DHM-OCR: A Deep Hybrid Model for Online Course Recommendation and Sustainable Development of Education

Original Scientific Paper

Sagar Mekala
Department of Computer Science and Engineering,
CVR College of Engineering, Telangana, Hyderabad, India
msagarphd@cvr.ac

Padma TNS
Assistant Professor, Department of CSE- Data Science,
Sreenidhi Institute of Science and Technology, Hyderabad, India

Rama Rao Tandu
Department of Computer Science and Engineering,
CVR College of Engineering, Telangana, Hyderabad, India

Abstract *– In recent years, there has been an increase in online education resources to help learners improve their skills. However, it is difficult to select the right course from available online education resources due to the demands and needs of learners with different knowledge domains. To solve this problem, an online course recommendation model has the important factor of enhancing learner's knowledge. Many existing recommendation systems (RS) use collaborative filtering (CF) to recommend courses to learners. The major problems with the Collaborative Filtering Recommendation System (CFRS) are the sparse preferences and the scalability of the data. According to the similarity of items, many recommendation models are proposed and developed, but none of these provide suggestions to users without their associations or preferences. We propose a deep hybrid model-online course recommendation (DHM-OCR) that uses high-level learner behavior and course objective features. We demonstrate the improvements and efficiency of the model for suggesting online e-learning courses. According to the analysis and evaluation results, it seems that our DHM-OCR outperforms the parallel research recommendation system. Experimental findings from online course data reveal that the suggested model and approach significantly improve classification accuracy and training efficiency, particularly limited available data.*

Keywords*:* *Recommendation System, Convolutional Neural Network, Content Based Recommendation System, Collaborative Filtering Recommendation System, Deep Hybrid Model, Ranking, Recurrent Neural Networks, Similarity, Preferences*

Received: November 1, 2023; Received in revised form: January 2, 2024; Accepted: January 5, 2024

1. INTRODUCTION

During and after the COVID-19 epidemic, the rapid development of online course providers focused on online course learners. As per parallel research on the online education system, those who have completed courses through the online learning system have benefited and enhanced their skill set in their domain area. Online education resources such as Udamy, MIT, Coursera, EdX, and Swayam are widespread and increasing in number. Courses learned through these sources, learners can develop academically and professionally. Many course modules and sub-modules are available through these online education platforms. These platforms offer hundreds of courses to learners. Parallel research results on online education platforms indicate that learners face great difficulty selecting an appropriate course due to the many online course providers and the large set of course data [1].

The recommendation system is a process of suggesting and/or predicting a priority set of items based on the user's interest and is constructed for many applications such as e-learning, games, movies, and e-commerce to simplify human daily life. As per the previous study, recommendation systems have received greater attention in academic and research circles. The recommendation system (RS) is a model that can obtain and produce a higher indexed and/or ranked collection of items in which users might be interested, according to their existing and present choices of items. The goal of RS is to make daily life easier for people by proposing and/or forecasting a prioritized list of goods based on the user's interests. RS makes user's daily activities on the internet and social media easy. RS is used in many daily applications, such as news article suggestions, movie and music recommendations, e-learning, e-commerce, and many more. Online course recommendation models are divided into four basic approaches: content-based recommendation systems (CBRS), collaborative filtering recommendation systems (CFRS), hybrid recommendation systems (HRS), and sequence-based recommendation systems (SBRS). With recent advancements in information technology, many algorithms and schemes have been adapted to enhance the performance of RS.

The Content-Based Recommendation System (CBRS) approach is built based on users' preferences. In the CBRS model, the items are suggested based on the user's previous preference information, data, or user profile. According to the CBRS, suggestions can be made mainly from the data files and past preferences. The CBRS is useful to solve the problem of cold starts. In the Collaborative Filtering Recommendation System (CFRS) [2, 3] technique, items are suggested or predicted for a particular user when other similar kinds of users also prefer those items. The CFRS focuses mainly on finding similarities between learners and courses. The outcomes of CFRS conclude that learners with similar behaviors would take similar online courses [4]. It can recommend courses to target learners based on similar behaviors, but it is difficult to solve the cold start problems. A mixed approach combining CBRS and CFRS enhanced the performance of the model and reduced the cold-start problem. A specialized recommendation model proposed for course recommendation is called the Sequence Based Recommendation System (SBRS). It is mainly based on updating data through sequences, learning behavior sequences, and the time series approach of the learner. A tripartite graph, known as a location-query-browse graph (LQB), is proposed to provide sophisticated contextual suggestions [5]. REQUEST a query-based RS [6] proposed to utilize a multidimensional recommender system paradigm, including contextual dimensions, user and item dimensions, and OLAP-type aggregation and filtering capabilities. Social psychology-based CF for recommendations proposed in [7]. Authors [8], proposed a social recommender system that utilizes a temporal clustering technique.

Parallel research has discussed the applications of deep learning, collaborative filtering, and content-based recommendation systems in online course recommendation systems. For example, academics or industries have used CBRS to model learners' browsing histories and construct their preferences for courses. Shan et al. [9] proposed a model for e-learning recommendation based on CNN. CNN is used to extract course features from a text dataset of e-learning resources, such as introductions and content. The primary goal of the online course recommendation model is to predict learner preferences for personalized and preferable courses. For learners, they can study online through learning resources to gain knowledge and skills in their domain of interest. And for service providers, they can prepare and upload all kinds of course materials with different objectives.

The study on the integrated DHM framework with an intelligent recommendation system for online course assistance is strongly justified for several compelling reasons. The traditional online learning experience often lacks personalized guidance and tailored recommendations for individual learners. This study addresses the need to enhance the overall learning experience by providing intelligent recommendations that align with the unique preferences, capabilities, and learning styles of each student. With the proliferation of online courses, students may feel overwhelmed by the abundance of available resources. An intelligent recommendation system helps in efficient resource allocation, directing learners to the most relevant and beneficial courses and materials. The Deep Hybrid Model (DHM) has become a major domain of research because of the increasing amount of sequential data available in online learning education. DHM uses this to construct a learner's course preferences and interests based on their past course objectives and behavior. It is evident that CBRS and CFRS are accurate to some extent, but they fail to extract the complex features and dependencies in sequential data. All these online learning recommendation models have high content, but most of them use recommendation techniques without considering the distinction of behavior, preferences, objectives, and time of learners, which are crucial parts of learning in online learning. Most of the recommendation models for online learning don't consider the course objectives to be accurate and necessary to reach learner goals. For every course, there are certain objectives. If the similarity of two or more course objectives is high, such courses will be suggested to target and present learners. We proposed a deep hybrid model (DBM) that combines a Convolutional Neural Network (CNN) and a Recurrent Neural Network (RNN) to learn semantic features of learners and courses. And build a sequence model for the courses. In our point of view, users with similar learning behaviors have similar preferences, so our proposed DHM consists of CNN and RAN to learn the learner's behavior and focuses on time series. The primary contributions in this article are summarized as follows:

A framework for modeling and analyzing the e-learning learner-course induced data.

A deep hybrid model consisting of CNN and RNN is used together to construct an efficient learning process and enhance classification precision.

A DHM-OCR with CNN and RNN is designed to recognize and extract the combination of semantic and time series characteristics.

A recommendation model to predict learners with course guidance and course objective advice based on various training mechanisms for intelligent online course support

Successive sections in this paper are organized as follows. A brief introduction related to the work is presented in Section 2. We present the intelligent online course recommendation system framework with DHM in Section 3. In Section 4, we introduce a deep hybrid model experimental framework and evaluation results based on the online learner course. We conclude this article in Section 5.

2. RELATED WORK

A Recommendation System with a Deep Hybrid Model utilizes more than one deep learning approach. The advantage of DHM makes it possible to merge several neural building blocks to complement one another and form a more efficient hybrid model.

Garivaldis et al. [10], presented how existing learning theories might improve online education. In recent years huge surveys done and published papers on recommendation systems (RS), for instance, Adomavicius et al [11] presented a comparative survey on content-based, collaborative filtering, and hybrid recommendation models. The authors highlighted some of the important limitations of various recommendation systems. The possible merits and demerits of the hybrid recommender system model were studied by Burke et al [12]. Park et al [13], presented basic and real-world application areas of recommendation system approaches. Many algorithms and approaches have been published, but these papers impact a particular area of recommender system development. Zhang Q et al [14] analyzes commonly used E-learning recommendation algorithms and suggests new research possibilities. This study presents content-based (CBR), collaborative filtering-based (CFBR), and knowledge-based recommendations (KBR). These techniques and how they meet E-learning needs are explained.

To enhance the accuracy of RS a trust model collaborative filtering scheme is introduced by Jiang et al. [15]. Trust awareness-based CFRS implemented [16] for user behavior filtering. Social psychology-based CF for recommendations developed by Liu et al. [17]. Yang et al. [18], proposed a model to adapt SRL techniques to design a hybrid job recommender by combining content-based and collaborative filtering. Chadem et al. [19] implemented a recommendation system based on user-based and item-based collaborative filtering by introducing time decay, and item–to–item similarity models based on time intervals for suggesting top N-items.

Deshpande et al. [20], specifies the user-item matrix, item-rating matrix, and their binary values. To construct similarity matrices Sarwar et al. [21] use the cosine and correlation-based approaches. Hui Chen et al. [22], introduced a recommendation model to specify the behavior of e-learners and proposed an adaptive model based on learner behavior. Parvatikar et al. [23] have adapted different approaches for suggesting books to purchase. They primarily according to the book-based collaborative filtering apply data mining schemes. Implementation results of their approach simplify the scalability and data sparsity problems in Badarerah et al. [24] proposed a model for course selection and recommendation according to the similarities among learners and to construct the cluster with similar learners they used K-mean and KNN algorithms to get the most closed learner clusters. Initially, generates the learner clusters as per the e-learning style and performs collaborative filtering and association mining to retrieve the user behavior and interests.

Dien et al. [25], propose a deep matrix decomposition approach that recommends learning courses depending on learners' skills and needs. The authors provide to researcher with a simpler explanation of conventional matrix factorization and deep matrix factorization. The authors performed a comparison of the deep and conventional matrix factorizations. The presented model may be used for course recommendation and mapped to a recommendation system issue. Altaf et al. [1] presented various distinct models for course performance prediction. The methodology developed in this research is broad enough to produce accurate predictions based on many inputs. The proposed system uses 1D Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks to analyze online learning data, which is complicated and diverse. This research's hybrid deep learning approach has increased accuracy in visualizing advanced study specialists and aided weak pupils in achieving precision instruction. Li [26] a deep learning-based course recommender system (DECOR) is proposed to record user behaviors and course attributes. DECOR reduces information overload, solves high-dimensional data sparsity, and increases feature extraction performance. Biletskiy et al. [27], proposed ontological models of learners and learning objects, as well as methods for determining and adjusting preferences using these models. Based on a student profile and learning objectives, adaptive e-learning systems may automatically create individualized learning experiences [28]. Modern model-based hybrid recommendation systems need substantial feature engineering to create a user profile that combines these two filtering methods. Authors in [18, 29], suggested using SRL state-of-the-art techniques to build a hybrid course and job recommendation system. Helping recommender system interfaces grow is the goal of this TOCHI special area [30]. The Alternating Least Squares (ALS) algorithm-based collaborative filtering recommendation system prototype was proposed [31]. Multiple applications benefit from hybrid recommender systems. A hybrid fuzzy linguistic recommender system to assist the Technology Transfer Office personnel in distributing user-friendly research materials is proposed [32].

We advocate scaling educational best practices to additional students and institutions and developing sustainable future practices. Using enduring education theory and practice to develop learning methodology makes the past relevant to the present and future and allows innovation to be scaled and expanded from one impact to many.

3. DEEP HYBRID BASED MODEL FOR ONLINE COURSE RECOMMENDATION (DHM – OCR)

An introduction to recommender systems explains the current generation of content-based, collaborative, and hybrid recommendation techniques [11]. This study also analyzes present recommendation technique limitations and suggests additions to increase recommendation capabilities and make recommender systems more versatile.

In this section, we propose to design an intelligent approach to provide learners with automatic e-learning course resource guidance based on the Deep Hybrid Model-Online Course Recommendation System (DHM-OCRS). In many cases, learners may find it difficult to identify appropriate e-learning resources and to get the course updated on time. Here we try to identify an effective way to suggest a more efficient recommendation based on the learner's course learning process, and personalized course inquiry. The proposed DHM model consists of CNN, which is used to identify and extract features from data sources. RNN can model the feature dynamics and time series of content data. More than one deep learning algorithm is combined to construct effective recommendation models. These deep hybrid models consist of deep architectures that contain both discriminative and generative components.

3.1. ANALYSIS OF LEARNER – COURSE GENERATED DATA

The online course recommendation system has two main modules: an intelligent learner behavior learning module and an intelligent course suggestion module. The former is used to guide a learner to online learning resources and courses based on his/her inquiries, simulating the role of the guiding teacher, while the latter predicts medicine names based on learner behavior, simulating the role of the instructor. For text feature extraction, DCNN (deep convolutional neural network) uses two parallel CNNs to predict learner behavior and course objectives from the dataset. This model addresses the sparsity problem and enhances personalized recommendations by extracting similar features from learner feedback texts. The extracted features are passed through a single convolutional layer (SCL) with various kernels, a max-pooling layer (MPL), and a fully connected layer (FCL) consecutively. The outputs of the learner network (Li) and course network (Ci) are finally annotated. It is input to the prediction layer (RNN), where the matrix factorization technique is applied to combine learner and course interaction for rating prediction. In the following, we discuss in detail the main functional components of our DHM.

We utilize a convolutional neural network (CNN) to map learner preferences to other course vectors. Convolutional operations construct the considerable local similarities of a learner. Formally, CNN can learn the optimal learner behavior and preference specifications for course recommendation. The CNN contains an SCL, an MPL, a FCL, and a CL. The primary role of the convolutional layer is to obtain local features from the learner and course data. The CNN for learner-course local feature extraction can be defined in equation 1. RNN is another important approach of deep learning neural networks for text extraction, generally used on the data with a sequential representation. An RNN model consists of the RNN layer, FCL, and a CL. The primary job of the RNN layer is to obtain sequential dependencies. $X = F(Y, \Theta)$ of RNN can be specified in equation 2.

$$L_c = classifier(FC(Max\_Pool(CoNN(X,\theta_{CNN}),\theta_{MP}), \theta_{FC})) \quad (1)$$

$$L_c = classifier\ (FC(RNN(X, \theta_{RNN}), \theta_{FC})) \quad (2)$$

CNN models effectively use spatial data characteristics, such as pictures. Traditional CNN cannot handle sequential data. On the other hand, RNN-based models excel in modeling sequential data, such as conversations. To create a new model, DHM, a mixture of CNN and RNN is suggested in equation 3. Since the input data contains short learner questions and course datasets, we created the DHM model based on CNN and RNN. In our DHM, a max pooling layer precedes the RNN layer, generating two max pooling layers. The deep hybrid model is in equation 4.

$$Lc = classifier\ (FC(RNN(max\_pool\ (CNN(X, \theta_{CNN}), \theta_{pool}), \theta_{RNN}), \theta_{FC})) \quad (3)$$

$$Lc=classifier(FC(max\_pool\ (RNN(max\_pool\ (CNN(X, \theta_{CNN}), \theta_{pool}), \theta_{RNN}), \theta_{pool}),\theta_{FC})) \quad (4)$$

Where the classifier describes the classification layer, θ_{CNN} denotes collection of attributes of the function CNN, FC specifies the fully connected layer, θ_{FC} is collection of attributes of the function Fully Connected Layer (FC), Max_Pool denotes the pooling layer, and θ_{MP} denotes collection of attributes of the procedure max pooling layer., function RNN specifies RNN Layer, θ_{RNN} denotes collection of attributes of the procedure RNN, FC represents the fully connected layer, θ_{FC} collection of attributes of the method Fully Connected Layer. Semantic features are retrieved via the max-pool layer after convolution, followed by sequential RNN feature extraction and another max-pooling layer. The procedure emphasizes the combination of semantic and sequential elements to be compressed. The DHM effectively handles short input data sequences by placing the max-pooling layer with salient characteristics after the RNN layer. As an example, the training dataset may consist of labeled online learners' course questions. Set CNN as the convolutional layer and max-pool as the two pooling layers. A basic RNN is utilized for short consecutive words. Input data is sent over many layers during training, including CNN, max_pool, RNN, and complete connection. Dropout is used to avoid overfitting in classification models. The suggested DHM algorithm for online course recommendation is defined in subsequent sections.

3.2. INTELLIGENT ONLINE COURSE RECOMMENDATION FRAMEWORK

The intelligent support module assigns learners to online learning courses based on their course history. The module receives learner course history and online learning materials (course feedback, grade, learning resource}) as input. The intelligent behavior module is responsible for pre-suggesting courses based on student inquiries and the teacher replies in the database. The module receives student course history and grade (course feedback, grade, learning resource). The suggested deep hybrid model is applied in both modules for the classification of short learner history, highlighting semantic and sequential aspects. The first module classification model assigns one learning resource per input history, whereas the second module classification model assigns one or more courses per learner.

The deep hybrid model in all modules uses a single-label classification model by assigning one learning resource to input data (i.e., courses). The multi-label classification methodology gives many categories (courses) to input data that course feedback, grades, and learning resources. Thus, we use Softmax in the first module of DHM's classifier layer and Sigmoid in the second module. In the training process, models are built using three steps: 1) Training the first deep hybrid model for inquiry-based classification using the dataset {Course - C; Learning Resource-L}. 2) Clustering data in learning resources into many groupings. 3) Train the second deep hybrid model for feedback-grade categorization using the dataset {Grade – G; Feedback - F; Course-C} in each group. For intelligent online course assistance, the first inquiry-based classifier will recommend the course recourse with the greatest likelihood of using Softmax for a given learner with a new query. Using the proposed deep hybrid model; the K-means algorithm clusters data into groups, selecting the most relevant one for the query q. Finally, the second feedback-grade classifier uses Sigmoid to propose courses with probabilities over a threshold. In the mathematical analysis for the proposed deep hybrid recommender, we consider the non-functional similarity to obtain the learner's behavior. The given dataset consists of 'm' course learners and 'n' courses offered in various platforms, this can be represented as m x n a matrix we call a learner-course matrix. The following equations from 5 to 8 demonstrate the learner-course matrix, the behavior of the learner is split into two types such as learner-based similarity (LSIM) and course-based similarity (CSIM) measures. Person Correlation Coefficient (PCC) was utilized in most of the RS for finding similarity, learner-based behavior measures the PCC can be computed to measure the behavior between two learners' u, and v by using.

$$LSIM(u,v) = \Sigma_{k\epsilon C}(P_{uk} - Q_u)(P_{vk} - Q_v) / \sqrt{A}\,\sqrt{B} \tag{5}$$

Where $\sqrt{A}$ is $\Sigma_{k\epsilon C}(P_{uk} - Q_u)^2$ and $\sqrt{B}$ is $\Sigma_{k\epsilon C}(P_{vk} - Q_v)^2$, and $C = Cu \cap Cv$ is the set of courses offered by learners u, v and P_{uk} is the satisfaction rating of the learner on the course C_k, and Q_u specifies the mean value of learner 'u' over the course in 'C_k'. Rates between -1 to1 are calculated using the behavior equation. In course-based similarity measure, the PCC can be evaluated to measure the learner behavior within any two courses by using.

$$CSIM(m,n)=\Sigma_{k\epsilon C}(P_{mk} - Q_m)(P_{nk} - Q_n) / \sqrt{A}\,\sqrt{B} \tag{6}$$

Where $\sqrt{A}$ is $\Sigma_{k\epsilon C}(P_{mk} - Q_m)^2$ and $\sqrt{B}$ is $\Sigma_{k\epsilon C}(P_{nk} - Q_n)2$, and $C = Cm \cap Cn$ is the set of courses both courses taken by learners u, v and Pmk is the satisfaction rating (feedback) of the learner on the course (Cm) taken by learner 'u' denoted as P_{mk}, Q_m and specifies the mean value of course 'm'. Similarly, the satisfaction rate of course (Cn) taken by learner 'v', is denoted as P_{nk} and Q_n specifies the mean evaluated value of the course (Cn) taken by learner 'v'. Re-ranking has been introduced to obtain similarity between target learners and courses based on the time of the satisfaction rating in the learner-course matrix. Then we perform computations of learner-based and course-based similarity measures. In learner-based similarity measures, the PCC is utilized to compute the similarity within the learners' 'u' and 'v' based on other courses.

$$LSIM(u,v) = 1 - \sqrt{\Sigma_{k\epsilon C}\{A-B\}} / \sqrt{|C|} \tag{7}$$

Where A is $(f_{uk} - f_{umin}) / (f_{umax} - f_{umin})$, B is $(f_{uk} - f_{umin}) / (f_{umax} - f_{umin})$, and $C = C_u \cap C_v$ is the set of courses taken by learners 'u' and 'v', the number of courses are denoted by $|C|$, and fuk represents the course 'k' with learner 'u' value in the modified learner-course matrix. The highest and lowest values for 'u' and 'v' are specified f_{umin}, f_{umax}, f_{umin}, and fumax respectively. The learner similarity (LSIM) is shown in equation 8.

$$LSIM(u,v) = \begin{cases} 0 & iff\ u_k \neq v_k \\ 1 & otherwise \end{cases} \tag{8}$$

Where 0 indicates that learners are dissimilar while considering courses (Ck), whereas 1 indicates both learners are similar by considering the course (Ck). The course matrix is constructed with rows consisting of courses and course objectives are separated by columns. These are important for constructing a course objective matrix. The course rating matrix for each learner according to the course ID is interpreted in a binary format. Every learner has rated one or more feedback rating each course. According to the course objective matrix and course rating matrix, we constructed the dot product matrix the result is again interpreted in binary format. An algorithm designed for building the hybrid recommendation model is as follows.

Input: 1. Identified Online Learning Resources with course objectives, course title, course Instructor, platform, and Course Identification (CID) as a

dataset. 2. Learner Objectives, history, goals, and Feedbacks as a dataset

Output: Top-N courses for target learner.

1. *Construct a Course Data Frame (CDF) from the various Learning Resources.*
2. *Construct a list that contains all the columns in the course.*
3. *For every Course Objective (CO), Course Feedbacks (CF), and Course Rating (CR) in course dataset.*
4. *if CO in CDF:*
5. *if CF is positive then:*
6. *CFeatures=Classifier(FC(MaxPool(RNN(MaxPool(CNN(CO, CF))))))*
7. *For every Learner Objective (LO), Learner History (LH) from Learner dataset.*
8. *if LO in LDF:*
9. *if LH is positive then:*
10. *LFeatures = Classifier(FC(MaxPool(RNN(MaxPool(CNN(LH, LO))))))*
11. *Step 5. Predicting_Learning_ Resource = Classifier(FC(RNN(MaxPool(CNN(LF,CF)))))*
12. *For every CO in CDF and CR.*
13. *13. for every CID:*
14. *if CR gt_Eql 3:*
15. *CR_matrix=1*
16. *else*
17. *17. CR_matrix = -1*
18. *Find the dot product between course matrix and CR_ matrix.*
19. *Convert the course dot matrix to binary format.*
20. *Find the Euclidian distance between the present learner and target learners.*
21. *Consider the rows with min(Euclidian distance): they are Top-N recommended courses for the target learner.*

This model based on the sequence of learners/users preferences and interests, and courses rated by the learners. Let us consider C as the set of courses, L, R, and P as the set of learners, rating and preferences in the system. Then $Sl = < x_1, x_2, x_3, \ldots, x_n>$ represents the sequence of rating of learner of l_i. $Slp = < p_1, p_2, p_3, \ldots, p_n>$ represents the sequence of preferences and interests of learner of l_i, and x_i in the above sequence indicates that $(C_i, r/p)$ specifies preference or rating of course given by learner l_i.

4. RESULTS

We undertake experiments to assess the performance of our suggested deep hybrid model and mechanism, comparing it to parallel research techniques. The most commonly performed operations are data preprocessing which is used for eliminating and avoiding incomplete data, and to find the order of course pattern events for each learner. For each course there will be a grade value between 0 - 1, we have converted grades into new grade scales. Evaluation trials acquired data from around 50 web pages, a realistic online learning service. There were 24,423 student feedback and grades. After screening data, including removing duplicates, incorrect formats, and misclassifications, 17,432 grades, and feedback were accessible for experiments and data descriptions from various online course resources. Out of 17,432 learner data, the average feedback length is 41, with 14,301 feedbacks ranging from 10 to 70. We utilize data from the "learning resources" to evaluate the performance of grade-feedback-based categorization in the intelligent course module. Based on 130 courses mentioned in learner feedback, we suggest the top 20 most commonly used courses based on 2654 samples. These courses include Python, C, C++, Java, Data Analysis, Big Data, Business Analysis, Deep Learning, Cloud Computing, Machine Learning, Cloud Analysis, Cyber Security, PHP, Pearl, R, Network Analysis, XML, SQL, Data Visualization, and Neural Network.

In our training system, input data is transformed to a 300 x 18 matrix for parameter setup. A 32-length vector describes each character in a sentence, with a maximum of 500 characters per phrase. The CNN convolutional layer uses 256 kernels (5 x 32) to generate 256 feature maps and extract textual information in a 496 x 256 matrix. Set the first max_pool layer to get 24 x 256 feature maps, and the basic RNN layer (RNN) to extract sequential features in a 24x32 matrix. A further max_pool layer compresses the output into a 32x1 matrix, integrating semantic and sequential data effectively. In inquiry-based classification, extracted characteristics are fed via a completely connected layer, forming a 256x1 vector, and categorized by the Softmax layer based on the six departments. The inquiry-answer-based classification classifies aspects using the sigmoid layer for the 20 courses mentioned. As previously mentioned, the initial intelligent learner module uses a single-label classification algorithm to propose courses based on learner feedback and grades in previous courses. We evaluate the recommendation performance of CNN and RNN neural network models with our proposed deep hybrid model on four categories (Table 2). Accuracy, Precision, Recall, and F1 are commonly used metrics to assess the performance of three classification models. Results may be measured using following equations from 9 to 12:

NTP: associated with learning resources and suggested

NFP: suggested but not part of the available learning resource;

NFN: not recommended, but associated with a learning resource;

NTN: not recommended and not related to learning resources.

$$Accuracy = NTP / TOTAL \tag{9}$$

$$Precision = NTP / (NTP + NFP) \tag{10}$$

$$Recall = NTP / (NTP + NFN) \tag{11}$$

$$F1 = 2 * Precision * Recall / (Precision + Recall) \tag{12}$$

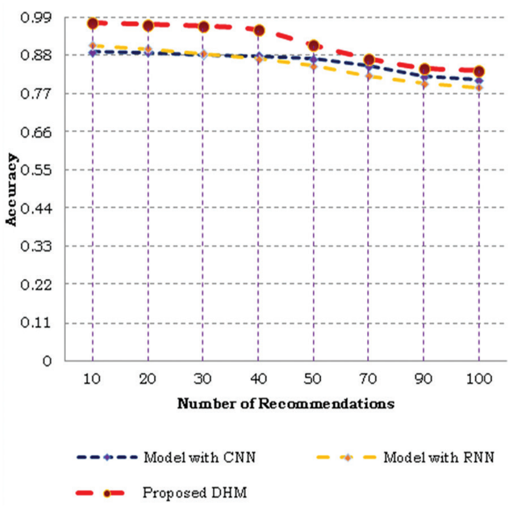
Table 2. Top 10 Recommendations Accuracy, Precision, Recall, and F1 Comparison among CNN, and RNN with proposed DHM-OCR

Course Name	Accuracy			Precision			Recall			F1		
	CNN	RNN	DHM	CNN	RNN	DHM	CNN	RNN	DHM	CNN	RNN	DHM
Data Science	0.879	0.911	0.974	0.891	0.901	0.972	0.881	0.934	0.965	0.904	0.918	0.964
Business Analysis	0.882	0.883	0.926	0.712	0.812	0.941	0.823	0.885	0.931	0.842	0.877	0.941
Machine Learning	0.929	0.932	0.944	0.909	0.949	0.967	0.909	0.898	0.946	0.871	0.895	0.946
Artificial Intelligence	0.911	0.887	0.948	0.891	0.911	0.934	0.889	0.810	0.946	0.871	0.888	0.937
Deep Learning	0.883	0.899	0.911	0.803	0.866	0.900	0.871	0.840	0.97	0.899	0.910	0.941
Web Development	0.871	0.881	0.954	0.851	0.887	0.912	0.836	0.810	0.837	0.830	0.810	0.868
Devops	0.886	0.898	0.921	0.896	0.866	0.895	0.867	0.816	0.816	0.816	0.799	0.833
Python Programming	0.885	0.895	0.949	0.885	0.918	0.932	0.885	0.811	0.821	0.905	0.905	0.912
Cyber Security	0.893	0.923	0.957	0.813	0.813	0.875	0.851	0.810	0.822	0.811	0.810	0.866
Data Visualization	0.893	0.890	0.933	0.898	0.754	0.801	0.888	0.821	0.833	0.877	0.867	0.919
Network Analysis	0.887	0.915	0.969	0.870	0.887	0.933	0.798	0.809	0.821	0.899	0.888	0.923
R	0.893	0.898	0.943	0.799	0.803	0.824	0.803	0.816	0.842	0.821	0.811	0.859
MAT Lab	0.865	0.915	0.960	0.790	0.821	0.868	0.845	0.837	0.878	0.910	0.899	0.922
Psychology	0.881	0.888	0.954	0.901	0.893	0.912	0.783	0.819	0.839	0.888	0.899	0.936
Marketing	0.892	0.879	0.945	0.892	0.910	0.926	0.881	0.801	0.866	0.865	0.851	0.895
Biology	0.890	0.876	0.955	0.755	0.846	0.875	0.789	0.827	0.850	0.805	0.833	0.878
C Programming	0.895	0.895	0.911	0.789	0.863	0.896	0.855	0.798	0.836	0.820	0.844	0.899

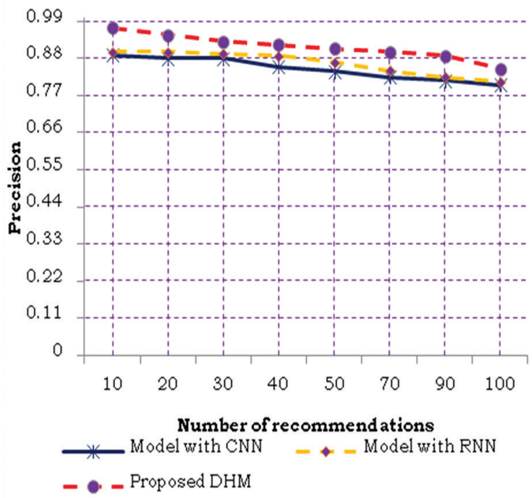
The testing data includes 6000 "feedback, grade, and learning resource" data from all learning resources, ensuring an average of 1000 samples per resource. In particular, the Stochastic Gradient Descent method and batch training are used during training. To address the imbalance in sample numbers, we guarantee that each batch has samples in each category throughout training. Accuracy, Precision, Recall, and F1 comparative data are in Table 1. The deep hybrid model performs best in Accuracy, Precision, Recall, and F1. The accuracy of the two baseline models across the above ten learning resources is above 0.96, whereas our suggested model averages 0.91. The suggested model achieves an average F1 value of 0.94, whereas the two baseline models among six learning resources have an F1 value above 0.92. This is because the deep hybrid model uses two max-pooling layers to highlight semantic and sequential features, which can better handle short input data lengths. In particular, our model performs better in "machine learning" and "data analysis" than in learning resources. Compared to other learning materials, learners in "machine learning" and "data analysis" may have a better understanding of their characteristics, which are easier to summarize.

The characteristics are complicated to convey in a few words. These findings suggest that providing intelligent assistance for diverse learners in online courses is challenging. Learners need time to improve their abilities. One group improves abilities throughout academics, while the other learns specialized skills for a short period as per projects, personal interests, and organizational desires. Learning goals are predicted over time using learner objectives, needs, and interests in this article. We are efficient in the first 10 recommendations, but as the number of ideas rises, the model's precision, accuracy, and recall diminish. CNN and RNN examine learner behavior over time. Based on this, each learner's group or cluster based on aims and interests might alter over time. To assess uniqueness, a 150-m dataset analysis was shown. CNN and RNN algorithms complete all tests 200 times.

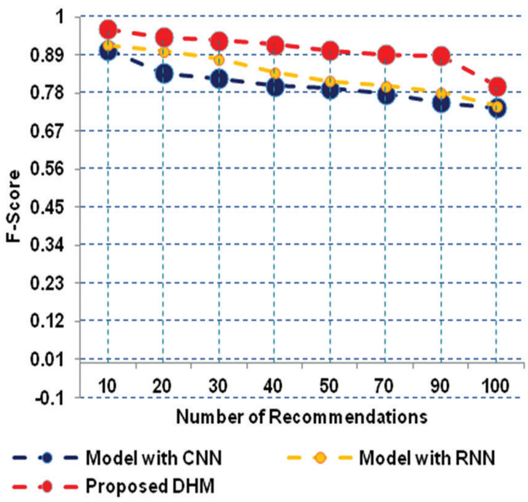
The DHM in Figures 3 (a) – 3 (d) and Table 2 underwent evaluation for precision, recall, F1-score, and accuracy. Compared to CNN and RNN models, the suggested DHM for different courses has the greatest precision, recall, accuracy, and F1 score. The first 10 recommendations based on DHM have an accuracy exceeding 0.964, but as the number of suggestions rises, all measures diminish. According to experiment findings, the suggested DHM outperforms CNN and RNN models in precision, recall, accuracy, and F1-score for different courses. According to Figures 3 (a) – 3 (d), the DHM system is extremely excellent in the top 10 recommendations.



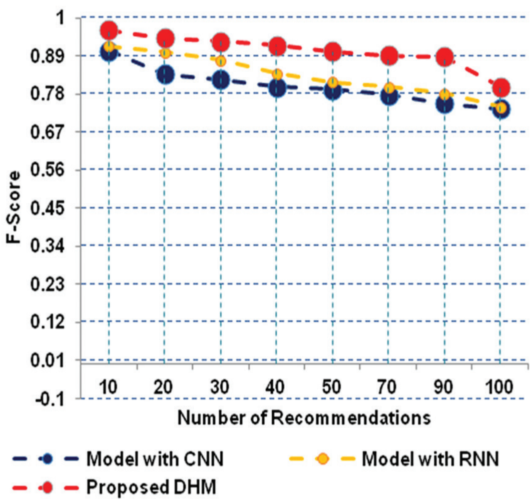
(a)



(b)



(c)



(d)

Fig. 1. (a) Accuracy (b) Precision (c) Recall and (d) F1 values of DHM compared with other approaches

5. CONCLUSION

This research introduces an integrated DHM framework with an intelligent recommendation system for online course assistance. CNN-RNN classification and clustering models modeled and analyzed course-resource-generated data. We created a novel CNN-RNN framework structure and devised a deep hybrid method to extract and highlight semantic and sequential information. The architecture for intelligent online course assistance includes a learning resource module that guides learners to the right course and a course module that predicts courses. A clustering model was created to optimize the learning process and reduce the course scope using more representative characteristics. A recommendation system was created to automatically offer learning materials and courses to learners. Experimental findings from online course data show that our suggested model and approach enhance classification accuracy and training efficiency, particularly for short input data. We want to address high-dimensional course-related data in future investigations. We will conduct evaluation tests to address challenging circumstances in online course learning data contexts.

6. REFERENCES

[1] S. Altaf, R. Asad, S. Ahmad, I. Ahmed, M. Abdollahian, M. Zaindin, “A Hybrid Framework of Deep Learning Techniques to Predict Online Performance of Learners during COVID-19 Pandemic”, Sustainability, Vol. 15, 2023, p. 11731.

[2] Z. Tan, L. He, “An efficient similarity measure for user-based collaborative filtering recommender systems inspired by the physical resonance principle”, IEEE Access, Vol. 5, 2017, pp. 27211-27228.

[3] Z. Liu, L. Wang, X. Li, S. Pang, “A multi-attribute personalized recommendation method for manufacturing service composition with combining collaborative filtering and genetic algorithm”, Journal of Manufacturing Systems, Vol. 58, 2021, pp. 348-364.

[4] L. Baltrunas, F. Ricci, “Experimental evaluation of context-dependent collaborative filtering using item splitting”, User Modeling and User-Adapted Interaction, Vol. 24, No. 1-2, 2014, pp. 7-34.

[5] Y. Ren, M. Tomko, F. D. Salim, J. Chan, C. L. A. Clarke, M. Sanderson, "A location-query-browse graph for a contextual recommendation," IEEE Transactions on Knowledge and Data Engineering, Vol. 30, No. 2, 2018, pp. 204-218.

[6] G. Adomavicius, A. Tuzhilin, R. Zheng, “REQUEST: A query language for customizing recommendations”, Information Systems Research, Vol. 22, No. 1, 2011, pp. 99-117.

[7] N. Joorabloo, M. Jalili, Y. Ren, “A probabilistic graph-based method to improve recommender system accuracy”, Proceedings of the International Conference on Engineering Applications of Neural Networks, Xersonisos, Greece, 24-26 May 2019, pp. 151-163.

[8] S. Ahmadian, N. Joorabloo, M. Jalili, M. Meghdadi, M. Afsharchi, Y. Ren, “A temporal clustering approach for social recommender systems”, Proceedings of the IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, Barcelona, Spain, 28-31 August 2018, pp. 1139-1144.

[9] O. C. Santos, J. G. Boticario, D. Pérez-Marín, “Extending web-based educational systems with personalized support through User Centred Designed recommendations along the e-learning life cycle”, Science of Computer Programming, Vol. 88, 2014, pp. 92-109.

[10] F. Garivaldis, S. McKenzie, D. Henriksen, S. Studente, "Achieving lasting education in the new digital learning world" Australasian Journal of Educational Technology, Vol. 38, No. 4, 2022.

[11] G. Adomavicius, A. Tuzhilin, “Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions”, IEEE Transactions on Knowledge and Data Engineering, Vol. 17, 2005, pp. 734-749.

[12] R. Burke, “Hybrid recommender systems: survey and experiments”, User Modeling and User-Adapted Interaction, Vol. 12, 2002, pp. 331-370.

[13] D. H. Park, H. K. Kim, I. Y. Choi, J. K. Kim, “A literature review and classification of recommender systems research”, Expert Systems with Applications, Vol. 39, 2012, pp. 10059-10072.

[14] Q. Zhang, J. Lu, G. Q. Zhang, “Recommender Systems in E-learning”, Journal of Smart Environments and Green Computing, Vol. 1, 2021, pp. 76-89.

[15] L. Jiang, Y. Cheng, L. Yang, J. Li, H. Yan, X. Wang, “A trust-based collaborative filtering algorithm for E-commerce recommendation system”, Journal of Ambient Intelligence and Humanized Computing, Vol. 10, No. 8, 2019, pp. 3023-3034.

[16] L. Tang, Z. Duan, Y. Zhao, “Toward using social media to support ridesharing services: challenges and opportunities”, Transportation Planning and Technology, Vol. 42, No. 4, 2019, pp. 355-379.

[17] F. Y. Liu, X. Q. Gao, Z. Zhang, “Improved Bayesian probabilistic model based recommender system”, Computer Science, Vol. 44, No. 5, 2017, pp. 285-289.

[18] S. Yang, M. Korayem, K. AlJadda, T. Grainger, S. Natarajan, “Combining content-based and collaborative filtering for job recommendation system: A cost-sensitive Statistical Relational Learning approach”, Knowledge-Based Systems, Vol. 136, 2017, pp. 37-45.

[19] C. Xia, X. Jiang, S. Liu, Z. Luo, Z. Yu, “Dynamic item-based recommendation algorithm with time decay”, Proceedings of the Sixth International Conference on Natural Computation, Yantai, 10-12 August 2010, pp. 242-247.

[20] M. Deshpande, G. Karypis, “Item-Based Top-N Recommendation Algorithms”, ACM Transactions on Information Systems, Vol. 22, No. 1, 2004, pp. 143-177.

[21] B. Sarwar, G. Karypis, J. Konstan, J. Riedl. “Item-Based Collaborative Filtering Recommendation Algorithms”, Proceedings of the 10th international conference on World Wide Web, May 2001, pp. 285-295.

[22] Y. Ma, S. Ji, Y. Liang, J. Zhao, and Y. Cui, “A hybrid recommendation list aggregation algorithm for group recommendation”, Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, Singapore, 6-9 December 2015, pp. 405-408.

[23] S. Parvatikar, B. Joshi, "Online book recommendation system by using collaborative filtering and association mining", Proceeding of the IEEE International Conference on Computational Intelligence and Computing Research, 10-12 December 2015, pp. 1-4.

[24] A. Al-Badarenah, J. Alsakran, "An Automated Recommender System for Course Selection", International Journal of Advanced Computer Science and Applications, Vol. 7, No. 3, 2016.

[25] T. T. Dien, N. Thanh-Hai, N. T. Nghe, “An approach for learning resource recommendation using

deep matrix factorization", Journal of Information and Telecommunication, Vol. 6, No. 4, 2022, pp. 381-398.

[26] Q. Li, J. Kim, "A Deep Learning-Based Course Recommender System for Sustainable Development in Education", Applied Sciences, Vol. 11, 2021, p. 8993.

[27] Y. Biletskiy, H. Baghi, I. Keleberda, M. Fleming, "An adjustable personalization of search and delivery of learning objects to learners", Expert Systems with Applications, Vol. 36, 2009, pp. 9113-9120.

[28] N. Capuano, M. Gaeta, P. Ritrovato, S. Salerno, "Elicitation of latent learning needs through learning goals recommendation", Computers in Human Behavior, Vol. 30, 2014, pp. 663-673.

[29] S. Yanga, M. Korayemb, K. AlJaddab, T. Grainger, S. Natarajan "Combining content-based and collaborative filtering for job recommendation system: A cost-sensitive Statistical Relational Learning approach", Knowledge-Based Systems, Vol. 136, 2017, pp. 37-45.

[30] P. Resnick, H. R. Varian, "Recommender systems – introduction to the special section", Communications of the ACM, Vol. 40, No. 3, 1997, pp. 56-58.

[31] J. B. Li, S. Y. Lin, Y. H. Hsu, Y. C. Huang, "An empirical study of alternating least squares collaborative filtering recommendation for movie lens on Apache Hadoop and spark," International Journal of Grid and Utility Computing, Vol. 11, No. 5, 2020, pp. 674-682.

[32] C. Porcel, A. Tejeda-Lorente, M. A. Martínez, E. Herrera-Viedma, "A hybrid recommender system for the selective dissemination of research resources in a technology transfer", Information Sciences, Vol. 184, 2012, No. 1, pp. 1-19.