P- ISSN 1991-8941 E-ISSN 2706-6703
2024,(18), (02):231 – 240
Journal of University of Anbar for Pure Science (JUAPS) Open Access

Machine Learning Techniques and Recommender Systems for Massive Educational Data

Ammar Abbood Mohammed[1*], Murtadha M. Hamad[2]

[1,2] Department of Computer Science, College of Computer and Information Technology, University of Anbar, Ramadi, Iraq;

ARTICLE INFO

Received: 03/ 02 /2024
Accepted: 23/ 04 /2024
Available online: 28/ 12 /2024

10.37652/juaps.2024.146489.1187

Keywords:
Machine Learning, Singular Value Decomposition, K Nearest Neighbor, Hit Rate, Grid Search.

ABSTRACT

E-learning approaches allow learners to choose appropriate courses from a range of courses on online platforms on the basis of their preferences and interests. Recommender systems help students select courses by analyzing their data and understanding their interests. In the domain of e-learning platforms, recommender systems that utilize machine intelligence (MI) techniques can be improved. This study aims to propose recommendations for e-learning courses on the basis of user assessments. To achieve this aim, this study applied a collaborative filtering technique. An educational dataset from Coursera's online courses was utilized. Machine learning (ML) models, such as K nearest neighbor (KNN) and singular value decomposition (SVD), were proposed. Evaluations were conducted using performance metrics, such as root mean square error (RMSE), mean absolute error (MAE), hit rate (HR), average reciprocal hit rating (ARHR), accuracy, and recall. Results showed that SVD and KNN achieved 91% and 96% HR, respectively. Among the models, KNN obtained the highest accuracy of 99%, followed by SVD with 96%. When the grid search technique was utilized with the recommended ML algorithms, SVD and KNN obtained 99% and 98% HR, respectively. Although accuracy improved only slightly after the changes, these results still highlight the efficacy of collaborative filtering methods in improving recommendations in e-learning settings, thereby enhancing the customized learning experiences of users.

Introduction

At present, people rely heavily on recommendation systems because these systems can help them quickly find the items they need [1]. Technological advancements have made e-learning platforms increasingly effective for learning [2]. E-learning pertains to the use of electronic platforms for educational purposes; it facilitates asynchronous learning, where students have the flexibility to engage in learning activities at their own convenience and from any location [3]. In contrast to conventional educational methods, remote learning does not require learners to physically attend classes; instead, they may participate from the convenience of their homes [4]. E-learning has contributed to the rise of reskilling, upskilling, and the enhancement of the conventional education system via its provision of knowledge dissemination. This meaningful learning approach is based on a constructivist methodology that involves the conceptual modeling of an individual's existing and prior knowledge or experiences, with the aim of achieving personalization [5].

*Corresponding author at: Department of Computer Science, College of Computer and Information Technology, University of Anbar, Ramadi, Iraq.
ORCID:https:// https://orcid.org/0000-0000-0000-0000 ,
Tel: +964 7813311137
Email: amm21c1017@uoanbar.edu.iq

Rapidly identifying the courses that users are interested in among vast amounts of data can contribute considerably to the dissemination of accurate knowledge [6]. One element of the e-learning process is distance learning, which enables individuals to exchange information despite restrictions and geographical boundaries. In colloquial language, e-learning refers to the use of electronic devices to access educational curricula outside of the conventional classroom setting [7]. Recommendation systems help users navigate overwhelming amounts of new and unfamiliar information [8]. With the expansion and evolution of the Internet, many challenges arise from information overload [9]. Recommendation systems depend on collaborative filtering techniques to efficiently cater to customer preferences [10]. The aim of a recommendation system is to predict courses that users want on the basis of specified preferences [11].

This paper is organized into the following sections. Section 2 presents related studies, and Section 3 introduces the proposed research method. Section 4 provides the results and discussions, and Section 5 presents the conclusions and future work directions.

Related Work

Dede et al. (2019) proposed an intelligent recommendation system designed for students. This system utilizes a combination of classification algorithms, including support vector machine (SVM), naïve Bayes (NB), decision tree (DT), and K nearest neighbor (KNN). The most accurate predictions of the grade point average were achieved by DT (91%), KNN (83%), SVM (80%), neural network (NN) (75%), and NB (75%). The proposed method combines vectors and neural networks [12]. Omer et al. (2021) proposed various machine learning (ML) techniques to predict student academic achievement at universities. This study used several ML techniques, such as KNN, linear regression (LR), NN, NB, DT, and random forest (RF), to examine factors, such as age, gender, urban area, family income, and family education. The results showed that the RF algorithm exhibits excellent performance in obtaining predictions and has an accuracy of 89.39% [13]. Khaoula et al. (2021) proposed the integration of convolutional neural networks and long short-term memory to study emotion in massive open online courses (MOOCs). The main goal was to establish methods that can improve the learning opportunities of students. This work examined the relationship between sentiments expressed in discussion posts and failure rates detected in MOOCs. The proposed model was compared with other models, and it achieved the highest accuracy of 91.27% among all the models [14]. Sara et al. (2022) utilized data mining techniques to help students find the MOOC that matches their requirements. They examined how students navigate MOOC platforms and predicted their motivation levels. They applied four ML models, namely, NB, LR, SVM, and RF. All of the models provided predictions (ranging from 89% to 95%) regarding which courses would ignite a student's interest, but RF had the highest accuracy. This study also discussed the limitations of recommending courses on the basis of content while disregarding each student's needs [15]. Kalyan et al. (2023) suggested recommender systems for e-learning courses on the basis of filtering. The objective was to assist students in selecting courses that align with their interests. The research utilized ML models, including KNN, SVD, and network-based collaborative filtering. The results indicated that the KNN model, with a hit rate (HR) of 96.46%, outperforms other models. Its mean absolute error (MAE) values are also lower than those of other models [16]. Amir et al. (2023) proposed a recommender system designed to determine classroom learning strategies on the basis of students' preferred learning styles. The system employs filtering techniques; specifically, it utilizes the NB algorithm. Through this approach, a system can achieve an accuracy rate of 90.91% in recommending learning strategies [17].

Methodology

This work proposes an intelligent recommendation system for massive educational data on the basis of Coursera's Course Reviews Dataset. It aims to achieve e-learning course recommendation. The process involves detecting similarities, streamlining the prediction process, and providing recommendations. This work uses two machine intelligence (MI)-based models: KNN and SVD. The proposed system architecture includes data collection, preprocessing, recommendation algorithms, and performance metrics, which are explained in detail in the following subsections.

General Structure of the Proposed System

The flowchart of the proposed system is illustrated in Figure 1. The first phase of the work involves loading a dataset, followed by the application of preprocessing techniques to the dataset in the second phase. Subsequently, the dataset is divided into subdatasets for training and testing purposes in the third phase, where two algorithms ML are applied. Then, automated services are provided to users as they make course choices. The aim is to recommend several e-learning courses to learners.

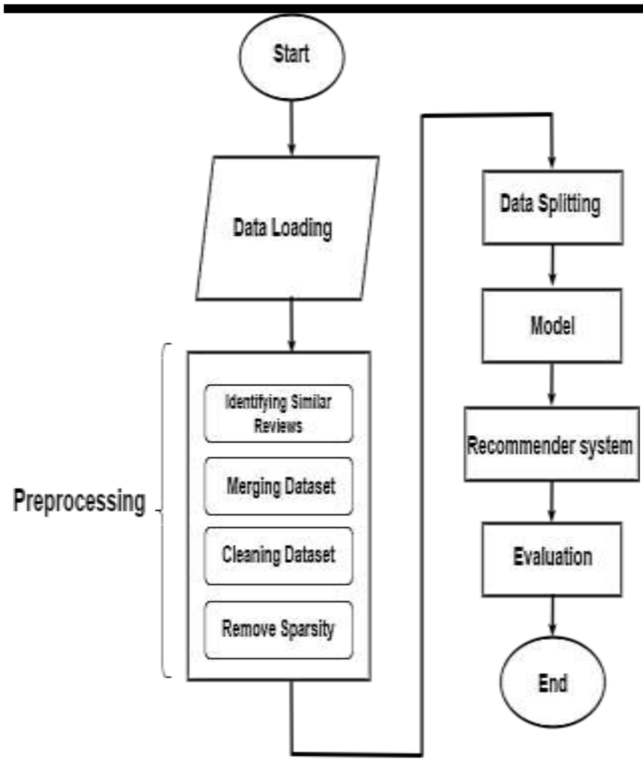

Figure 1. Flowchart System

The fundamental phases of the proposed system are illustrated in detail in Figure 2.

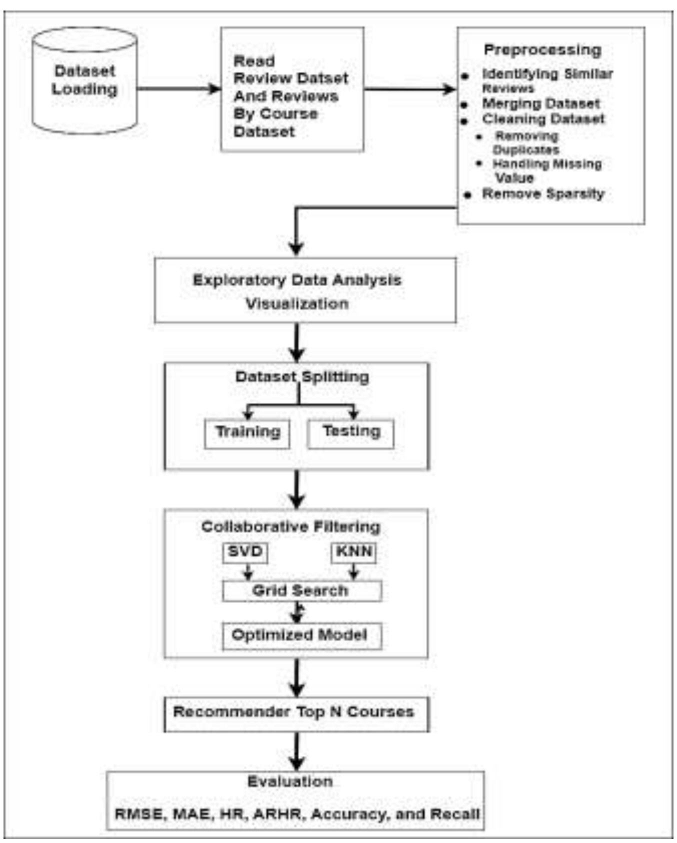

Figure 2. Proposed System's Diagram

The proposed system's phases are explained in detail in the subsequent sections.

Data Acquisition Sources

Datasets are obtained from the Kaggle website [18], which is a platform for data science competitions where participants compete to create the best models for solving specific problems or analyzing certain datasets. The dataset comprises assessments of Coursera courses. The dataset consists of two comma-separated values (CSV) files, namely, reviews.csv, which has 107,018 rows and 3 columns for ID, Review, and Label, and reviews_by_course.csv, which contains 140,320 rows and columns for Course-ID, Review, and Label. Courses are evaluated using a five-star rating scale, with one, two, three, four, and five stars denoting very negative, negative, neutral, positive, and very positive ratings, respectively.

Dataset Preprocessing

Preprocessing is conducted on the CSV dataset. The steps include identifying similar reviews, combining data, eliminating entries, addressing missing values, and reducing sparsity. These steps aim to enhance data quality and facilitate data use through recommendations. The preprocessing phase plays a role in organizing the data for efficient recommendation tasks. The following sections present the details of this useful process.

Identifying Similar Reviews

This subsection shows the process of comparing two columns of reviews (reviews.csv and reviews_by_course.csv) in the datasets and includes finding and extracting reviews from both columns. We take the intersection of these columns, where a subset of reviews that appear in both datasets can be found. Perceptions are displayed as shared sentiments or thoughts. Such an analysis is useful when overlapping evaluations exist across courses or categories. The intersection operation helps us focus on the elements within the reviews, allowing for an understanding of similar emotions or themes present in the dataset. The number of similar reviews is 100,020.

Merging Datasets

In this phase, the information contained in two datasets is merged into one dataset. This dataset combination is important for data preparation and analysis and ML purposes. The combination of datasets can help in understanding complicated subjects and extracting valuable knowledge. The dataset obtained after combination consists of 664,418 rows and includes four columns: Student-ID, Course-ID, Review, and Label.

Dataset cleaning

Data cleaning is important because it increases the quality and usefulness of data. This study employs data cleaning to remove all invalid or wrong information, leaving us with the highest information quality possible. Data cleaning methods include removing duplicates and missing values. After cleaning, the dataset consists of 451,494 rows and 4 columns.

Removing Sparsity

An important step in the proposed technique is dealing with the sparsity found in the dataset. Sparsity refers to having a dataset quantity of zero or having missing values within the dataset. A limited number of features can negatively affect the performance of ML models and lead to increased computational problems. To solve this issue, we implement an approach for removing or replacing characteristics to enhance the dataset's overall quality and the model's efficiency [4].

User–Item Matrix by Using the Pivot Table Function

This study creates a user–item matrix by utilizing the axis table function. The first step is to organize and structure the data in a matrix format. In this format, users and items are represented by rows and columns, respectively. The elements of the matrix correspond to relations, ratings, or relevant metrics between users and items. The axis table function transforms the dataset into this matrix, thus making analysis and modeling easy. This matrix is the basis of the filtering tasks.

Building the Rating Matrix

A matrix data structure is constructed to capture the relationships among students, courses, and their associated ratings after dataset combination. We build a rating matrix by extracting the student ID, course ID, and rating from the combined datasets. In this matrix, the columns represent the course ID, and the rows represent the student ID. Each intersection between a student and a course contains a cell value that represents the rating.

Compressed Sparse Row

Compressed sparse row (CSR) is used in representational linear algebra and numerical computation to handle matrices. It properly stores null values in their row and column indices to reduce memory usage and enhance performance. This format has three arrays: data, index, and non-zero elements. It is useful for tasks that require individual access to rows, such as matrix vector multiplication and ML algorithms, and helps achieve balance between memory efficiency and resource usage. Notably, CSR cannot handle not-a-number (NAN) values in the user–object matrix. To overcome this obstacle, we use a rating matrix construction method to convert any missing value (NAN) to zero.

Exploratory Data Analysis

Plots, histograms, and box plots are commonly used to graphically display patterns and distributions, and statistical measures, such as correlation coefficients, provide valuable insights into possible correlations. The primary goals of exploratory data analysis (EDA) are to generate hypotheses, explore patterns worthy of further investigation, and provide guidelines for building sophisticated research models.

Visualization

Visualization plays a role in data analysis and communication by simplifying statistics into an easy-to-understand format. It uses aids, such as charts, diagrams, and maps, to transform complex information into meaningful pictures. EDA relies heavily on visualization techniques because it allows researchers to interact with data and extract insights. Figure 3 shows a representation of the top five courses within our dataset, and Figure 4 presents the top five courses with the highest number of reviews.

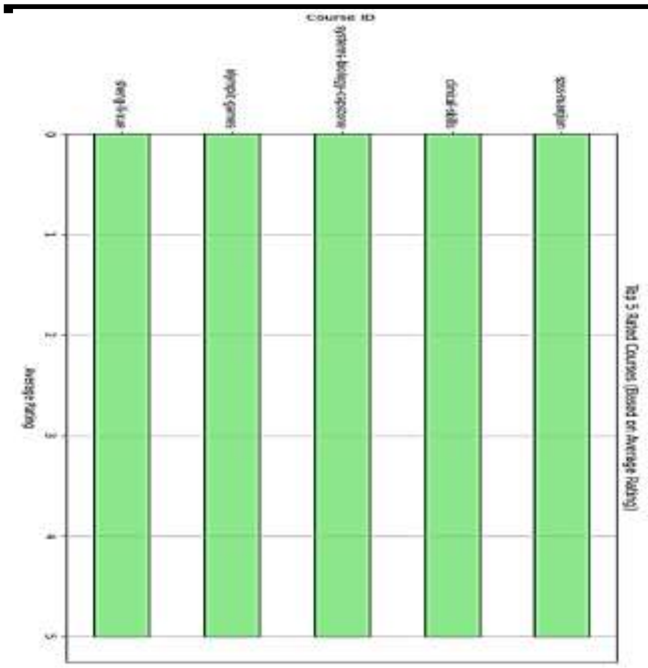

Figure 3. Bar Plot of the Top Five Rated Courses

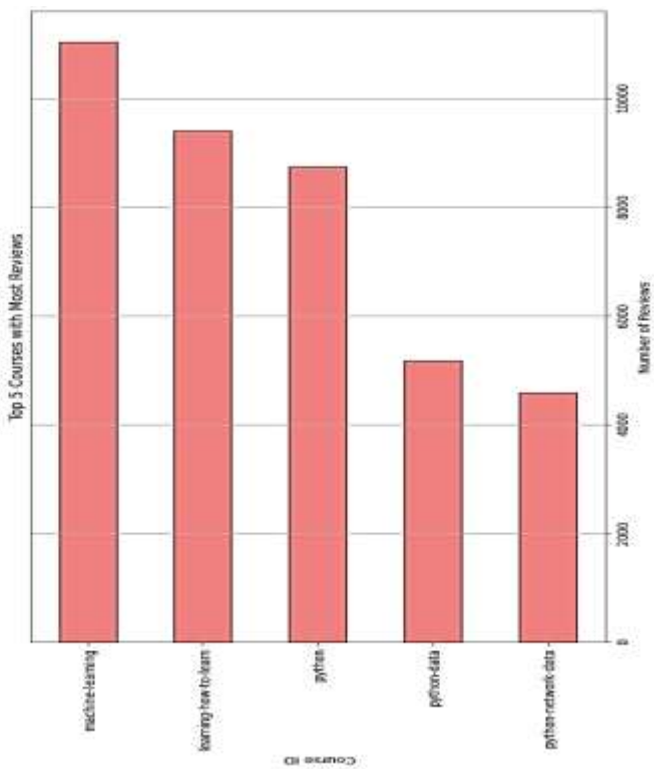

Figure 4. Bar Plot of the Top Five Courses with the Most Reviews

Dataset Splitting Phase

The dataset splitting phase, which comes after preprocessing, divides the dataset into two parts. The first part is used to train the model, and the second part employs untrained data for efficiency and accuracy testing. Table 1 displays the ratios used in the training and testing processes.

Table 1. Ratios Used

Training	Testing
80%	20%
70%	30%
60%	40%

Proposed Algorithms

The work constructs e-learning course recommendations by using SVD, KNN, and a grid search (GS) model for optimization. The aim is to enhance the reliability and robustness of the conclusions. The algorithms are applied to the proposed system to achieve high-accuracy recommendations. The details of applying these algorithms are provided in the next subsections.

SVD

SVD is among the most efficient algorithms for recommender systems [19]. Initialization is a challenge for recommender systems because of the repetitive nature of SVD algorithms, which considerably affects the algorithms' convergence and performance [20]. With the help of SVD, a matrix is decomposed into three other matrices, as specified in Equation (1), where H is the (a x b) matrix; I is the (a x c) orthogonal matrix, which is a left singular matrix that represents the relationship between the learner and latent factors; J is the (c x c) diagonal matrix, which shows the strength of latent factors; and V is the (c x b) orthogonal matrix, which is a right singular matrix that reflects course and latent factor similarity. The latent factors are course features derived from the topic or area of the e-learning course. SVD decreases the dimensionality of utility matrix H by eliminating its latent elements. It assigns each student and course to a latent space with c dimensions. This mapping enables a transparent depiction of the connections between learners and courses [16]. Figure 5 shows how SVD works

$$\mathrm{H} = IJV^{M} \quad (1)$$

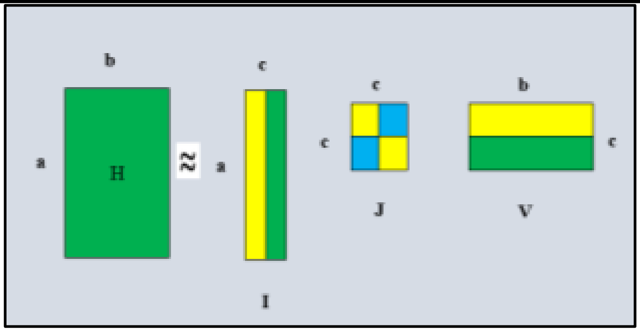

Figure 5. How the SVD Algorithm Works [19]

KNN Algorithm

KNN is a supervised learning technique used for classification tasks. It utilizes training data to create distance-based predictions via the Euclidean distance measure. The KNN approach is employed in recommendation systems, and it consists of numerous steps. The first step requires finding the value that reflects the number of nearest neighbors to be assessed. The next step involves computing distances from the training set to determine the nearest neighbors. Calculation of the Euclidean distance may be performed using Equation (2). The arrangement of distances is determined by the minimum value of k [21]. Figure 6 shows the KNN technique [22].

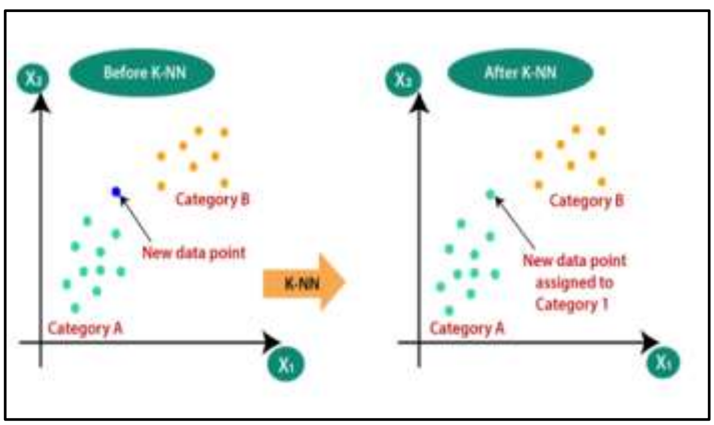

Figure 6. KNN Technique [22]

$$\mathbf{sim(p,q)}=\sqrt{\sum_{j=1}(pj-qj)^2}, \qquad (2)$$

where pj is the total variable of p and qj is the total variable of q.

GS to Optimize the SVD Model

The recommendation system development process involves using GS to fine-tune hyperparameters. One of the hyperparameters being examined is the number of latent components, along with the choice between randomized or arpack as the SVD method and the number of convergence iterations. The algorithm works by breaking down the user–course interaction matrix and making predictions for the test set on the basis of the combinations of these hyperparameters by using SVD. GS runs through predetermined ranges for algorithms, iteration counts, and latent factors. For each combination, it calculates RMSE between predicted and actual scores on the filtered test set. The goal is to identify the hyperparameter configuration that minimizes RMSE and achieves an accuracy rating that is comparable to that of the model.

GS to Optimize the KNN Model

The GS approach is utilized to fine-tune the parameters of the KNN model, with the aim of optimizing its performance. GS involves systematically exploring various combinations of hyperparameters, such as the number of neighbors (K). The primary objective is to identify the configuration that maximizes the model's accuracy and effectiveness in making recommendations.

Measurement of the Performance of the Recommendation System

Metrics are used to assess the recommended items' ranking. Every recommendation model employs a separate dataset in an attempt to resolve an issue in its own unique way [23]. The confusion matrix displays the dataset's present state and the proportion of true and false model predictions [24]. This matrix is shown in Table 2 [25].

Table 2. Confusion Matrix of Recommendations

	Relevant	Irrelevant
Recommended	**TP**	**FP**
Not recommended	**FN**	**TN**

The metrics utilized in this study are as follows:

A. Root Mean Square Error (RMSE): It is the standard measure used for model evaluation. It is computed by accumulating the squared deviations between the predicted rating and the actual rating, dividing the sum by the number of test points, and taking the square root of the resulting quotient [25]. The relevant equation is as follows:

$$\text{RMSE}=\sqrt{\frac{\sum_{i=1}^{n}(r_i-pi)2}{n}}\ . \qquad (3)$$

B. MAE: Measures the average of the absolute differences between the predicted rating and the actual rating. Prediction accuracy increases as MAE decreases [23]. It may be computed using the equation

$$MAE = \frac{\sum_{i=1}^{n} |ri - pi|}{n}, \quad (4)$$

where r1, r2, r3,..., rn are the actual ratings and p1, p2, p3,..., pn are the predicted ratings.

C. HR: This metric delineates the proportion of course hits within the test set relative to the summation of hits and misses. Its formal representation involves the variables denoting the number of course hits (NCH) and the aggregate of true negatives and true positives, which is the total number of test cases (TNTC) [16]. The following equation can be used to calculate HR.

$$HR = \frac{NCH}{TNTC} \quad (5)$$

D. Average Reciprocal Hit Rate (ARHR): This measure calculates the ratio of the sum of the reciprocals of the rank of each hit (SRR) to TNTC. It measures the reciprocal rank of hits' relative measure with respect to the set of test cases that were used [16]. It may be calculated as

$$ARHR = \frac{SRR}{TNTC}. \quad (6)$$

E. Accuracy: It is the number of correct estimates (TP + TN) divided by the total number of instances (TP + TN + FP + FN) [24]. Accuracy can be calculated using the equation

$$Accuracy = \frac{TP+TN}{(TP+TN+FP+FN)}. \quad (7)$$

F. Recall: It refers to the ratio of positive cases that are correctly classified to the total number of occurrences whose true class is positive [26]. Recall can be calculated as

$$Recall = \frac{TP}{(TP+FP)}. \quad (7)$$

Results and Discussion

This section includes an analysis of the outcomes of the e-learning recommendation system and the evaluation criteria employed for performance assessment. The outcomes of KNN and SVD prior to the optimization procedure are presented in Tables 3, 4, and 5 and Figures 7, 8, and 9. The results of applying the GS algorithm to each model are given in Tables 6, 7, and 8 and Figures 10, 11, and 12. The findings provide an overview of the suggested approach and the employed metrics, such as RMSE, MAE, HR, ARHR, accuracy, and recall. Evaluation is conducted for the optimized and nonoptimized models. The tables display a comparative analysis of different models to aid in the selection of the most appropriate model for the e-learning recommendation system. The KNN algorithm with 96% HR demonstrates better performance compared with the SVD algorithm (91% HR). However, after optimization, the SVD algorithm achieves a remarkable HR of 99%, which surpasses the KNN algorithm's post-optimization HR of 98%. The proportions are 20% and 80% for the testing and training phases, respectively.

Table 3. Model Performance Evaluation at 80:20 Training–Testing Ratio before Optimization

Model	SVD	KNN
RMSE	0.73	0.459
MAE	0.182	0.043
ARHR	0.235	0.169
HR	0.91	0.959
Accuracy	0.959	0.99
Recall	0.955	0.95

Table 4. Model Performance Evaluation at 70:30 Training–Testing Ratio before Optimization

Model	SVD	KNN
RMSE	0.61	0.46
MAE	0.146	0.044
ARHR	0.25	0.159
HR	0.9	0.96
Accuracy	0.96	0.99
Recall	0.95	0.95

Table 5. Model Performance Evaluation at 60:40Training –Testing Ratio before Optimization

Model	SVD	KNN
RMSE	0.55	0.46
MAE	0.128	0.044
ARHR	0.25	0.161
HR	0.9	0.96
Accuracy	0.95	0.99
Recall	0.93	0.95

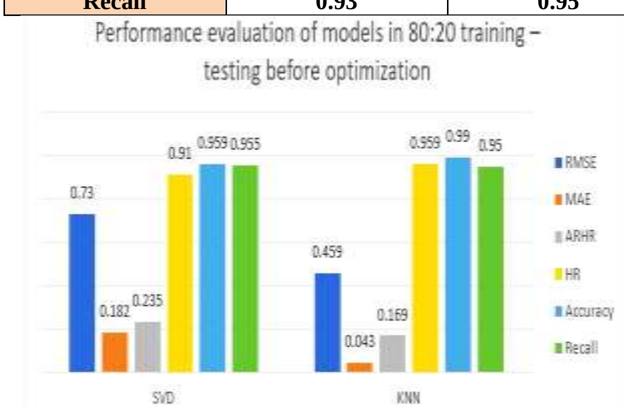

Figure 7. Model Performance Evaluation before Optimization at 80:20 Ratio

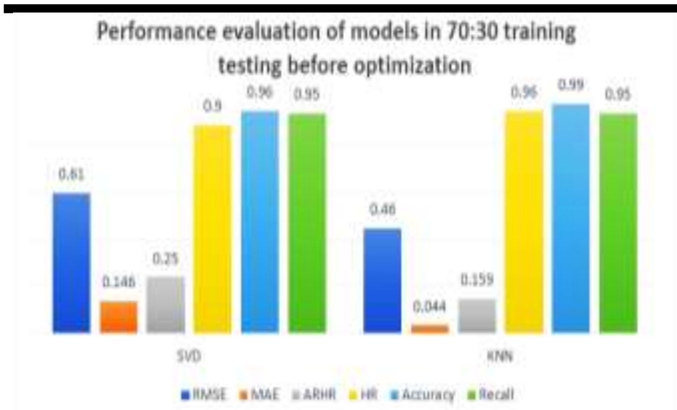

Figure 8. Model Performance Evaluation before Optimization at 70:30 Ratio

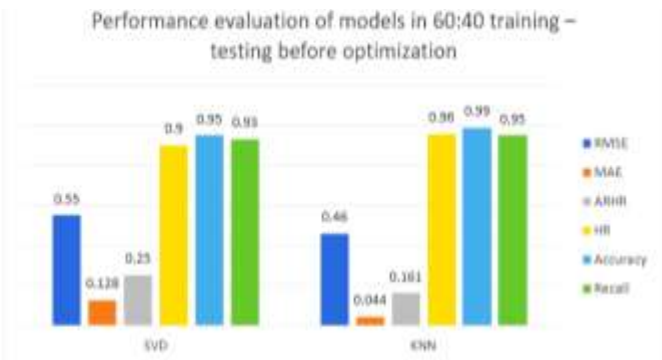

Figure 9. Model Performance Evaluation before Optimization at 60:40 Ratio

Table 6. Model Performance Evaluation at 80:20 Training–Testing Ratio after Optimization

Model	SVD	KNN
RMSE	0.59	0.46
MAE	0.161	0.043
ARHR	0.41	0.171
HR	0.99	0.98
Accuracy	0.957	0.99
Recall	0.952	0.96

Table 7. Model Performance Evaluation at 70:30 Training–Testing Ratio after Optimization

Model	SVD	KNN
RMSE	0.54	0.46
MAE	0.134	0.044
ARHR	0.46	0.159
HR	0.99	0.98
Accuracy	0.96	0.99
Recall	0.95	0.95

Table 8. Model Performance Evaluation at 60:40 Training–Testing Ratio after Optimization

Model	SVD	KNN
RMSE	0.53	0.46
MAE	0.118	0.044
ARHR	0.32	0.161
HR	0.93	0.98
Accuracy	0.95	0.99
Recall	0.93	0.94

Figure 10. Model Performance Evaluation after Optimization at 80:20 Ratio

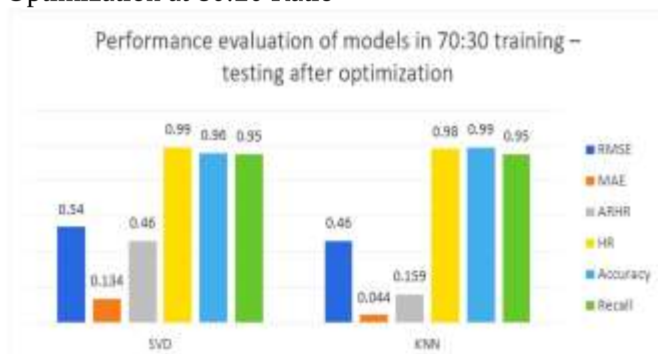

Figure 11. Model Performance Evaluation after Optimization at 70:30 Ratio

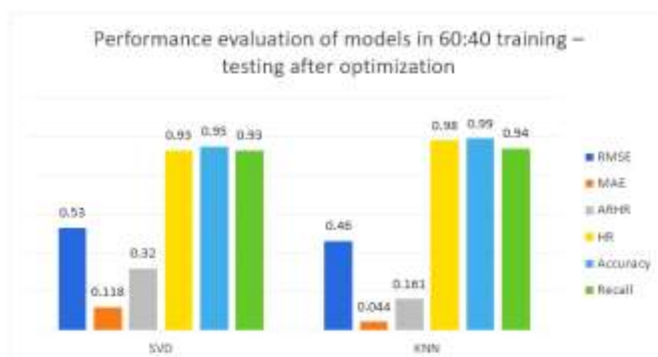

Figure 12. Model Performance Evaluation after Optimization at 60:40 Ratio

Conclusions and Future Work Directions

This study presents an e-learning recommendation system that is based on collaborative filtering. The system uses models, such as KNN and

SVD, to generate recommendations. Several performance metrics, including RMSE, MAE, HR, ARHR, accuracy, and recall, are utilized to evaluate the performance of the models and the effectiveness and accuracy of learning recommendation algorithms. The aim of the study is to evaluate the effectiveness of the collaborative filtering recommendation process and identify models that provide accurate recommendations to learners. Doing so can help students by reducing the time and effort required to search for relevant courses. The recommender system examines learners' preferences and activities, recognizes patterns and similarities, and utilizes this information to recommend courses to the user. The primary goal is to enhance the learning experience by offering a curated selection of e-learning courses that align with user preferences.

Future research could expand the proposed e-learning course recommendation system by developing upgraded and hybrid models to enhance the recommendation results. Additionally, the topics discussed in this study can be further explored and applied in the context of deep learning models.

References

[1] G. Behera and N. Nain, "Collaborative Filtering with Temporal Features for Movie Recommendation System," *Procedia Comput. Sci.*, vol. 218, pp. 1366–1373, 2023, doi: 10.1016/j.procs.2023.01.115.

[2] V. B. P. Tolety and E. V. Prasad, "Hybrid content and collaborative filtering based recommendation system for e-learning platforms," *Bull. Electr. Eng. Informatics*, vol. 11, no. 3, pp. 1543–1549, 2022, doi: 10.11591/eei.v11i3.3861.

[3] A. A. Nafea *et al.*, "Enhancing Student ' s Performance Classification Using Ensemble Modeling," *Iraqi J. Comput. Sci. Math.*, vol. 4, no. 4, pp. 204–214, 2023.

[4] B. Rawat, J. K. Samriya, N. Pandey, and S. C. Wariyal, "WITHDRAWN: A comprehensive study on recommendation systems their issues and future research direction in e-learning domain," *Mater. Today Proc.*, no. March 2021, 2020, doi: 10.1016/j.matpr.2020.09.796.

[5] H. Ezaldeen, S. K. Bisoy, R. Misra, and R. Alatrash, "Semantics aware intelligent framework for content-based e-learning recommendation," *Nat. Lang. Process. J.*, vol. 3, no. April, p. 100008, 2023, doi: 10.1016/j.nlp.2023.100008.

[6] B. Li *et al.*, "A personalized recommendation framework based on MOOC system integrating deep learning and big data," *Comput. Electr. Eng.*, vol. 106, no. October 2022, p. 108571, 2023, doi: 10.1016/j.compeleceng.2022.108571.

[7] A. Moubayed, M. Injadat, A. B. Nassif, H. Lutfiyya, and A. Shami, "E-Learning: Challenges and Research Opportunities Using Machine Learning Data Analytics," *IEEE Access*, vol. 6, no. c, pp. 39117–39138, 2018, doi: 10.1109/ACCESS.2018.2851790.

[8] A. Lopes and B. Amaral, "A machine learning approach for mapping and accelerating multiple sclerosis research," *Procedia Comput. Sci.*, vol. 219, pp. 1193–1199, 2023, doi: 10.1016/j.procs.2023.01.401.

[9] L. V. Nguyen, M. S. Hong, J. J. Jung, and B. S. Sohn, "Cognitive similarity-based collaborative filtering recommendation system," *Appl. Sci.*, vol. 10, no. 12, pp. 1–14, 2020, doi: 10.3390/APP10124183.

[10] M. M. H. Muntaha Kamal Chyad, "A Proposed Movie Recommender System to Solve Sparsity, Cold Start and Diversity Problems using Clustering Algorithms," *Solid State Technol.*, vol. 63, no. 6, pp. 2762–2774, Oct. 2020, Accessed: Apr. 07, 2023. [Online]. Available: http://www.solidstatetechnology.us/index.php/JSST/article/view/3179

[11] L. Abed, M. Hamad, and A. Aljaaf, *A review of marketing recommendation systems*, vol. 2591. 2023. doi: 10.1063/5.0119651.

[12] R. Systems *et al.*, "A proposed framework in an intelligent recommender system for the college student A proposed framework in an intelligent recommender system for the college student," 2019, doi: 10.1088/1742-6596/1402/6/066100.

[13] О. Ф. Акмеше, Х. Кьор, and Х. Ербей, "Use of Machine Learning Techniques for the Forecast of Student Achievement in Higher Education," *Inf. Technol. Learn. Tools*, vol. 82, no. 2, pp. 297–311, 2021, doi: 10.33407/itlt.v82i2.4178.

[14] K. Mrhar, L. Benhiba, S. Bourekkache, and M. Abik, "A Bayesian CNN-LSTM Model for Sentiment Analysis in Massive Open Online Courses MOOCs," *Int. J. Emerg. Technol. Learn.*, vol. 16, no. 23, pp. 216–232, 2021, doi: 10.3991/ijet.v16i23.24457.

[15] S. Assami, N. Daoudi, and R. Ajhoun, "Implementation of a Machine Learning-Based

MOOC Recommender System Using Learner Motivation Prediction," pp. 68–85.

[16]K. K. Jena *et al.*, "E-Learning Course Recommender System Using Collaborative Filtering Models," *Electron.*, vol. 12, no. 1, 2023, doi: 10.3390/electronics12010157.

[17]A. Saleh, N. Dharshinni, D. Perangin-Angin, F. Azmi, and M. I. Sarif, "Implementation of Recommendation Systems in Determining Learning Strategies Using the Naïve Bayes Classifier Algorithm," *Sinkron*, vol. 8, no. 1, pp. 256–267, 2023, doi: 10.33395/sinkron.v8i1.11954.

[18]"100K Coursera's Course Reviews Dataset | Kaggle."https://www.kaggle.com/datasets/septa97/100k-courseras-course-reviews-dataset (accessed Jul. 28, 2023).

[19]T. Widiyaningtyas, M. I. Ardiansyah, and T. B. Adji, "Recommendation Algorithm Using SVD and Weight Point Rank (SVD-WPR)," *Big Data Cogn. Comput.*, vol. 6, no. 4, 2022, doi: 10.3390/bdcc6040121.

[20]T. Huang, R. Zhao, L. Bi, D. Zhang, and C. Lu, "Neural Embedding Singular Value Decomposition for Collaborative Filtering," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 33, no. 10, pp. 6021–6029, 2022, doi: 10.1109/TNNLS.2021.3070853.

[21]D. R. Anamisa, A. Jauhari, and F. A. Mufarroha, "K-Nearest Neighbor Method for Recommendation System in Bangkalan's Tourism," *ComTech Comput. Math. Eng. Appl.*, vol. 14, no. 1, pp. xx–xx, 2023, doi: 10.21512/comtech.v14i1.7993.

[22]K. Hiran, R. Jain, K. Lakhwani, and R. Doshi, "Machine Learning: Master Supervised and Unsupervised Learning Algorithms with Real Examples (English Edition) ٢٠٢١ "،, Accessed: Nov. 30, 2023. [Online]. Available: https://books.google.com/books?hl=ar&lr=&id=4VVDEAAAQBAJ&oi=fnd&pg=PT25&dq=Machine+Learning+Maste+Supervised+and+Unsupervised+Learning+Algorithms+with+Real+Examples&ots=OM8aBAQ7oM&sig=i7lb8ZvEz5Y7aqYxJLtFjOzHG9M

[23]J. Shokeen and C. Rana, "Social recommender systems: techniques, domains, metrics, datasets and future scope," *J. Intell. Inf. Syst.*, vol. 54, no. 3, pp. 633–667, 2020, doi: 10.1007/s10844-019-00578-5.

[24]M. Yağcı, "Educational data mining: prediction of students' academic performance using machine learning algorithms," *Smart Learn. Environ.*, vol. 9, no. 1, 2022, doi: 10.1186/s40561-022-00192-z.

[25]S. Review, "MF-RISE : Benchmarking for Multifaceted Recommender System Engine MF-RISE : Benchmarking for Multifaceted Recommender System Engine," 2023.

[26]A. A. Nafea, N. Omar, and Z. M. Al-qfail, "Artificial Neural Network and Latent Semantic Analysis for Adverse Drug Reaction Detection," *Baghdad Sci. J.*, 2023.

تقنيات التعلم الآلي وأنظمة التوصية للبيانات التعليمية الكبيرة

عمار عبود محمد ، مرتضى محمد حمد

قسم علوم الحاسبات، كلية علوم الحاسوب وتكنولوجيا المعلومات ، جامعة الأنبار، الأنبار ، العراق
Email: amm21c1017@uoanbar.edu.iq

الخلاصة:

يقدم التعلم الإلكتروني للمتعلمين مجموعة من الدورات التدريبية على المنصات، مما يسمح لهم بالاختيار وفقًا لتفضيلاتهم واهتماماتهم. تلعب أنظمة التوصية دورًا في مساعدة الطلاب على اختيار المقررات الدراسية من خلال تحليل بياناتهم وفهم اهتماماتهم. في مجال منصات التعلم الإلكتروني، هناك إمكانية لتحسين أنظمة التوصية باستخدام تقنيات الذكاء الآلي (MI). تهدف هذه الدراسة إلى تقديم توصيات مقترحة لدورات التعلم الإلكتروني بناءً على تقييمات المستخدم، ولتحقيق هذا الهدف، يجب تطبيق تقنية التصفية التعاونية. استخدمت هذه الدراسة مجموعة البيانات التعليمية من دورات كورسيرا عبر الإنترنت. اقترحت هذه الدراسة نماذج التعلم الآلي (ML) مثل K Nearest Neighbor (KNN) وتحليل القيمة المفردة (SVD) بينما تستخدم هذه الورقة تقييمات مقاييس الأداء مثل جذر متوسط مربع الخطأ (RMSE)، ومتوسط الخطأ المطلق (MAE)، ومعدل الإصابة (HR)، ومتوسط تصنيف الضربات المتبادلة (ARHR)، والدقة، والاستدعاء. تظهر النتائج أن SVD حقق ٩١% من HR وKNN حقق ٩٦% من HR بينما حققت خوارزمية KNN أعلى دقة بنسبة ٩٩%، تليها SVD بدقة ٩٦% ولكن عند استخدام تقنية Grid Search (GS) مع حققت خوارزميات ML الموصى بها معدل ضربات القلب بنسبة ٩٩٪ لـ SVD و٩٨٪ لـ KNN. على الرغم من أن الدقة لم تظهر سوى تحسينات طفيفة بعد التغييرات، إلا أن هذه النتائج تسلط الضوء على فعالية طرق التصفية التعاونية في تحسين التوصيات في إعدادات التعلم الإلكتروني، وبالتالي تعزيز تجارب التعلم المخصصة للمستخدمين.

الكلمات المفتاحية :التعلم الآلي، تحليل القيمة المفردة، أقرب جار K، معدل الإصابة، بحث الشبكة.