Bulletin of Electrical Engineering and Informatics
Vol. 15, No. 6, December 2026, pp. 5315~5327
ISSN: 2302-9285, DOI: 10.11591/eei.v15i6.11936

Deep learning driven personalized learning: CNN, DNN, and Keras Tuner for activity recommendation

Bensalem Assri[1], Amine Mrhari[2], Moulay Youssef Hadi[1], Mohamed Boutaib[1]
[1]Scientific Research and Innovation Laboratory, Higher School of Technology, Ibn Tofail University, Kenitra, Morocco
[2]The National Institute of Planning and Urbanism (INAU), Rabat, Morocco

Article Info

Article history:

Received Nov 23, 2025
Revised Sep 19, 2026
Accepted Oct 4, 2026

Keywords:

Convolutional neural network
Deep neural network
Felder-Silverman learning style model
Open university learning analytics dataset
Recommendation
Student outcome
Virtual learning environment

ABSTRACT

Personalized learning for individual needs is one of the most important research areas in e-learning, aiming primarily to improve students' outcomes. This paper proposes a novel framework for personalized learning based on students' learning styles (LS) and behavior data. The prediction of student performance is crucial to identify students who need personalization interventions. Identifying LS is a major step to provide each student with a pedagogical experience tailored to their needs. The LS is identified based on the Felder-Silverman learning style model (FSLSM) and student interaction with virtual learning environments (VLE). A learning activity recommendation is used to recommend the most suitable activities for each student. Open university learning analytics dataset (OULAD) is used, which contains various data about students' interaction with VLE. We employed a methodological core combining convolutional neural network (CNN) and deep neural network (DNN) models using Keras Tuner for hyperparameter tuning to predict student performance and identify learners' LS. The results indicate that DNN and CNN achieved improved outcomes for 51.37% and 40.82% of students, respectively, after the recommendations were made. This study demonstrates the effectiveness of deep learning (DL) with hyperparameter tuning for personalized learning in distance learning.

This is an open access article under the CC BY-SA license.



Corresponding Author:

Bensalem Assri
Scientific Research and Innovation Laboratory, Higher School of Technology, Ibn Tofail University
Kenitra, Morocco
Email: bensalem.assri@uit.ac.ma

1. INTRODUCTION

E-learning has evolved significantly over the last years [1]. Several studies indicated a deterioration in the performance of most students during the COVID-19 period [2], [3]. Virtual learning environments (VLE) permit teachers to connect with their students, communicate, and engage with one another. VLEs allow students access to learning resources and requesting help from their professors [4]. Personalized learning allows students to work on their unique VLE material at their own pace. Therefore, it's critical to use individualized learning to consider each student's unique characteristics [5] to improve their outcomes. The identification of students' learning styles (LS) is a major step to adapt the content and learning techniques according to their preferred way of learning. The concept of LS means that people have different preferences for how they learn best and denotes that individuals acquire knowledge through diverse modalities. Many LS models have been suggested in literature. Felder-Silverman learning style model (FSLSM) is widely accepted in education. Numerous strategies are suggested to identify students' LSs; the traditional one is a survey in which several questions are asked, and then the results indicate the dominant LS. This solution has notable disadvantages declared by Aissaoui *et al.* [6]. The second is to automatically

detect the LS based on machine learning (ML) techniques. Most proposed works aimed to identify one LS for each student. However, students can have multiple LSs. Identifying just one LS is ineffective for those with mixed or no dominant LS [7], as supported by Azzi *et al.* [8], who showed that various LSs exist. Clark *et al.* [9] suggest that many students switch LS depending on whether they are learning English or mathematics. Therefore, it appears to support the existence of dynamics in LS.

Many researchers have focused on personalized learning approaches. For instance, Nazempour and Darabi [10] investigated students' LS to enhance learning outcomes. They analyzed behavior data based on students' interactions with the learning management system (LMS). The FSLSM maps students' interactions with online learning resources to LS features. They utilized ML training and LS selection to identify the specific LS features that optimize students' academic performance. Azzi *et al.* [8] propose learning objects (LO) automatically in intelligent e-learning systems. The study prepared data to serve as input for the fuzzy C-Means (FCM) clustering algorithm. The FCM maps each LO to an LS based on FSLSM. Research by Hasibuan *et al.* [11] classify students' LS using FSLSM and ML algorithms based on information gathered from questionnaires. This classification improves lesson planning for instructors and helps students decide what course materials to use. The results showed that motivation and satisfaction were better, rising from 60% to 87% when LSs were considered. Xiao *et al.* [12] introduced a new cross-course learning path planning model that combines resources from different courses through special matching functions based on a knowledge graph and item response theory. The improved differential evolution algorithm optimizes learning personalization. Results indicated that cross-course paths improve learning outcomes. Nafea *et al.* [13] propose a new recommender framework that combines student ratings with their LS for personalized course recommendations. FSLSM is utilized to detect LS and LO profiles. The model investigated 80 students for an online course created in the LMS.

Research by Amin *et al.* [14], apply a multi-agent approach with deep reinforcement learning (DRL) to personalize learning paths. Research considers factors like student sentiments, LS, and abilities to provide customized course suggestions. The researchers built intelligent recommender systems using a DRL-based actor-critic model called DRR to treat recommendations as a sequential decision-making process. The result shows the DRR model's effectiveness in making long-term recommendations. Shahbazi and Byun [15] suggest a system using virtual agents to recommend suitable content based on users' profiles and preferences. It employs semantic analysis and natural language processing (NLP) to recommend course selection for students and tutors. The system learns the user characteristics and processes the LS through the clustering technique, achieving a prediction accuracy of 98%. Singh *et al.* [16] present a personalized intelligent tutoring system named SeisTutor, which analyzes issues and recommends a customized curriculum. SeisTutor actively assesses the student's behavior and psychological state to enhance their learning experience. Results indicated that participants who used SeisTutor with customized learning saw a 44.34% improvement compared to others. Kaiss *et al.* [17] involve NLP and ML to introduce an adaptive learning approach based on students' LS, using a chatbot called LearningPartnerBot integrated with the Moo NIDS e platform. This research employs the FSLSM to assess LS and evaluates the LearningPartnerBot with 71 students, providing them with real-time answers and tailored suggestions. Results showed an improvement on learning outcomes. Mustapha *et al.* [18] select the most effective learning materials for students according to their learning preferences and motivation score. They employed restricted Pearson correlation coefficient, K-nearest neighbor algorithm (KNN), modified cosine measure, and collaborative filtering approach. The results indicate that a learner's motivation score and LS increase the quality of their education.

Osman [19] studies the application of deep learning (DL) and ML techniques to provide customized learning experiences in adaptive e-learning systems. A dataset of 1,210 students was used to estimate each student's LS. Random forest (RF), extreme gradient boosting (XGBoost), adaptive boosting (AdaBoost), logistic regression (LR), and deep neural networks (DNN) are tested, with DNN demonstrating the best LS prediction. Based on students' learning preferences, El-Sabagh [20] creates an adaptable online learning environment and examines its impact on student engagement. A quasi-experimental design was used for comparing the adaptive technique with a conventional e-learning system. The engagement's emotional and behavioral components were evaluated. The results showed that the experimental group had higher engagement. Tapalova and Zhiyenbayeva [21] discuss how artificial intelligence (AI) in education (AIEd) forms a personalized educational platform. It provides a systematic framework for the integration of technologies such as VLE, chatbots, expert systems, intelligent agents, and ML. Results indicated better learning effectiveness. Bousalem *et al.* [22] proposed a framework that integrates DL, learner profile ontology, and Myers–Briggs type indicator (MBTI) personality prediction. This method classifies personalities by bidirectional encoder representations from transformers (BERT) model. Predicted personality types are integrated into a semantic web rule language (SWRL) ontology with WordNet to improve resource suggestions. The results demonstrate a marked improvement in educational achievements. Using the open university learning analytics dataset (OULAD), Almuqhim and Berri [23] suggested that

higher education should provide AI-based personalized microlearning e-courses. They assess AI algorithms to generate adaptive learning pathways for individual students. AI algorithms dynamically learn, predict, and respond to individual learning needs. The results achieved were an F1-score of 99% and a precision of 99%.

Most personalized learning studies rely on surveys or conventional ML techniques. This study investigates the effectiveness of DL algorithms for personalizing learning; it is among the limited studies that employ the hyperparameter tuning with convolutional neural network (CNN) and DNN architectures. Furthermore, to establish a more robust and generalizable classification of learners, this research considers all accessible courses in the OULAD, unlike many works that focus on a particular course. The main goal of this study is to evaluate CNN and DNN as DL models with Keras Tuner as a hyperparameter tuning for recommending activities. Most works on OULAD have focused on traditional ML models like DT, RF, LR, KNN, and SVM [10], [24], [25]. Although interpretable, these approaches often lack the ability to model the complex non-linear interactions between student behaviors, course content, and assessment performance. These algorithms require static, tabular inputs and manual feature engineering. Thus, they are unable to model sequential dependencies, long-range temporal correlations, and complex clickstream behaviors. The few studies that have implemented DL have been mostly performed without Keras Tuner [26], [27]. Others focused on one or a limited set of courses, raising questions about the generalizability of the findings [23]. To fill these gaps, our study proposes a new methodological framework combining DL with the widely used LS model in educational domain called FSLSM [28]. We are also validating this integrated approach across all courses in OULAD. We tuned our DL architectures systematically for best performance and reproducibility using Keras Tuner with Bayesian optimization, a rarely adopted practice in educational data mining research. This study proposes a hybrid framework that combines CNN, DNN, Bayesian optimization, and LS classification. CNNs learn local temporal motifs in VLE clickstreams without manual feature engineering; DNNs combine these features into high-dimensional representations of complex interactions. We apply Bayesian optimization to efficiently tune the non-convex parameter space of the model, avoiding over-fitting to sparse data. Finally, the inductive bias of LS classification constrains the predictions by leveraging the known pedagogical theories to improve the generalizability and actionable insight.

The main novelty of our work is this combination of DL+FSLSM+full-course validation+automated tuning, which distinguishes it from previous studies and provides a robust and generalizable solution for generating recommendations. We emphasize that our contribution does not merely consist of combining existing components but rather in demonstrating, through quantitative metrics, that their synergistic integration yields statistically significant gains. Specifically, we quantify our advances along three axes: i) predictive accuracy, measured by F1-score improvement over baselines (9%); ii) LS classification, assessed by recall gains (3%); and iii) recommendation effectiveness, which gives us a 40.82 to 51.37% improvement in student outcomes after recommendation. Additionally, automated hyperparameter tuning contributes not only to accuracy but also to computational efficiency, reducing training time over 50% by avoiding excessive testing.

This work answers the research questions: i) how can students in need of personalization be effectively identified? ii) in what ways can personalized learning be designed based on LS? iii) how can deriving LS from successful students improve learning outcomes? and iv) how can suitable activities be recommended for each student, and how might such recommendations contribute to improving student outcomes?

The remaining sections in this paper are organized as follows: section 2 proposes our methodology. Section 3 discusses the results. The last section summarizes the study's findings and identifies areas for future research.

2. METHOD

This study aims to improve students' outcomes through personalized learning. The study's workflow, proposed in Figure 1, starts with a data preprocessing technique, followed by hyperparameter tuning to identify the best hyperparameters. Subsequently, an initial prediction of students' outcome is employed. After that, the identification of students' LS is introduced. Next, the learning activity recommendation is executed based on students' LS and behavioural data. Finally, a second prediction of a student's outcome after the learning activity recommendation is applied. The approach's effectiveness is confirmed by comparing the initial prediction with the post-recommendation prediction.

2.1. Datapreprocessing

OULAD has large-scale data on course enrollments, student demographics, learning outcomes, assessment scores and VLE interactions. It is freely accessible and includes a wide range of information about 32,593 registered learners in 22 courses. OULAD is provided as a collection of seven tables named Courses, StudentInfo, Assessments, StudentAssessment, Vle, StudentVle, and StudentRegistration, which can be linked using unique IDs. Data preprocessing involves several steps to ensure the OULAD quality.

First, data cleaning addresses outliers [29] and missing values [30]. Table 1 demonstrates some examples and how they are treated. Second, feature extraction selects the relevant features. Third, feature scaling normalizes numerical variables. Finally, one-hot encoding converts text attributes to numerical formats. Figure 2 shows the workflow of data preprocessing.

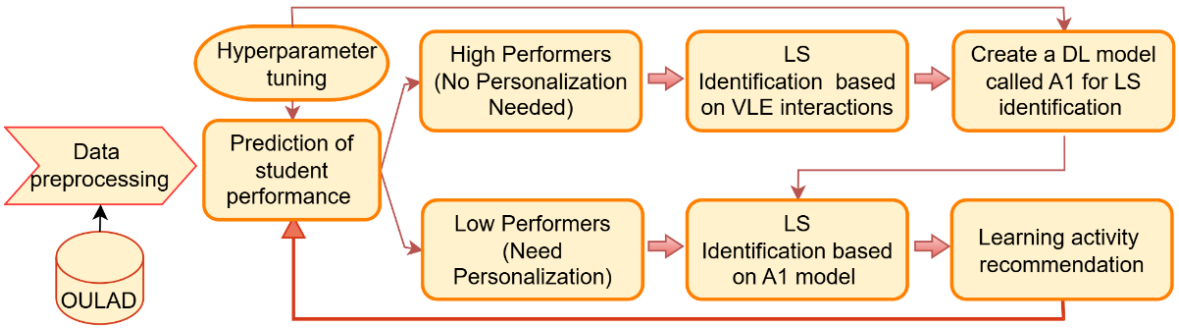


Figure 1. Workflow of the proposed method, from data preprocessing to recommendation output and final prediction

Table 1. Examples of data quality issues and their treatment

	Problem	Action
Outliers	In assessments, exams must have a weight of 100%; in the module "CCC," the weight of the exam is 200%	We changed to 100%
	In assessment: in module GGG, Sum(weight (CMA and TMA))=0. Except for GGG, 28 lines have sum(weight(CMA))=0	Since CMA often has a weight of 0, we assign a total weight of 100 to TMA.
	in studreg: some students who unregistered and have "Fail" in the value of "final result"	The value of "final_result" in these records is changed into "Withdrawn"
Missing value	In the assessments, there are 11 missing values for the exam date; since our prediction is made before the exam, the date does not have any impact.	We ignored the missing values associated with her
	In studentASS, we found 173 missing values for the scores. If the student does not submit the assessment, no result is recorded, then NAN is between 0 and 40	We decided to fix it to 20
	In studinfo, we found 1111 missing values for "imdband"	We replaced them with mode values

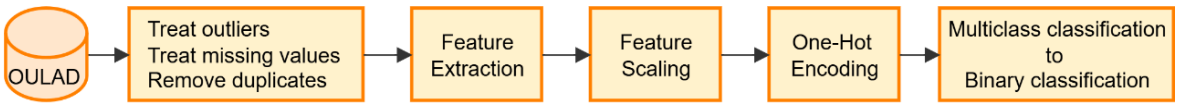


Figure 2. Stepwise data preprocessing workflow from OULAD to model-ready input sequences

After preprocessing these features are selected: gender, highest_education, age_band, disability, imd_band, dataplus, dualpane, externalquiz, folder, forumng, glossary, homepage, subpage, url, htmlactivity, oucollaborate, oucontent, ouelluminate, ouwiki, page, questionnaire, quiz, resource, sharedsubpage, repeatactivity, num_of_prev_attempts, and studied_credits. The target variable Final_result in outcome prediction has four classes: withdrawn, pass, fail, and distinction. To enable the training of ML models, the multiclass classification task was simplified to a binary classification. The classes withdrawn and fail were merged into a fail class as both indicate the student failed the course. However, pass and distinction classes were merged into a pass class, since both represent students who have successfully passed the course and can be used as reference cases for the identification of patterns related to successful learning. Thus, the label Distinction was changed to pass and withdrawn to fail. We used the same features in predicting academic performance and identifying LS because VLE interaction features are indicators of students' behavioral and engagement patterns with different types of learning activities and therefore relevant for both performance prediction and LS identification. Additional context about the learner is gained from demographic and academic-related features. Thus, the same student representation is used in both stages, but the objectives are different: predicting academic performance and identifying LS to facilitate further activity recommendation. All repeated interactions of each student were aggregated into one record before the data were split. The

resulting data set has one unique student per row. This data set was then split into training, validation, and testing sets, using stratified splitting to preserve the class distribution. Thus, each student appears in only one subset, avoiding data leakage and reducing the possibility of overfitting.

2.2. Hyperparameters tuning

Keras Tuner is a framework for hyperparameter optimization that was created to solve typical problems with ML hyperparameter search. It enables researchers to dynamically define the search space, promoting adaptable and repeatable experimentation [31]. To avoid overfitting, we are using stratified cross-validation with cv=5 and early stopping with patience=5 for both CNN and DNN models. Table 2 presents the different hyperparameters for DNN and CNN models, and Figure 3 shows the hyperparameter tuning's workflow.

Table 2. Hyperparameters for DNN and CNN models

Model	Hyperparameters	Description
CNN with number of trials=50	conv_filters (32 to 128, steps=32)	Number of filters in the first convolutional layer
	conv_kernel ([3, 5])	Kernel size for the first convolutional layer
	dense_units_X (64 to 512, steps=64)	Number of neurons in each dense layer
	dense_activation_X (['relu', 'tanh'])	Activation function for dense layers
	dropout_X (0.1 to 0.5, steps=0.1)	Dropout rate for each dense layer
	lr (1e-4 to 1e-2, log sampling)	Learning rate for Adam optimizer
DNN with number of trials=20	units_i (32 to 512, steps=32)	Number of neurons in each dense layer (where i is the layer index)
	activation_i (['relu', 'tanh'])	Activation function for each dense layer
	dropout_i (0.1 to 0.5, steps=0.1)	Dropout rate for each dense layer
	lr ([1e-4, 1e-3, 1e-2])	Learning rate for Adam optimizer

All experiments were performed on a computer with an 11th Gen Intel(R) Core(TM) i7-1165G7 processor at 2.80 GHz (8 CPUs) and 8 GB of RAM, running the Windows 11 64-bit operating system. Model development and training were carried out using Google Colab, which gave approximately 12.7 GB of RAM and 107.7 GB of disk space. All experiments were performed using a fixed random seed of 42 for reproducibility. The dataset was split into training (64%), validation (16%), and testing sets (20%) with the class distribution in all subsets preserved.

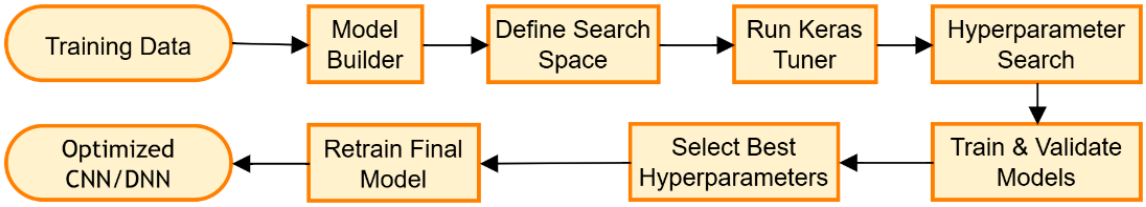


Figure 3. Workflow of the automated hyperparameter tuning using Bayesian search for CNN–DNN architecture

2.3. Prediction of student performance

In personalized learning, the prediction of student performance is the first step to identify students who need tailored instructional support. This prediction is done before and after the learning activity recommendation stage to measure its effectiveness; it is considered performed if the number of students in category "Pass" has increased. DL is investigated here using CNN and DNN models. Figure 4 shows the student performance prediction's workflow.

2.4. Learning style identification

The identification of LS is crucial for recommending learning activities based on the individual needs of each student. In this study, the attribution of LS is determined using the classification reported in [10], as summarized in Table 3. Figure 5 illustrates the process for assigning the LS based on student interactions with the VLE.

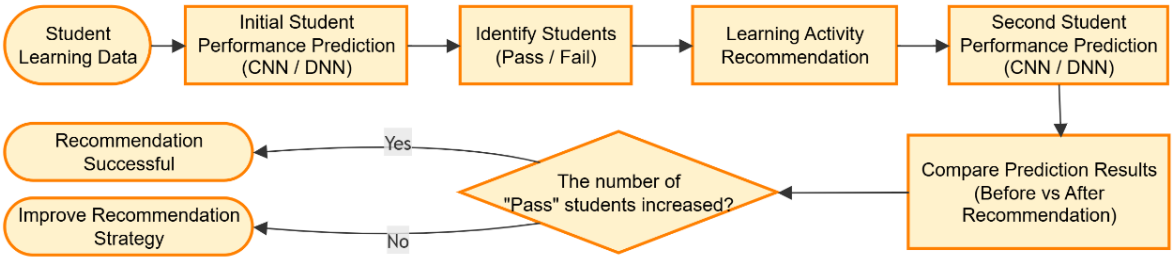


Figure 4. Prediction workflow for student performance using the proposed CNN–DNN hybrid model

Table 3. Associating different activity types with LS characteristics according to FSLSM

Dimension	Preference	Activities
Processing style	Active	Forumng, oucollaborate, and ouwiki
	Reflective	Glossary and htmlactivity
Perception style	Sensitive	Oucontent and questionnaire
	Intuitive	Quiz and externalquiz
Input style	Visual	Dataplus, dualpane, folder, page, and homepage
	Verbal	Resource, url, ouelluminate, and subpage
Understanding style	Sequential	Repeatactivity
	Global	Sharedsubpage

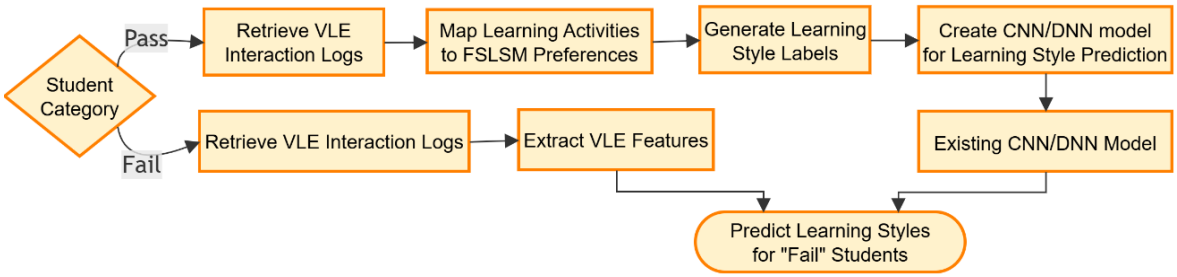


Figure 5. Learning-style classification workflow for student profiling using the proposed hybrid model

2.5. Model architecture

The architecture of CNN and DNN models is based on results from tuning hyperparameters using Keras tuner, which chooses the best parameters, such as the number of layers, activation function, dropout rate, and optimizer settings. Tables 4 and 5 illustrate detailed architecture for both CNN and DNN models in student performance prediction and LS identification, respectively. Although CNNs are originally designed for spatial data, recent works have shown that they can be efficiently applied to high-dimensional tabular data by learning local patterns among neighboring features after proper feature representation. In this work, CNN was chosen because of its ability to automatically learn discriminative feature representations through convolutional layers, which capture local interactions between related variables. This is especially relevant for educational interaction data where groups of behavioral features may show correlated patterns. In addition, the use of weight sharing and local connectivity reduces the number of trainable parameters compared to DNN, leading to higher computational efficiency and fewer chances of overfitting. CNNs are very suitable for extracting informative patterns to improve prediction accuracy, given the size and complexity of OULAD.

2.6. Learning activity recommendation

The learning activity recommendation process started with the students in the "Fail" category identified in the performance prediction stage and their LS identified in the LS identification stage. Then the system retrieves the specific FSLSM dimensional preferences assigned to each student. Next, match the student's preferences and the mapped educational tasks, as mentioned in Table 3. Finally, aggregate a tailored set of learning activities to provide final recommendations. Figure 6 shows the workflow for learning activity recommendation.

Table 4. CNN and DNN architectures for student performance prediction

Model	Layer type	Element number	Activation function	Kernel regularizer	Optimizer	Learning rate	Epoch	Batch size
DNN with 5 hidden layers	Dense with Dropout(0.4)	64	ReLU	l2(0.01)	Adam	0.0001	100	64
	Dense with Dropout(0.2)	416	ReLU					
	Dense with Dropout(0.5)	32	tanh					
	Dense with Dropout(0.1)	256	ReLU					
	Dense with Dropout(0.1)	288	tanh					
	Dense output	1	sigmoid					
CNN with 2 conv layers and 2 dense layers	Conv layer 1 with kernel_size=3	96		l2(0.01)	Adam	0.0017	100	128
	Dense layer 1 with dropout_rate=0.4	128	ReLU					
	Dense layer 2 with dropout_rate=0.1	256	tanh					
	Conv layer 2 with kernel_size=3	192						
	Dense layer 3 with dropout_rate=0.3	320	tanh					
	Output dense layer	1	sigmoid					

Table 5. CNN and DNN architectures for LS identification

Model	Layer type	Element Number	Activation function	Kernel regularizer	Optimizer	Learning rate	Epoch	Batch size
DNN with 4 hidden layers	Dense with Dropout(0.5)	64	ReLU	0.001	Adam	0.001	100	64
	Dense with Dropout(0.1)	352	tanh					
	Dense with Dropout(0.1)	128	ReLU					
	Dense with Dropout(0.4)	192	tanh					
	Dense output	1	sigmoid					
CNN with 2 conv layers and 3 dense layers	Conv layer 1 with kernel_size=3	96		0.001	Adam	0.001	100	64
	Dense layer 1 with dropout_rate=0.4	128	ReLU	0.0				
	Dense layer 2 with dropout_rate=0.1	256	tanh	0.0				
	Conv layer 2 with kernel_size=3	192		0.001				
	Dense layer 3 with dropout_rate=0.3	320	tanh	0.0				
	Output dense layer	1	sigmoid	0.0				

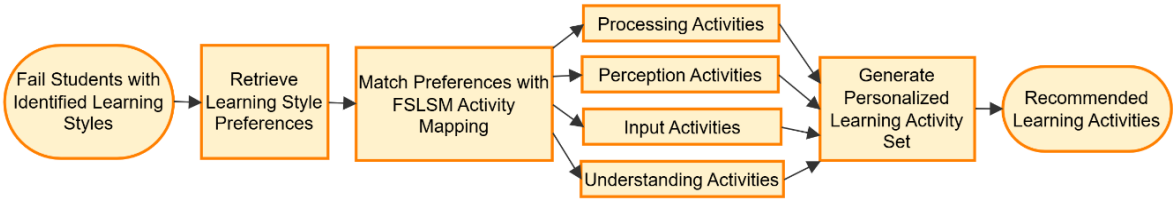


Figure 6. Recommendation workflow for personalized learning activities based on learning-style profiles

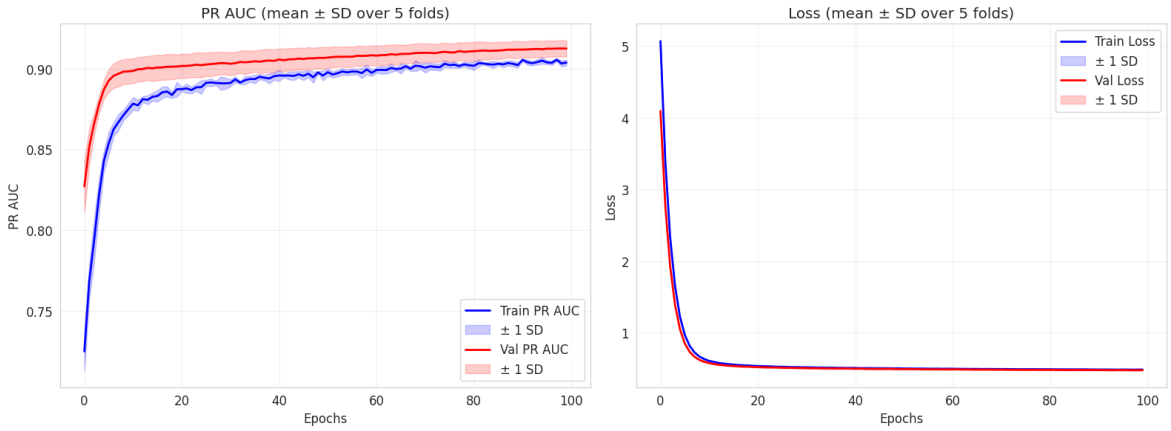
3. RESULTS AND DISCUSSION

3.1. Prediction of students' performance before learning activity recommendation

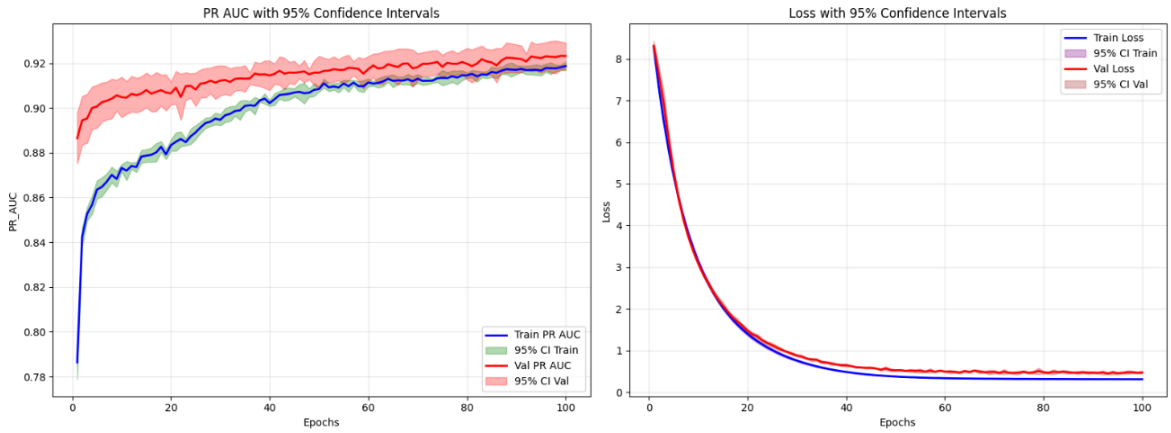
Table 6 shows the metric values for both CNN and DNN models. The findings indicate that CNN outperforms DNN in predicting students' performance. Its ability to capture spatial relationships and complex interaction patterns. Figure 7 presents the training and validation plots for PR_AUC and losses for these models, with Figures 7(a) and (b) concerning the DNN and CNN models, respectively. These plots indicate that both models perform consistently and converge efficiently. The developed model was retained to assess improvements in students' outcomes following the implementation of the proposed activity recommendation.

Table 6. Metrics values for CNN and DNN models in student performance prediction

Model	Accuracy	Precision	Recall	F1-score	ROC_AUC	PR_AUC
CNN	0.842	0.833	0.959	0.892	0.885	0.928
DNN	0.820	0.816	0.949	0.877	0.865	0.915



(a)



(b)

Figure 7. Training and validation curves of; (a) DNN and (b) CNN models, evaluated using PR-AUC and loss

3.2. Identification of students' learning styles

Table 7 shows the metrics values for both CNN and DNN models. The findings demonstrate that CNN outperforms again the DNN in terms of generalization and prediction stability, most likely because of its convolutional layers, which can efficiently capture structural patterns in the students' interaction data. Figure 8 presents the plot of the ROC curves for both models, with Figures 8(a) and (b) concerning DNN and CNN models, respectively. The plot shows exceptional classification ability reflected by a macro-average AUC of 0.99 for both models.

Table 7. Metrics values for CNN and DNN models in LS identification

Model	Accuracy	Precision	Recall	F1	ROC_AUC
CNN	0.867	0.977	0.973	0.969	0.953
DNN	0.846	0.973	0.969	0.963	0.883

3.3. Learning activity recommendation

Table 8 shows the results of the learning activity recommendation concerning both CNN and DNN models for each student in the category "Fail". Both models provide a dominant sequence: 71.53% for the DNN and 37.04% for the CNN, suggesting significant convergence in recommendations. DNN's distribution is more concentrated around a few core sequences, implying more stability in its recommended patterns. In comparison, CNN produces a wider set of sequences in categories ranging from 0.02% to 31.92%.

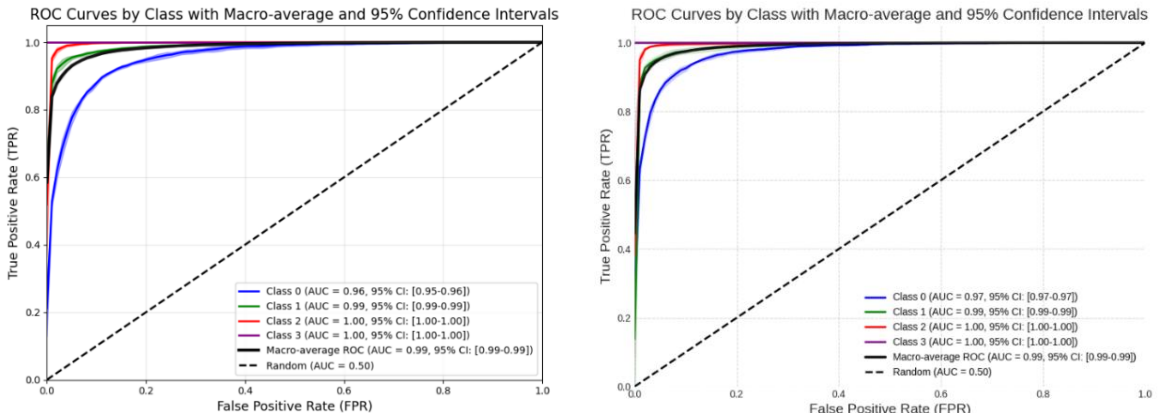


Figure 8. Macro-averaged ROC curves for; (a) DNN and (b) CNN models

Table 8. Distribution of students by generated recommendation sequence

Model	Recommendation	Percentage (%)
DNN	glossary, htmlactivity, oucontent, questionnaire, dataplus, dualpane, folder, page, homepage, repeatactivity	71.53
	glossary, htmlactivity, oucontent, questionnaire, resource, url, ouelluminate, subpage, repeatactivity	14.91
	glossary, htmlactivity, quiz, externalquiz, dataplus, dualpane, folder, page, homepage, repeatactivity	10.77
	forumng, oucollaborate, ouwiki, oucontent, questionnaire, dataplus, dualpane, folder, page, homepage, repeatactivity	1.05
	glossary, htmlactivity, quiz, externalquiz, resource, url, ouelluminate, subpage, repeatactivity	0.85
	forumng, oucollaborate, ouwiki, oucontent, questionnaire, resource, url, ouelluminate, subpage, repeatactivity	0.81
	glossary, htmlactivity, quiz, externalquiz, resource, url, ouelluminate, subpage, sharedsubpage	0.03
	forumng, oucollaborate, ouwiki, quiz, externalquiz, dataplus, dualpane, folder, page, homepage, repeatactivity	0.03
	glossary, htmlactivity, oucontent, questionnaire, resource, url, ouelluminate, subpage, sharedsubpage	0.01
CNN	glossary, htmlactivity, oucontent, questionnaire, resource, url, ouelluminate, subpage, repeatactivity	37.04
	glossary, htmlactivity, quiz, externalquiz, resource, url, ouelluminate, subpage, repeatactivity	31.92
	glossary, htmlactivity, oucontent, questionnaire, dataplus, dualpane, folder, page, homepage, repeatactivity	14.27
	glossary, htmlactivity, quiz, externalquiz, dataplus, dualpane, folder, page, homepage, repeatactivity	6.80
	forumng, oucollaborate, ouwiki, oucontent, questionnaire, resource, url, ouelluminate, subpage, repeatactivity	5.78
	forumng, oucollaborate, ouwiki, oucontent, questionnaire, dataplus, dualpane, folder, page, homepage, repeatactivity	2.21
	forumng, oucollaborate, ouwiki, quiz, externalquiz, resource, url, ouelluminate, subpage, repeatactivity	0.85
	forumng, oucollaborate, ouwiki, quiz, externalquiz, dataplus, dualpane, folder, page, homepage, repeatactivity	0.72
	glossary, htmlactivity, quiz, externalquiz, resource, url, ouelluminate, subpage, sharedsubpage	0.31
	glossary, htmlactivity, oucontent, questionnaire, resource, url, ouelluminate, subpage, sharedsubpage	0.08
	forumng, oucollaborate, ouwiki, oucontent, questionnaire, resource, url, ouelluminate, subpage, sharedsubpage	0.02

3.4. Students' performance prediction after the learning activity recommendation

Following the recommendation process, the recommended activities retained their "sum_click" values, but the non-recommended ones were replaced by zeros, meaning that the student did not interact with them. To measure our approach's effectiveness, the students' outcome prediction is made using the models previously trained before the recommendation step. Table 9 presents the students' distribution in the "Pass" and "Fail" categories before and after the recommendation stage. The model is applied to students in the "Fail" category, yielding 3190 (51.37%) and 2535 (40.82%) in the "Pass" category for the DNN and CNN models, respectively. The findings show that our method works well in improving student performance.

Table 9. Students' distribution in "Pass" and "Fail" categories before and after activity recommendation

Model	Recommendation stage	Number of students in pass category	Number of students in fail category	Improvement (%)
DNN	Before recommendation	0	6209	
	After recommendation	3190	3019	51.37
CNN	Before recommendation	0	6209	
	After recommendation	2535	3674	40.82

The results indicate that CNN is superior to DNN for predicting students' performance and LS identification. The ability of the CNN model to capture spatial relationships and complex interaction patterns is important. The results of predicting students' performance and identifying LS indicated also, that CNN and DNN had better performance than traditional ML algorithms, as shown in Tables 6, 7, and 10. CNN achieves the highest accuracy (0.842), precision (0.833), and F1-score (0.892) in predicting student performance. DNN also performs equally well with an accuracy of 0.820, and F1-score of 0.877, like LR (0.818 and 0.877). For LS identification, CNN is the best for overall performance with the highest accuracy (0.867), precision (0.977), recall (0.973), and F1-score (0.969), better than both DNN and all traditional algorithms. The results indicate that CNN is more effective in capturing the complex patterns in the learner behavioral data for both prediction tasks.

The proposed recommendation framework can be embedded in LMS to provide individualized educational support to students at risk of failing. The system can automatically analyze students' learning behaviors and styles and, after a predefined period, recommend the best learning activities. Students can be nudged to do the things that will improve their grades via notifications or personalized messages within a real LMS. The integration of recommendation results in teacher dashboards, enabling instructors to monitor at-risk students, review recommended activities, and provide timely pedagogical interventions. Our approach can help adaptive curricula by tailoring learning pathways to align with student LSs, behaviors, and progress. This study concerns data from OULAD and can't generalize directly to other institutions and is based on the FSLSM model; changing the model yields different insight. Additionally, all these activity recommendations in this study were simulated and not tested in real classrooms.

About the socio-technical aspect, the successful adoption of the proposed recommendation system depends on the acceptance and trust of the students in the recommendations provided. For ethical considerations, this study used a publicly available and anonymized dataset, minimizing privacy risks in the research phase. Only features that were directly related to the prediction of student performance were analyzed. For the real-world deployment in VLE, additional safeguards would be needed, including data encryption, role-based access control, and well-defined data retention policies to ensure compliance with privacy regulations and the ethical use of students' behavioral data.

Table 10. Metrics values for traditional ML algorithms in student performance prediction and LS identification

Model	For student performance prediction				For LS identification			
	Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score
KNN	0.804	0.795	0.962	0.871	0.697	0.918	0.950	0.914
SVM	0.820	0.821	0.820	0.808	0.763	0.965	0.941	0.940
LR	0.818	0.813	0.953	0.877	0.839	0.972	0.971	0.964
DT	0.810	0.807	0.949	0.873	0.589	0.857	0.966	0.888

4. CONCLUSION

This study examined the effectiveness of DL and OULAD for personalizing learning in VLE. DNN and CNN were used to identify the learners who need personalization, and accuracy was 0.821 and 0.842, respectively. They were also used to identify LS based on FSLSM, achieving accuracies of 0.846 and 0.867, respectively. The finding demonstrates that CNN outperformed DNN because CNN captures structural patterns in VLE interactions. Keras Tuner explored hyperparameters that optimize the model's performance and avoid excessive testing. A learning activity recommendation was executed to provide each learner with a list of activities adapted to their LS. After that, a prediction of students' performance was executed to measure the effectiveness of our approach. The findings indicate an outcome enhancement for 51.37% of students, demonstrating that the proposed model reduces the proportion classified in the "Fail" category and could be used to enhance success in distance learning. The use of CNN and DNN with Keras Tuner improved the personalization and outcomes. However, this study is limited by insufficient hyperparameters tested and by the absence of real-world validation, leaving practical deployment challenges unexplored. To facilitate the practical adoption of this framework, the DL-based LS detection model can be embedded into VLE through an API that automatically detects at-risk students and produces customized educational activity suggestions. Such recommendations can be provided to the students and instructors for timely interventions.

In future work, the proposed framework will be extended to incorporate multi-modal learning data, such as video-based engagement analysis, textual information extracted from forum discussions, and clickstream data that represents students' navigation and interaction patterns. We will focus on validation of the proposed framework in a real educational platform. We also plan to explore the use of graph neural networks (GNN) to achieve more effective results and reinforcement learning (RL) to dynamically

personalize learning pathways. This framework can guide institutions in deploying ML-driven personalized learning at scale.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Bensalem Assri	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓		✓	✓
Amine Mrhari	✓	✓		✓	✓					✓	✓	✓	✓	
Moulay Youssef Hadi		✓		✓	✓					✓		✓	✓	
Mohamed Boutaib		✓			✓					✓				

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

The data that support the findings of this study are openly available in the Open University at https://doi: 10.1038/sdata.2017.171, reference number [32].

REFERENCES

[1] S. Bashir and A. L. Lapshun, "E-learning future trends in higher education in the 2020s and beyond," *Cogent Educ.*, vol. 12, no. 1, p. 2445331, Dec. 2025, doi: 10.1080/2331186X.2024.2445331.

[2] S. Mudenda *et al.*, "Impact of Coronavirus Disease 2019 (COVID-19) on College and University Students: A Global Health and Education Problem," *Aquademia*, vol. 4, no. 2, p. ep20026, Aug. 2020, doi: 10.29333/aquademia/8494.

[3] E. J. Sintema, "Effect of COVID-19 on the Performance of Grade 12 Students: Implications for STEM Education," *Eurasia J. Math. Sci. Technol. Educ.*, vol. 16, no. 7, Apr. 2020, doi: 10.29333/ejmste/7893.

[4] J. D. Dayag and S. Faramarzi, "Virtual Learning Environment: Enriching 21st Century Academic Landscape," *Eur. J. Theor. Appl. Sci.*, vol. 2, no. 1, pp. 248–253, Jan. 2024, doi: 10.59324/ejtas.2024.2(1).19.

[5] S. Y. Chen and J.-H. Wang, "Individual differences and personalized learning: a review and appraisal," *Univers. Access Inf. Soc.*, vol. 20, no. 4, pp. 833–849, Nov. 2021, doi: 10.1007/s10209-020-00753-4.

[6] O. E. Aissaoui, Y. E. A. El Madani, L. Oughdir, and Y. E. Allioui, "Combining supervised and unsupervised machine learning algorithms to predict the learners' learning styles," *Procedia Comput. Sci.*, vol. 148, pp. 87–96, 2019, doi: 10.1016/j.procs.2019.01.012.

[7] M. F. I. Cardozo *et al.*, "Mapping the learning styles of medical students in Brazil," *BMC Med. Educ.*, vol. 24, no. 1, p. 47, Jan. 2024, doi: 10.1186/s12909-024-05028-7.

[8] I. Azzi, A. Radouane, L. Laaouina, A. Jeghal, A. Yahyaouy, and H. Tairi, "Fuzzy Classification Approach to Select Learning Objects Based on Learning Styles in Intelligent E-Learning Systems," *Informatics*, vol. 11, no. 2, p. 29, May 2024, doi: 10.3390/informatics11020029.

[9] Q. Clark, J. Mohler, and A. Magana, "Learning Style Dynamics," in *2015 ASEE Annual Conference and Exposition Proceedings*, Seattle, Washington: ASEE Conferences, Jun. 2015, pp. 26.1076.1-26.1076.23, doi: 10.18260/p.24413.

[10] R. Nazempour and H. Darabi, "Personalized Learning in Virtual Learning Environments Using Students' Behavior Analysis," *Educ. Sci.*, vol. 13, no. 5, 2023, doi: 10.3390/educsci13050457.

[11] M. S. Hasibuan, R. Z. A. Aziz, D. A. Dewi, T. B. Kurniawan, and N. A. Syafira, "Recommendation Model for Learning Material Using the Felder Silverman Learning Style Approach," *HighTech Innov. J.*, vol. 4, no. 4, pp. 811–820, 2023, doi: 10.28991/HIJ-2023-04-04-010.

[12] Q. Xiao, Y.-W. Zhang, X.-Q. Xin, and L.-W. Cai, "Sustainable Personalized E-Learning through Integrated Cross-Course Learning Path Planning," *Sustainability*, vol. 16, no. 20, 2024, doi: 10.3390/su16208867.

[13] S. M. Nafea, F. Siewe, and Y. He, "On Recommendation of Learning Objects Using Felder-Silverman Learning Style Model," *IEEE Access*, vol. 7, pp. 163034–163048, 2019, doi: 10.1109/ACCESS.2019.2935417.

[14] S. Amin, M. I. Uddin, A. A. Alarood, W. K. Mashwani, A. O. Alzahrani, and H. A. Alzahrani, "An adaptable and personalized framework for top-N course recommendations in online learning," *Sci. Rep.*, vol. 14, no. 1, 2024, doi: 10.1038/s41598-024-56497-1.

[15] Z. Shahbazi and Y.-C. Byun, "Agent-Based Recommendation in E-Learning Environment Using Knowledge Discovery and Machine Learning Approaches," *Mathematics*, vol. 10, no. 7, 2022, doi: 10.3390/math10071192.
[16] N. Singh, V. K. Gunjan, A. K. Mishra, R. K. Mishra, and N. Nawaz, "SeisTutor: A Custom-Tailored Intelligent Tutoring System and Sustainable Education," *Sustainability*, vol. 14, no. 7, 2022, doi: 10.3390/su14074167.
[17] W. Kaiss, K. Mansouri, and F. Poirier, "Effectiveness of an Adaptive Learning Chatbot on Students' Learning Outcomes Based on Learning Styles," *Int. J. Emerg. Technol. Learn.*, vol. 18, no. 13, pp. 250–261, 2023, doi: 10.3991/ijet.v18i13.39329.
[18] M. Mustapha, M. Qbadou, E.-S. Aoula, and S. Gouraguine, "The new e-learning adaptation technique based on learner's learning style and motivation," *J. Educ. Learn.*, vol. 17, no. 3, pp. 472–482, 2023, doi: 10.11591/edulearn.v17i3.20826.
[19] A. A. F. Osman, "A New Approach to Machine Learning Algorithms in Adaptive E-Learning Systems," *J. Inf. Syst. Eng. Manag.*, vol. 10, no. 15s, pp. 419–432, Mar. 2025, doi: 10.52783/jisem.v10i15s.2480.
[20] H. A. El-Sabagh, "Adaptive e-learning environment based on learning styles and its impact on development students' engagement," *Int. J. Educ. Technol. High. Educ.*, vol. 18, no. 1, 2021, doi: 10.1186/s41239-021-00289-4.
[21] O. Tapalova and N. Zhiyenbayeva, "Artificial Intelligence in Education: AIEd for Personalised Learning Pathways," *Electron. J. E-Learn.*, vol. 20, no. 5, pp. 639–653, 2022, doi: 10.34190/ejel.20.5.2597.
[22] S. Bousalem, F. Benchikha, and N. Marir, "Personalized Learning Through MBTI Prediction: A Deep Learning Approach Integrated with Learner Profile Ontology," *IEEE Access*, vol. 13, pp. 30861–30873, 2025, doi: 10.1109/ACCESS.2025.3542701.
[23] S. Almuqhim and J. Berri, "AI-Driven Personalized Microlearning Framework for Enhanced E-Learning," *Comput. Appl. Eng. Educ.*, vol. 33, no. 3, 2025, doi: 10.1002/cae.70040.
[24] Y. Wang, "Artificial intelligence in student management systems to enhance academic performance monitoring and intervention," *Sci. Rep.*, vol. 15, no. 1, 2025, doi: 10.1038/s41598-025-19159-4.
[25] A. R. Sayed, M. H. Khafagy, M. Ali, and M. H. Mohamed, "Exploring the VAK model to predict student learning styles based on learning activity," *Intell. Syst. Appl.*, vol. 25, 2025, doi: 10.1016/j.iswa.2025.200483.
[26] Y. Liang and P. He, "Design and implementation of an AI-powered adaptive learning system for University students," *Int. J. Data Sci. Anal.*, vol. 22, no. 1, 2026, doi: 10.1007/s41060-026-01045-5.
[27] I. Chetoui, E. El Bachari, and M. El Adnani, "Multi-modal graph neural networks for cross-domain educational recommendation: integrating behavioral analytics and institutional context for personalized learning," *Smart Learn. Environ.*, vol. 13, no. 1, 2026, doi: 10.1186/s40561-026-00452-2.
[28] S. G. Essa, T. Celik, and N. E. Human-Hendricks, "Personalized Adaptive Learning Technologies Based on Machine Learning Techniques to Identify Learning Styles: A Systematic Literature Review," *IEEE Access*, vol. 11, pp. 48392–48409, 2023, doi: 10.1109/ACCESS.2023.3276439.
[29] D. Iacobucci, S. Román, S. Moon, and D. Rouziès, "A Tutorial on What to Do With Skewness, Kurtosis, and Outliers: New Insights to Help Scholars Conduct and Defend Their Research," *Psychol. Mark.*, vol. 42, no. 5, pp. 1398–1414, 2025, doi: 10.1002/mar.22187.
[30] X. Liu, B. Wang, S. Jin, and Z. Song, "Missing value interpolation algorithm for long-term temperature observation data based on data augmentation multiple interpolation method," *Results Eng.*, vol. 27, p. 106211, Sep. 2025, doi: 10.1016/j.rineng.2025.106211.
[31] S. Roy, R. Mehera, R. K. Pal, and S. K. Bandyopadhyay, "Hyperparameter optimization for deep neural network models: a comprehensive study on methods and techniques," *Innov. Syst. Softw. Eng.*, vol. 21, no. 3, pp. 789–800, 2025, doi: 10.1007/s11334-023-00540-3.
[32] J. Kuzilek, M. Hlosta, and Z. Zdrahal, "Open University Learning Analytics dataset," *Sci. Data*, vol. 4, no. 1, p. 170171, Nov. 2017, doi: 10.1038/sdata.2017.171.

BIOGRAPHIES OF AUTHORS



Bensalem Assri is a Ph.D. student at Scientific Research and Innovation Laboratory, Higher School of Technology, Ibn Tofail University, Kenitra, Morocco. He earned a master's degree in software engineering for the Cloud from the Faculty of Science, Ibn Tofail University, Morocco. He is currently a professor at the Moroccan Ministry of National Education, Preschool, and Sports. His main research interests focus on programming languages, machine learning, deep learning, and distance learning. His work has appeared in peer-reviewed International Journal of Online and Biomedical Engineering (IJOE). He can be contacted at email: bensalem.assri@uit.ac.ma.



Amine Mrhari earned his Ph.D. from Ibn Tofail University, Kenitra, Morocco, in 2022. He is an associate professor at the National Institute of Planning and Urbanism (INAU), Rabat, and a member of the Computer Science Research Laboratory at Ibn Tofail University, Kenitra. His research interests include game theory, deep learning, GeoAI, digital twins, smart-city applications, and resource management in cloud computing. His work has appeared in peer-reviewed journals such as the Journal of Advances in Information Technology, Informatica, the International Journal of Online and Biomedical Engineering (IJOE), and Complex Systems. He can be contacted at email: a.mrhari@inau.ac.ma.



Moulay Youssef Hadi did his Ph.D. in 2008 at Mohammed V University, Faculty of Sciences, Rabat, Morocco. He is a full professor at Ibn Tofail University in Kenitra, Morocco. He is also the Deputy Director of Educational Affairs at Kenitra's Higher School of Technology. His research specializes in the subjects of model-driven engineering, programming languages, software engineering, artificial intelligence, virtualization, and cloud computing. He has made significant contributions and publications. He can be contacted at email: hadi@uit.ac.ma.



Mohamed Boutaib is a Ph.D. student at Scientific Research and Innovation Laboratory, Higher School of Technology, Ibn Tofail University, Kenitra, Morocco. He earned a master's degree in software engineering for the cloud from the Faculty of Science, Ibn Tofail University, Morocco. He is currently a professor at the Moroccan Ministry of National Education, Preschool, and Sports. His main research interests focus on machine learning, deep learning and e-learning. He can be contacted at email: Mohamed.boutaib1@uit.ac.ma.