrast, item-based collaborative filtering is responsible for searching for items similar to items the user has given a high rating and recommending them to the user. For example, user-based collaborative filtering is employed in the recommendation system if User A and User B share the same interest and use the same shopping platform. User A purchases an item that User B has never rated on the shopping platform and gives a high rating. Since they share the same interest, that item would have a high possibility of being recommended to User B.

Model-based method: The model-based approach has a pre-requisite compared with other techniques. This approach will assume that an underlying model exists and that the prediction as the outcome would match the model perfectly. This method could undergo matrix factorization to decrease redundant user-item matrices and improve the algorithm's performance. In addition, memory usage and computation time could also be reduced as this process can free the empty or sparse user-item matrix. Matrix factorization is applicable in several fields, including image recognition and recommendation. Due to the likelihood that one user may only rate a limited number of movies, matrices utilized in this type of problem are generally sparse.

3) Hybrid Filtering Technique: Real-world recommender systems usually combine the best features among several recommendation techniques into a hybrid technique to increase performance and overcome limitations in each technique. Based on the hybrid techniques, the advantages of one algorithm might be outweighed by the disadvantages of another algorithm, resulting in better suggestions than those provided by a single algorithm [21]. Multiple recommendation approaches can be employed to conceal the flaws of each individual recommendation strategy in a combined model. Integrating the techniques may be carried out in a variety of ways, including individually applying the techniques and combining the results [22]. There are seven basic hybridization techniques of different recommendation method combinations: weighted, switching, mixed, feature combination, feature augmentation, cascade, and meta-level.

- Weighted hybrid system: The weighted hybrid system enables the combination of multiple models applied to the recommendation process simultaneously. The results produced from different models could be combined in a static weight manner where the weight could remain the same throughout the train and test set.
- Switching hybrid: The switching hybrid is special as it can switch recommendation systems. An extra layer built on the recommendation model makes the switching process available. This layer can help the system choose the most suitable model depending on the condition and apply that particular model in the system.
- Mixed hybrid strategy: This hybrid strategy creates many candidate datasets by first using user profiles and characteristics. The recommendation system could feed these candidates to the recommendation model and generate a series of recommendations as output. In this method, a partial dataset is fitted to the right model in order to optimize the performance.
- Hybrid feature: In the hybrid feature combination approach, a virtual contributing recommendation model that serves as feature engineering for the original profile dataset is introduced to the system. This hybrid model makes integrating certain features from other recommendation models into the existing model possible.
- Feature augmentation: This is a contributing recommendation model that aims to grade or categorize the profile of the user or item. This rating or categorization is then used in the recommendation system to deliver the forecast result. The feature augmentation hybrid can enhance the main system performance without altering the primary recommendation model.
- Cascade Hybrid: According to the Cascade Hybrid notion of a strict hierarchical structure recommendation system, the primary model generates the principal result. In contrast, the secondary model deals with a few minor issues with the original outcome. Since the majority dataset appears to be sparse, the secondary recommendation project is beneficial in addressing problems with incomplete data or equivalent scores.
- Meta-level hybrid: In a meta-level hybrid, similar to the feature augmentation hybrid, the contributing model offers an improved dataset to the primary recommendation model. Meta-level substitutes the initial dataset using a fully trained model from the contributing model and is further used as input to match the primary recommendation model, in contrast to the feature augmentation hybrid.

The previous sections have discussed the most common varieties of recommender system strategies used in industries. However, it is known that each approach has different aspects, and it is not easy to select the most appropriate technique for the task at hand. Table 1 highlights the advantages and disadvantages of each recommender system approach to visualize the difference.

TABLE I
ADVANTAGES AND DISADVANTAGES OF RECOMMENDER SYSTEM TECHNIQUES

Techniques	Advantage	Disadvantage
Content-based Filtering	• As the provided recommendations are made for an individual, the system does not need data from others, which results in new users being able to receive their recommendations and overcome the cold-start issue [23]. This could make scaling for massive numbers of users even simpler. • The system can detect a user's interest and further generate recommended products that maybe other users are not interested in [24]	• As feature representation of the items is hand-engineered, the system needs domain knowledge. Hence, the system's performance is consequently restricted to its hand-engineered components. • The algorithm can only offer recommendations depending on the user's current interests. This is due to the system being only partially able to capitalize on the interests that the customers have previously formed.
Collaborative Filtering	• Since the embeddings are automatically learned, the system does not require domain expertise [23]. • The system is able to help users to discover new interests. The system can recommend similar items to users even without knowing the user's interest in that particular item. • To a certain degree, the system may train a matrix factorization model exclusively utilizing the feedback matrix. The system may be used as a candidate generator as no specific contextual qualities are needed.	• The system lacks sufficient data to suggest items to new users who have not yet given a rating, leading to cold-start issues [24]. • Due to the system's data sparsity issue, it is difficult to implement side features for items.
Hybrid Filtering	• The system combines the strengths of multiple filtering techniques, which overcome most of the	• The system is too complex to be applied to some datasets.

Techniques	Advantage	Disadvantage
	drawbacks of other approaches [24] • Provide the most accurate recommendations than the previous approaches.	• The system is too expensive for implementation.

By observing Table I, it is clear that each type of recommender technique is unique and possesses its own advantages and disadvantages. The content-based filtering system has no issue with the cold-start problem as this technique depends on user ratings while the performance of collaborative filtering depends on the collected data from users to suggest high-rating items to users. The hybrid filtering system adopts more than one technique; hence, the recommendation can be more accurate and suitable for most users. However, this approach requires far more computational power than the other techniques and might be expensive regarding hardware implementation.

It is concluded that the most powerful technique does not necessarily mean that approach suits the most. Acknowledging the budget limit, like time and cost, the most important part is that the invested budget could always meet the expected result. Although a strong approach could sound promising, if the application goes beyond the scope of your resources, it might not be the best choice. In order to implement the plan effectively within the allocated budgetary constraints, it is crucial to establish a balance between the required degree of effectiveness and the practicalities of doing so. The decision-making can be further improved by ensuring the selected approach complies with the practical requirements and limits by being aware of these constraints and matching the money spent with the anticipated results.

C. Related Works

Tarus et al. [25] proposed a recommendation technique that integrates the ontology and collaborative filtering technique for a customized recommendation learning resources to online learners. As collaborative filtering produces item recommendation and forecasts user rating, ontology is employed for incorporating learner traits in the recommendation process together with the ratings. The ontological knowledge is also applied when the cold-start issue is first tackled due to insufficient user ratings. To assess the accuracy of the proposed technique, two tests were applied on the same dataset. A collaborative filtering and ontology (Ontology-CF) combination was implemented in the first trial. The second experiment made use of CF exclusively. Based on the evaluation findings, the proposed recommendation method is proven to deliver a higher performance, including customization and recommendation effectiveness, than the collaborative filtering technique. The disadvantages of this approach could include an unavoidable cold start during starting stages of recommendation as the data provided to the system is insufficient.

In another separate research, Tarus et al. [26] introduced a hybrid recommendation technique that integrates context awareness, sequential pattern mining (SPM) and CF algorithms to deliver suitable learning materials to the learners. The proposed hybrid recommendation algorithm utilizes the use of the Generalized Sequential Pattern (GSP) algorithm to mine web logs and identify the sequential access patterns of learners, context awareness to incorporate contextual information about learners, and collaborative filtering to generate recommendations based on contextual data. The accuracy of this technique is assessed by computing and comparing the recall, precision, and F1 measure values. According to experimental findings, the proposed recommendation technique shows greater accuracy and suggestion quality. Furthermore, the suggested hybrid technique can assist in resolving issues with data sparsity by using contextual data and learner sequential access patterns to provide forecasts without overlapping learner ratings.

Wan and Niu [27] proposed a hybrid recommendation technique to provide customized and varied e-learning suggestions. First, the researchers put forth an influence-based learner model that is unrelated to rating data. This impact model is accessible to close data deficiency gaps in the CF recommendation's underlying data. Intuitionistic fuzzy logic (IFL) is used to improve the learner model while considering uncertainty and vagueness, which contributes to the presentation of a more accurate and adaptable learner influence model. The researchers employ self-organization theory to mimic the cooperative behaviors of learners to cluster the ideal learner clique for an active learner.

In contrast to previous recommendation techniques that are entirely learner-active, the clusters are produced by the learners' dynamic interaction with one another due to information propagation. Entropy, evolution time, distance between and within classes, and assessments of the learning process, comprising learning time and scores, are some of the metrics used to evaluate the effectiveness of various parameters. The suggested hybrid strategy is demonstrated to be successful, flexible, available to individualized and various realizations based on the experimental findings.

More recently, Ezaldeen et al. [28] suggested a brand-new framework Enhanced e-Learning Hybrid Recommender System (ELHRS) to match learner needs with appropriate e-content. They developed a novel model to generate the Semantic Learner Profile, which interacts between learning rules and patterns. The system uses sentiment analysis models with five discrete classes of fine-grained sentiment categorization to forecast e-learning material evaluations from text reviews. They used a customized dataset and open dataset, combining Natural Language Processing (NLP) algorithms with a Convolutional Neural Network (CNN). Two augmented language models were demonstrated based on Skip-Gram and Continuous Bag of Words approaches. A strong language model combining these techniques resulted in improved vocabulary representation, resulting in an accuracy improvement of 89.1% for the CNN-Three-Channel-Concatenation model. Performance assessment of CNN-based models included accuracy, precision, recall, and F1 measure value. The proposed recommendation method considers learner preferences, background experiences, and reviews of top learning resources. However, fine-grained sentiment analysis is challenging, leading most research to focus on two-polarity categorization.

Shahbazi and Byun [29] emphasized that to offer the appropriate material and search recommendation according to the users' search history, a virtual and intelligent agent-based recommendation system is proposed. In order to improve the recommendation process and benefit user choice, this paper demonstrated the coupling of NLP techniques with semantic analysis and data mining technologies. Machine learning performance analysis is adopted for further enhancement of the user ratings predictions. The introduced approach and the simulation results indicate the smallest metric error in contrast to the existing work. The successes of the suggested strategy include giving the user a pleasant environment for educational studies and suggestions. Similar to this, suggestion of the same materials and programs is refrained. The algorithm examines user preferences and enhances the accuracy of the recommendation system to produce crucial content according to the user profile scenario.

Alatrash et al. [30] proposed a novel method of e-learning hybrid Recommendation System Based on Sentiment Analysis (RSBSA) is proposed to make suitable content suggestions according to the learner preferences. This methodology applies the technology of NLP and Convolutional Neural Network (CNN) techniques at the same time. To categorize text evaluations of virtual content provided on an e-learning platform, the combination of two methodologies is carried out using fine-grained sentiment analysis models. Their paper presents two enhanced language models that rely on S-G and CBOW. Three rigorous language models are built to estimate resource ratings from online student reviews using a variety of Convolutional Neural Network models. The suggested models are assessed and tested using ABHR-1 and two open-source datasets. The simulation results are analyzed with the empirical findings, and the Multiplication-Several-Channels-CNN model has successfully outperformed the remaining models by achieving an accuracy of 90.37% from fine-grained sentiment classification on 5 different classes.

Therefore, the model-based CF performs better than other recommended system techniques. This could be because the model-based CF technique takes the perspectives of both users or objects into consideration, unlike other techniques. However, the previous studies also illustrate that recommendation based on memory-based CF and item-based CF generally works well. In addition, the majority of recommended systems are assessed using the Mean Absolute Error and Root Mean Squared Error metrics. This demonstrates that these measures are widely used to assess the performance of a recommended system.

D. Proposed Framework for e-Learning

Though there are efforts to build frameworks that can be adopted in various domains, it is not yet matured, and a long way to go before widespread use [31]. This paper proposes a framework for recommender systems in the e-Learning domain. Yet, many issues still persist, especially in technical, personalization, and user acceptance. The flowchart of prototype implementation is presented in Figure 3. At the implementation stage, the developed prototype is designed to offer several selections of machine learning techniques to perform further operations. Next, the prototype could carry out background operations like data cleansing, model training and evaluation. The accuracy results produced by applying different machine learning approaches in the prototype are presented and compared at the next step. A graphical user interface is created to smoothen the simulation process and a better visualization.

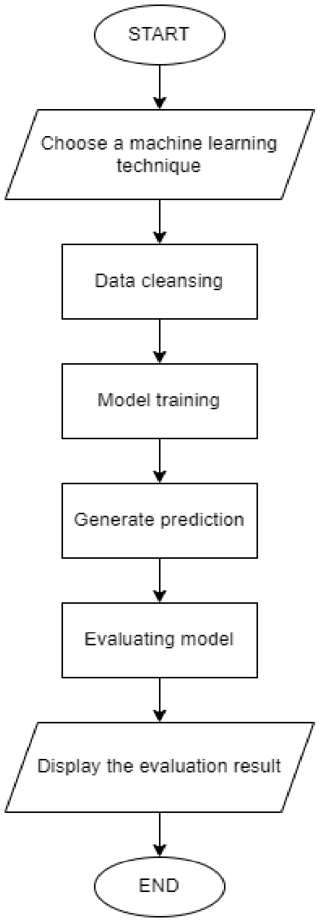

Fig. 3 Prototype Flowchart

1) Choosing the ML Techniques: This research project could involve two ML techniques, which are k-Nearest Neighbour Baseline (KNNBaseline) and Singular Value Decomposition (SVD). The first ML approach is KNNBaseline. This straightforward collaborative filtering (CF) algorithm considers the baseline and applies the target function's local minimum to discover an unknown function while maintaining appropriate precision and accuracy. The second ML approach is SVD. This method is also a CF approach which is capable of performing dimensionality reduction or breaking down data into the critical parts required for analysis, comprehension, and characterization of data.

These two algorithms are chosen due to their application popularity in many research papers in related fields. SVD can be used as a model-based CF to learn the data's characteristics. This approach does not need large storage space to keep the entire dataset, and this algorithm can measure similarity on the resultant matrix, which can be significantly more scalable when working with huge sparse datasets. KNNBaseline can identify the underlying pattern in non-linear decision boundaries, which is essential for classification and regression tasks. Moreover, this algorithm is extremely flexible as it can modify the K value to determine the decision boundary range.

2) Dataset: The dataset chosen in this research project for application in prototype is E-learning Recommender System Dataset [32]. This public dataset combines two tables: item and explicit rating. Table II and Table III explain each column in these two tables.

TABLE II
COLUMN DESCRIPTION OF ITEM TABLE

No	Column	Description
1	item_id	Course ID
2	language	Language used in the course
3	name	Course name
4	nb_views	Number of views
5	description	Course description
6	created_at	The year of course created
7	Difficulty	Difficulty level of course
8	Job	Job that suits this course
9	Software	Software that involves the knowledge of course
10	Theme	Theme of course
11	duration	Course recording duration
12	type	Lesson type

TABLE III
COLUMN DESCRIPTION OF EXPLICIT RATING TABLE

No	Column	Description
1	user_id	Identification of user
2	item_id	Identification of item/course
3	watch_percentage	Watched percentage of this course
4	created_at	The timestamp of course rated
5	rating	Rating given by user (range from 1 to 10)

3) Data Cleansing: The data cleansing starts with stripping unwanted punctuation marks and spaces at a column named 'Theme' as shown in Figure 4.

```
In [3]: # Separate 'Theme' column
        df_items["Theme"] = df_items["Theme"].str.strip("[]").str.replace("'","").str.replace(" ","")
        df_items["Theme"]

Out[3]: 0                     Discover
        1        Share,Produce,Organize
        2             Share,Collaborate
        3              Produce,Organize
        4              Produce,Organize
                         ...
        1162          Produce,Customize
        1163                    Produce
        1164                    Produce
        1165                   Organize
        1166                    Produce
        Name: Theme, Length: 1167, dtype: object
```

Fig. 4 Strip unwanted punctuation marks and spaces

The course theme is separated into new columns (see Figure 5) using the function get_dummies() and then joined with the main dataset.

```
In [3]: # Separate 'Theme' column
        df_items["Theme"] = df_items["Theme"].str.strip("[]").str.replace("'","").str.replace(" ","")
        df_items["Theme"]

Out[3]: 0                     Discover
        1        Share,Produce,Organize
        2             Share,Collaborate
        3              Produce,Organize
        4              Produce,Organize
                         ...
        1162          Produce,Customize
        1163                    Produce
        1164                    Produce
        1165                   Organize
        1166                    Produce
        Name: Theme, Length: 1167, dtype: object
```

Fig. 5 Separate course theme

Next, a few unwanted columns are dropped from the dataset as the upcoming operation does not require these columns (see Figure 6).

```
In [8]: # drop unwanted columns
        df_items.drop(columns=['language', 'nb_views','description','Job','Software','duration','type','Difficulty'],inplace=True)
        df_ratings.drop(columns=['created_at'],inplace=True)
```

Fig. 6 Remove unwanted attributes

Thirdly, both tables are merged on the column 'item_id' as shown in Figure 7. The duplicated data is removed by only keeping the last rating record in cases where the user rates the particular item more than once (see Figure 8). Last but not least, data filtration is applied according to two criteria, as depicted in Figure 9. The first criterion entails that only users who rate at least four books must be preserved, while the second criterion is to maintain the least received e-learning course rating at 4. At the end of data cleansing, the dataset has no missing value. According to Figure 10, we can see that each column has 3659 rows of data, which is complete and free of missing values.

```
In [9]: # merge both dataframe
        df = pd.merge(df_ratings, df_items, on="item_id")
        df.head(3)

Out[9]:
```

	user_id	item_id	watch_percentage	rating	name	created_at	Theme	Accessibility	Analyze	Collaborate	...	Discover	Mobility	Newfeatures	Organize
0	224557	510	100	10	What is OneDrive for Business?	2016	Discover	0	0	0	...	1	0	0	0
1	224293	510	100	10	What is OneDrive for Business?	2016	Discover	0	0	0	...	1	0	0	0
2	224442	510	100	10	What is OneDrive for Business?	2016	Discover	0	0	0	...	1	0	0	0

3 rows × 22 columns

Fig. 7 Merge both datasets

```
In [10]: # Only keep the last record if user rates the same content multiple times
         df = df[~df[['user_id','item_id','rating']].duplicated(subset=['user_id','item_id'], keep='last')]
```

Fig. 8 Remove duplicated data

```
In [11]: # Data filtration is applied according to 2 criteria.
         # 1st criterion : Only users which rate at least 4 books must be preserved
         more_rated_users = (df['user_id'].value_counts() >= 4)
         more_rated_users = more_rated_users[more_rated_users==True].index
         df = df[df['user_id'].isin(more_rated_users)]

         # 2nd criterion : Maintain the least received e-learning course rating at 4
         df = df[df['rating']>=4]
```

Fig. 9 Data filtration

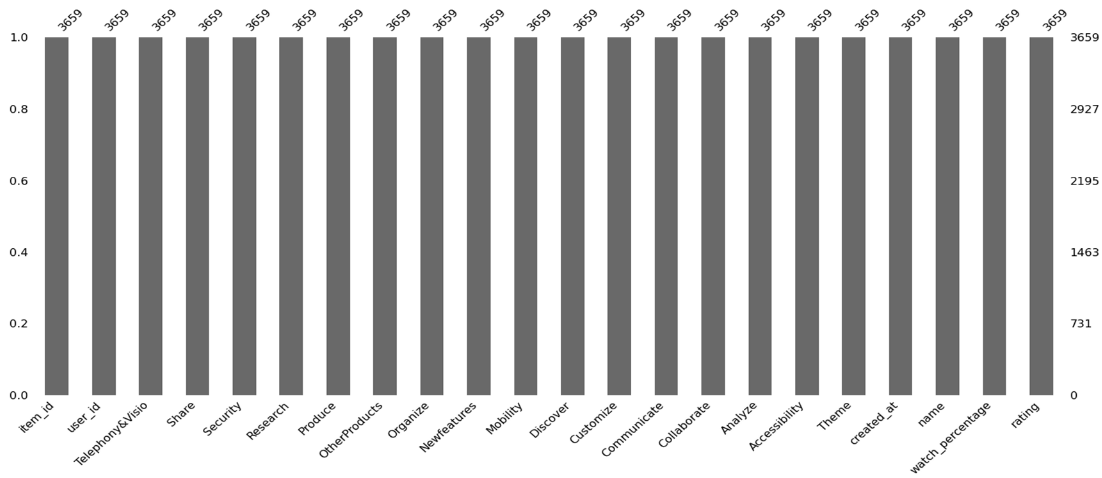

Fig. 10 Missing value in dataset

4) Recommender Engine: At this stage, the preprocessing of the dataset is considered completed. The user-item matrix is prepared for fitting into the recommendation system. The Python library used in this research project is named Surprise. This library is an easy-to-use library that enables users to swiftly construct a rating-based recommender system without the need for reinventing the wheel. Moreover, this library also provides various helpful built-in features specifically for constructing a RS, including train-test split function and different accuracy metrics for evaluation purposes.

III. Results and Discussion

In the framework, users can choose one Machine Learning technique for generating a recommendation. The first tab of the prototype is shown as Figure 11.

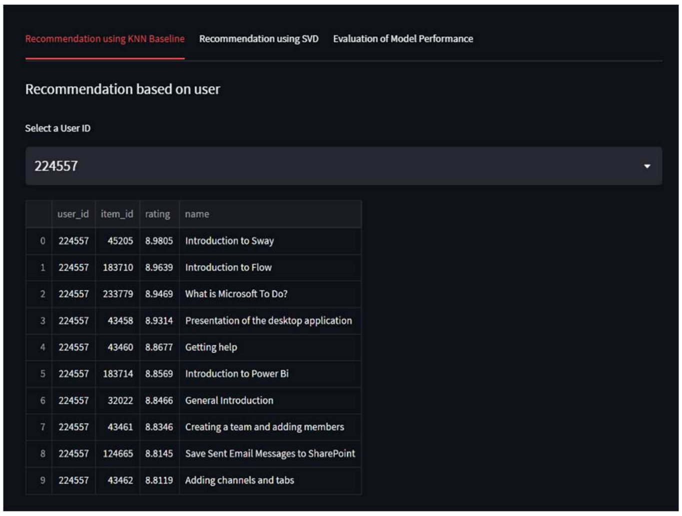

Fig. 11 Recommendation using KNNBaseline model

The first tab in this prototype shows the generation of recommendations based on the chosen user using KNNBaseline model. The selection of user ID is set to the first selection by default. The recommendation is generated according to the selection of user ID. All user ID present in the dataset is available for selection in this section. The second tab is similar to the first one except the second one uses the SVD model. The second tab is shown as Figure 12.

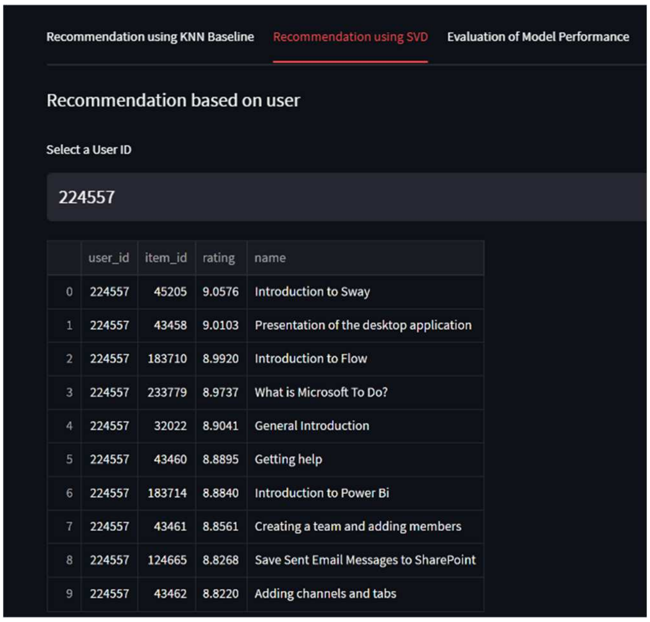

Fig. 12 Recommendation using SVD model

The third tab, which is the last tab, shows the evaluation of model performance using several evaluation metrics. The RMSE and MAE of different models are visualized as shown in Figure 13. From Figure 13, it is clear that model SVD has a higher performance than the KNNBaseline model as the value of RMSE and MAE obtained from employment of the SVD model is both lower than KNNBaseline model.

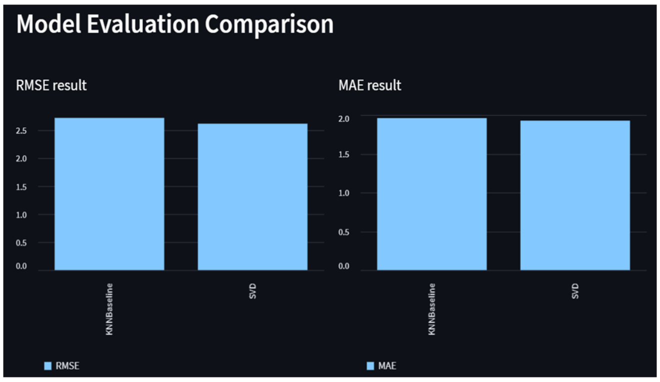

Fig. 13 RMSE and MAE result

In addition, the cross-validation result is also presented under the third tab. This method aims to evaluate Machine Learning models by performing model training on several Machine Learning models on a certain number of subsets of the dataset and evaluating the returned accuracy result. This can reduce the chances of overfitting. The cross-validation result is presented in Figure 14. According to Figure 14, there are 5 subsets divided in total. The information presented in the cross-validation table includes the RMSE and MAE value of each subset of the test-set and the time required for fitting and testing data. Overall, the SVD model outperforms the KNNBaseline model. Last but not least, the user can exit this program by closing the streamlit tab in the browser.

Evaluating applied model on 5 split(s)

KNNBaseline model

	test_rmse	test_mae	fit_time	test_time
0	2.6779	1.7670	0.0502	0.0226
1	2.7483	1.7455	0.0481	0.0285
2	2.7357	1.7511	0.0453	0.0243
3	2.6914	1.7333	0.0574	0.0305
4	2.9664	1.9316	0.0569	0.0240

SVD model

	test_rmse	test_mae	fit_time	test_time
0	2.4559	1.6578	0.0565	0.0080
1	2.5541	1.7379	0.0480	0.0080
2	2.5788	1.7670	0.0480	0.0080
3	2.6635	1.8150	0.0500	0.0070
4	2.4502	1.6565	0.0647	0.0016

Fig. 14 Cross Validation Table

The dataset is divided into 2 portions, 80% of the dataset reserved for model training and 20% for model testing. After completing model training, the fitted model can now generate predictions of course rating using the test data. The performance of the recommendation system adopting different machine learning approaches is evaluated using several evaluation metrics, including MAE and RMSE. In order to ease the process of interpreting the evaluation result, a bar chart is plotted better to visualize the comparison between each recommendation system approach. The accuracy score obtained from models applying different machine learning techniques is all collected and included in the result depicted in Table IV.

TABLE IV
MAE AND RMSE EVALUATION RESULTS

Machine Learning Technique	Evaluation metric	
	MAE	RMSE
KNNBaseline	1.9601	2.7200
Singular value decomposition (SVD)	1.9226	2.6161

According to Table IV, the accuracy score obtained by applying the KNNBaseline technique is 1.9601 for MAE and 2.7200 for RMSE, while the accuracy score obtained by applying the SVD technique is 1.9226 for MAE and 2.6161 for RMSE. Overall, the SVD model performs more accurately than the KNNBaseline model. The RMSE score generally tends to be higher than the MAE values. This is possible considering the procedures are often correct, but glitches are significant if they occur, thus explaining why the RMSE score is greater. For practical applications in industry, it is not recommended to choose a memory based CF as this method requires precomputation before adding data to the database, and this would result in the application loading necessarily slow. The database would gradually become larger in the long-term aspect, and the time required for data processing could only be longer than before. Hence, a model-based CF, for example: SVD algorithm, would be most recommended as this method contributes to a high accuracy and short data processing time output.

IV. CONCLUSION

Various Machine Learning (ML) techniques are explored, and a certain level of understanding is gained during background research and literature review in the field of e-learning course recommender systems. This paper presents a study on various types of recommenders. It also provides some insights to practitioners on implementing e-learning recommender systems and contributes to the literature by providing a comprehensive overview. The study also postulates the topics that need attention and gives future research trends.

In future work, we could implement some other ML techniques such as CoClustering and Linear Regression with more extensive evaluation such as F1 score, Precision, and Recall. In addition, the outcome could be integrated with some graphical user interface and chart for better visualization and analysis.

REFERENCES

[1] E. Dalal, P. Singh, "Comparative analysis of various recommendation systems", New Approaches for Multidimensional Signal Processing, Springer Singapore, pp. 187–194, 2021. doi:.10.1007/978-981-33-4676-5_14.

[2] M. Srivastava, "YouTube and Movie Recommendation System Using Machine Learning", IEEE International Conference on Electronics and Renewable Systems, pp.1352-1356, 2023. doi:10.1109/ICEARS56392.2023.10084999

[3] S.C. Haw, L.J. Chew, K. Ong, K.W. Ng, P. Naveen, E.A. Anaam, "Content-based Recommender System with Descriptive Analytics", Journal of System and Management Sciences, vol. 12(5), pp. 105-120, 2022. doi:10.33168/JSMS.2022.0507

[4] R.K. Mishra, J.A.A Jothi, S. Urolagin, K. Irani, "Knowledge based topic retrieval for recommendations and tourism promotions", International Journal of Information Management Data Insights, vol. 3(1), 100145, 2023. doi:10.1016/j.jjimei.2022.100145

[5] S.C. Haw, L.J. Chew, K.W. Ng, P. Naveen, "Ontology-based Recommender System with Descriptive Analytics in e-Commerce", IEEE International Conference on Big Data Engineering and Education, pp. 47-52, 2022. doi:10.1109/BDEE55929.2022.00015

[6] C. Susaie, C.K. Tan, P.Y. Goh, Learning Experience with LearnwithEmma, Journal of Informatics and Web Engineering, vol. 1(2), pp. 30-44, 2022. doi:10.33093/jiwe.2022.1.2.3

[7] A.A. Zaveri, R. Mashood, S. Shehmir, M. Parveen, N. Sami, M. Nazar, AIRA: An Intelligent Recommendation Agent Application for Movies, Journal of Informatics and Web Engineering, vol. 2(2), pp. 72-89, 2023. doi:10.33093/jiwe.2023.2.2.6

[8] A. Peuker, T. Barton, "Recommendation Systems and the Use of Machine Learning Methods", Apply Data Science: Introduction, Applications and Projects, pp. 79-93. 2023.

[9] D.T. Tran, J. H. Huh, J," New machine learning model based on the time factor for e-commerce recommendation systems", The Journal of Supercomputing, pp. 1-46, 2022. doi:10.1007/s11227-022-04909-2

[10] S. Khademizadeh, Z. Nematollahi, F. Danesh, "Analysis of book circulation data and a book recommendation system in academic libraries using data mining techniques", Library & Information Science Research, vol. 44(4), 101191, 2022. doi:10.1016/j.lisr.2022.101191

[11] Z. Jiang, L. Zhang, L. Zhang, W. Wen," Investor sentiment and machine learning: Predicting the price of China's crude oil futures market", Energy, vol. 247, 12347, 2022. doi:10.1016/j.energy.2022.123471

[12] Ishwarappa, J. Anuradha, "A Brief Introduction on Big Data 5Vs Characteristics and Hadoop Technology", Procedia Computer Science, vol. 48, pp. 319-324, 2015. doi:10.1016/j.procs.2015.04.188

[13] R.J. Kuo, H.R. Cheng, "A content-based recommender system with consideration of repeat purchase behavior", Applied Soft Computing, vol. 127, 109361, 2022. doi:10.1016/j.asoc.2022.109361

[14] R. Yin, K. Li, G. Zhang, J. Lu, "A deeper graph neural network for recommender systems", Knowledge-Based Systems, vol. 185,105020, 2019. doi:10.1016/j.knosys.2019.105020

[15] Y. Deldjoo, M. Schedl, P. Cremonesi, G. Pasi, "Recommender Systems Leveraging Multimedia Content", ACM Computing Surveys, vol. 53(5), pp. 1–38, 2021. doi:10.1145/3407190

[16] S. Wu, F. Sun, W. Zhang, X. Xie, B. Cui, "Graph Neural Networks in Recommender Systems: A Survey", ACM Computing Surveys, vol. 55(5), pp. 1–37, 2022. doi:10.1145/3535101

[17] J.H. Zelaya, C. Porcel, J.B. Moreno, A.T. Lorente, E.H. Viedma, "New technique to alleviate the cold start problem in recommender systems using information from social media and random decision forests", Information Sciences, vol. 536, pp. 156-170, 2020. doi:10.1016/j.ins.2020.05.071

[18] I.H. Sarker, A. Colman, J. Han, A.I. Khan, Y.B. Abushark, K. Salah, "BehavDT: A Behavioral Decision Tree Learning to Build User-Centric Context-Aware Predictive Model", Mobile Networks and Application, vol. 25, pp. 1151–1161, 2020. doi:10.1007/s11036-019-01443-z

[19] K.K. Jena, S.K. Bhoi, T.K. Malik, K.S. Sahoo, N.Z. Jhanjhi, S. Bhatia, F. Amsaad, "E-Learning Course Recommender System Using Collaborative Filtering Models", Electronics, vol. 12(1), pp. 157, 2022. doi:10.3390/electronics12010157

[20] A. Jeejoe, V. Harishiv, P. Venkatesh, S.K.B. Sangeetha, "Building a Recommender System Using Collaborative Filtering Algorithms and Analyzing its Performance", Advances in Science and Technology, vol. 124, pp. 478-485, 2023. doi:10.4028/p-1h18ig

[21] L.J. Chew, S.C. Haw, S. Subramaniam, "A hybrid recommender system based on data enrichment on the ontology modelling", F1000Research, vol. 10, 937, 2021. doi:10.12688/f1000research.73060.1

[22] S.M. Asaad, K.Z. Ghafoor, H. Sarhang, A. Mulahuwaish, "Point-of-Interests Recommendation Service in Location-Based Social Networks: A Survey, Research Challenges, and Future Perspectives", Sustainable Smart Cities: Theoretical Foundations and Practical Considerations, pp. 43-64, 2022. doi:10.1007/978-3-031-08815-5_4

[23] F.O. Isinkaye, Y.O. Folajimi, B.A. Ojokoh, "Recommendation systems: Principles, methods and evaluation", Egyptian Informatics Journal, vol. 16(3), pp. 261–273, 2015. doi:10.1016/j.eij.2015.06.005

[24] J. Wei, J. He, K. Chen, Y. Zhou, Z. Tang, "Collaborative filtering and deep learning based recommendation system for cold start items", Expert Systems with Applications, vol. 69, pp. 29-39, 2017. doi:10.1016/j.eswa.2016.09.040

[25] J. Tarus, Z. Niu, B. Khadidja, "E-Learning Recommender System Based on Collaborative Filtering and Ontology", International Journal of Computer and Information Engineering, vol. 11(2), pp. 256-261, 2017. doi:10.5281/zenodo.1129067

[26] J.K. Tarus, Z. Niu, D. Kalui, "A hybrid recommender system for e-learning based on context awareness and sequential pattern mining", Soft Computing, vol. 22, pp. 2449–2461, 2018. doi:10.1007/s00500-017-2720-6

[27] S. Wan, Z. Niu, "A Hybrid E-Learning Recommendation Approach Based on Learners' Influence Propagation", IEEE Transactions on Knowledge and Data Engineering, vol. 32(5), pp. 827-840, 2019. doi:10.1109/TKDE.2019.2895033

[28] H. Ezaldeen, M. Misra, S.K. Bisoy, R. Alatrash, R. Priyadarshini, "A hybrid E-learning recommendation integrating adaptive profiling and sentiment analysis", Journal of Web Semantics, vol. 72, 100700, 2022. doi:10.1016/j.websem.2021.100700

[29] Z. Shahbazi, Y. Byun, "Agent-Based Recommendation in E-Learning Environment Using Knowledge Discovery and Machine Learning Approaches", Mathematics, vol. 10(7), pp. 1192, 2022. doi:10.3390/math10071192

[30] R. Alatrash, R. Priyadarshini, H. Ezaldeen, A. Alhinnawi, "Augmented language model with deep learning adaptation on sentiment analysis for E-learning recommendation", Cognitive Systems Research, vol. 75, pp. 53-69, 2022. doi:10.1016/j.cogsys.2022.07.002

[31] D. Pramod, P. Bafna, "Conversational recommender systems techniques, tools, acceptance, and adoption: A state of the art review", Expert Systems with Applications, 117539, 2022. doi:10.1016/j.eswa.2022.117539

[32] M. Hafsa, P. Wattebled, J. Jacques, L. Jourdan, "Multi-objective recommender system for corporate MOOC", The Genetic and Evolutionary Computation Conference Companion, pp. 2314–2317, 2022. doi:10.1145/3520304.3534058