Heliyon 10 (2024) e39088

Contents lists available at ScienceDirect

Heliyon

journal homepage: www.cell.com/heliyon

Research article

Towards multi-agent system for learning object recommendation

Ahmed Salem Mohamedhen [a], Abdullah Alfazi [b], Nouha Arfaoui [c], Ridha Ejbali [c,*], Mohamedade Farouk Nanne [a]

[a] *Department Mathematics and Computer Science, Faculty of Science and Technology, University of Nouakchott, Nouakchott, Mauritania*
[b] *Department of Computer Science, College of Computer Engineering and Sciences in Al-Kharj, Prince Sattam Bin Abdulaziz University, Al-Kharj 11942, Saudi Arabia*
[c] *Research Team in Intelligent Machines (RTIM), National Engineering School of Gabes, University of Gabes, Gabes 6029, Tunisia*

ARTICLE INFO

Keywords:
Deep learning
Knowledge level
Learning object
Learning style
Recommender system

ABSTRACT

The rapid increase of online educational content has made it harder for students to find specific information. E-learning recommender systems help students easily find the learning objects they require, improving the learning experience. The effectiveness of these systems is further improved by integrating deep learning with multi-agent systems. Multi-agent systems facilitate adaptable interactions within the system's various parts, and deep learning processes extensive data to understand learners' preferences. This collaboration results in custom-made suggestions that cater to individual learners. Our research introduces a multi-agent system tailored for suggesting learning objects in line with learners' knowledge levels and learning styles. This system uniquely comprises four agents: the learner agent, the tutor agent, the learning object agent, and the recommendation agent. It applies the Felder and Silverman model to pinpoint various student learning styles and organizes educational content based on the newest IEEE Learning Object Metadata standard. The system uses advanced techniques, such as Convolutional Neural Networks (CNN) and Multilayer Perceptrons (MLP), to propose learning objects. In terms of creating personalized learning experiences, this system is a considerable step forward. It effectively suggests learning objects that closely match each learner's personal profile, greatly enhancing student engagement and making the learning process more efficient.

1. Introduction

Technological advanced advances have made e-learning one of the most efficient learning strategies in recent years [1], providing students with flexible access to course materials anywhere, and anytime [2]. Its significance was amplified during the COVID-19 pandemic, proving to be an effective alternative to traditional classroom learning [3]. With the vast selection of online courses available, learners frequently require guidance in selecting the most suitable ones [4]. To tackle this, recommender systems are utilized in e-learning, delivering personalized services by discerning individual learner preferences [5]. These recommender systems have been evolving, especially with the advent of new machine learning techniques and the expansion of big data [6]. Traditionally, these systems fall into three major types: Content-Based, Collaborative Filtering, and Hybrid [6,7]. The recent increase in data has led to the appearance of deep learning (DL) in many fields related to computer science, including recommender systems [8]. The advent of DL in recent years has brought significant improvements to these systems [9,10], and overcome the drawbacks of traditional recommender

* Corresponding author.
E-mail address: ridha_ejbali@ieee.org (R. Ejbali).

https://doi.org/10.1016/j.heliyon.2024.e39088
Received 31 March 2024; Received in revised form 4 October 2024; Accepted 7 October 2024
Available online 11 October 2024

systems [8]. In e-learning, where customization and adaptability are key, the potential of DL to enhance learning object recommendations is immense.

Nevertheless, the intricate and variable nature of e-learning environments present distinct challenges. This is where the role of Multi-Agent Systems (MAS), comprising independent agents working in concert for a common goal, becomes pivotal [11,12]. The MASs are being introduced in online education to solve problems and to make the most out of networked computers to improve the learning experience [13]. These systems are tailored to be more personal, expand easily, and work smarter. Every agent in the system has its own task – some figure out what the student likes, others keep track of how far the student has come, some manage the Learning objects (LO), and some help everyone get along and work together. Because each agent takes care of a different part of teaching, the system can suggest LOs that are just right for each student. This is important in teaching because every student is different and needs different kinds of help. Also, these systems are good at adapting to new learn or new ways of learning, which is great because the way we learn online keeps changing all the time.

The objective of this paper is to propose a MAS. This system comprises four agents: the learner agent, the tutor agent, the learning object agent, and the recommender agent, and it is designed to match learners with the most suitable LOs, considering their individual choices, preferences, and learning styles, as well as their knowledge level.

The overall structure of this article takes the form of six sections, including this introduction section. In section 2 delves into related work in the field, while the background information is provided in section 3. In section 4, we introduce our proposed system. Section 5 focuses on the experimental analysis and the outcomes we achieved. Finally, the conclusion is presented in section 6.

2. Related works

This section describes the recommendation systems using the MAS concept. Recommender systems in e-learning have been greatly improved by the incorporation of multi-agent systems, which address problems such as data sparsity, adaptation, and cold starts. These systems can cluster users and items, anticipate associations, and provide personalized information based on learning style and preferences using social networks and collaborative agents [14–16]. Several e-learning systems have been proposed and implemented, supporting various combinations of agents, using different technologies, and employing diverse methodologies. The common objective of these systems is to enhance the students' learning abilities.

In their comparative study the authors [11,17], presented the MAS in the e-learning field, covering the period from the emergence of MAS to 2014. However, in this paper, we only consider MAS proposals made after 2014, which are summarized in Table 1.

In [18], the authors suggested a method for developing a MAS to recommend LOs. This approach considers learning styles, preferences, and previous knowledge. They implemented the system using collaborative filtering and the BDI (Beliefs-Desires-Intentions) architecture.

In [19], the authors introduced a personalized e-learning system that utilizes a MAS approach and ontology-based methods to recommend a learning path to learners based on their individual profiles.

In [20], the MAS for recommending accessible learning object (SIMROAA) was developed to assist disabled learners and instructors' computation domain. In order to give the most relevant LOs to the students with disabilities based on their preferences, the system employs a content-based approach to compare new and favorite learning objects.

In [21] a MAS was proposed to recommend LOs based on the pervious ratings, learning styles, students', and prior knowledge. This system grouped LOs using the k-means algorithm and employed a combined approach of content-based, collaborative filtering, and knowledge-based methods in the recommendation process.

In [22], the authors introduced a MAS that delivers LOs to learners based on their preferences and learning style. The system initially uses a content-based approach to generate the top-N learning objects and then reclassifies them based on students' comments.

In [23], the author presented a personalized recommendation system, that uses the MAS approach. The system uses a combination of collaborative filtering and K-NN classification to recommend items based on user preferences and requirements.

In [24], a MAS was suggested to offer learners the most appropriate learning materials according to their learning style. This system used the VARK (Visual, Auditory, Read/Write, Kinesthetic) model to identify learning styles and applied a deep neural network for making recommendations.

In [25] a MAS presented for providing LOs to the learners based on their learning style, knowledge level, and disabilities. To determine learning styles, the system used the Felder-Silverman Learning Style Model (FSLSM). The best possible learning path was then suggested by combining ontology-based and Q-learning techniques.

In [26], the authors presented an Emotional Multi-Agent System (EMAS) designed to provide learners with the most suitable book based on their needs, preferences, and emotions. The system used a hybrid approach that combined collaborative filtering, content-based filtering, tree-based filtering, and emotion-aware techniques to achieve this objective.

In [27], the authors presented a MAS that uses rules-based and ontology to suggest LOs based on the learners' preferences and knowledge level. Additionally, the system considers the learners' social situation, COVID-19 history, and mental status in the recommendation process.

3. Backgrounds

3.1. Learning object

Researchers have attempted to define the LO since its apparition in order to fill gaps in its definition, resulting in several proposed

Table 1
Overview of the multi-agent system for learning object recommendations.

Article	Attributes	Methods	Dataset	Evaluation's Methods
[18], **2021**	Learning style, preferences, and prior knowledge	Collaborative filtering	–	
[19], **2022**	Learner profile	Ontology-based	–	
[20], **2019**	Learners' preferences	Content-based	ROAA (Repository of Accessible Learning Objects)	Accuracy 93 %
[28], **2015**	Learning styles, students' prior knowledge, and previous ratings	Hybrid approach	Repository Federation of Learning Objects Colombia (FROAC)	Precision 54 %
[22], **2021**	Learners' preferences and learning style	Content-based	The dataset of university 8012 courses	
[23], **2022**	Users' preferences and needs	Collaborative filtering and KNN	–	Precision 78 % Recall 60 % F1-score 70 %
[24], **2021**	Learning style	Deep neural network and VRAK	Moodle	
[25], **2021**	Learning style, disabilities, and Knowledge level	Q-learning, Ontology-based, and FSLSM	Moodle	
[26], **2021**	Learners' preferences, needs, and emotions	collaborative filtering, content-based filtering, tree-based filtering, and emotion-aware	Faculty database	
[27], **2022**	Learners' preferences, and Knowledge level,	Rules-based, and ontology-based	–	

definitions [29]. However, in this article, we adopt the definition of the IEEE Std 1484.12.1™-2020, which defines a LO as an "entity, digital or non-digital, that is used for learning, education, or training" [30]. The authors [31,32] presented and compared the Learning Object Metadata (LOM) standards, concluding that the most widely used metadata standard for representing LOs is the IEEE LOM. This finding was later corroborated by Refs. [33,34].

The IEEE standard LOM specifies how LO should be described. It has nine categories (See Fig. 1): General, Life cycle, Meta-Metadata, Technical, Educational, Rights, Relationship, Annotation, and Classification [30].

The goal of the standard is to simplify the process of finding, evaluating, acquiring, sharing, trading, and using LOs for students, teachers, and automated software systems [30]. The IEEE Std 1484.12.1™-2020 standard on metadata for LOs describes each category as follows in Table 2 [30].

3.2. Learning style

According to Ref. [35], Herb Thelen was the first introduced the concept of learning styles to the field of educational psychology in the 1950s. This concept refers to an individual learner's preferred method of participating in the learning process [36]. Furthermore, researchers concur that identifying learners' preferred learning styles is essential for customizing instruction, as shown Fig. 2, meeting students' needs, and enhancing learning opportunities, especially in learning environments [37]. The literature has been shown the diversity of learning styles models, such as: Felder Silverman Learning Style Model (FSLSM) [38], VARK (Visual, Aural, Read/write, and Kinesthetic) [39], Learning Style Model of Kolb [40], and the learning style model of Honey and Mumfold [41]. Among those, the most commonly used is FSLMS according to [37,42,43].

The FSLSM has four dimensions, which each contain two types of learning styles as illustrated in Fig. 3. The four dimensions are named: input, perception, processing, and understanding [42]. The full description of FSLSM is presented in Table 3.

3.3. Recommendation techniques

Neural networks are often employed in recommender system domain. These include feedforward, convolutional, and recurrent neural network types [10], each of which has a distinct purpose in suggestions. Deep learning, particularly with neural networks that handle complicated data and provide tailored suggestions, has greatly improved the area. Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), and Multi-Layer Perceptrons (MLP) are the important deep learning approaches used in this context [10,45,46].

3.3.1. Multilayer Perceptrons

Multilayer Perceptrons (MLPs), a form of artificial neural networks, has a layered design that helps it manage complex tasks. One or more hidden layers, an output layer, and an input layer are the usual three layers that make up a MLP. The connections between one layer and the next are complete. The hidden layers, situated between the input and output layers, are essential for transforming and processing the input data to enable linear separation before reaching the output layer. The number of hidden layers in a MLP can be customized to meet specific project requirements [45,46].

3.3.2. Convolutional Neural Networks

Convolutional Neural Networks (CNNs) are a type of feed-forward neural network commonly used in deep learning. They consist of five major parts: the input layer, the convolution layer, the activation function, the pooling layer, and the fully connected layer. The convolutional layer pulls out features from the input, and the pooling layer cuts down the size of the data and helps prevent overfitting. The fully connected layer, which is often the final layer, focuses on classifying the extracted features. Together, these components enable CNNs to efficiently process and analyze data, making them crucial in deep learning applications [45,46].

3.3.3. Recurrent Neural Nets

Recurrent Neural Nets (RNNs) are useful for sequential data analysis. They have input, output, and hidden layers like neural networks. However, their unique feature is that RNNs maintain different states at different time steps. In this model, the hidden layer's output at one time step affects the hidden layer at a later time step. RNN versions, such as Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) networks, are widely used to handle problems such as the vanishing gradient problem [45,46].

4. Proposed system

The proposed system aims to recommend LOs that are most suitable for each learner's profile and preferences. This section provides an overview of the proposed conceptual framework for personalized e-learning recommendation systems. As illustrated in Fig. 4, the suggested architecture is divided into three layers: the presentation layer, the recommendation layer, and the data layer. Each layer integrates numerous key components that interact both with components at different levels and with each other. The recommendation layer is a central element that forms the foundation of the customized e-learning system, as detailed below and represented in the Architectural Framework design in Fig. 4.

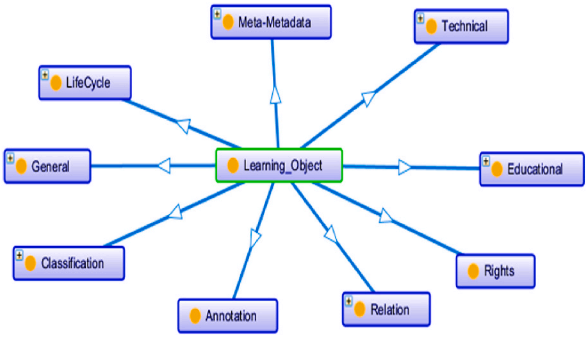

Fig. 1. Ieee standard for learning object metadata.

Table 2
Summary of the IEEE standard for learning object metadata.

Category	Description
General	Groups the overall information that characterizes the LO.
Life cycle	Provides an overview of the history and current state of the LO, as well as the entities that have influenced its development over time.
Meta-metadata	Focuses on the metadata record itself instead the LO it describes.
Technical	Explains the technical prerequisites and benefits of the LO.
Educational	Outlines the fundamental educational or pedagogical features of the LO.
Rights	Explains how the LO can be used and its intellectual property right
Relationship	Specifies the relationship between the LO and other LOs,
Annotation	Offers insights on the educational application of the LO, including details about the timing and authorship of the provided comments
Classification	Specifies the placement of the LO within a specific classification system

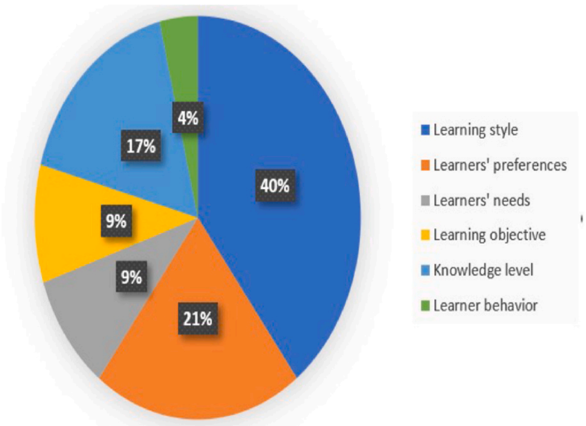

Fig. 2. Distribution of attributes learners [29].

4.1. Presentation layer

The presentation layer facilitates user-system interaction and serves as a communication conduit with the system's underlying layers. It transmits pertinent information to both the recommendation layer and the data layer, while also displaying the outcomes of interactions with the framework's internal layers and modules. Serving as the primary interface for user access, this layer embodies a graphical user interface (GUI) encompassing the administrator interface, the tutor interface, and the learner interface, as depicted in Fig. 4.

4.1.1. Administrator interface

The administrator interface plays an essential role in the system, enabling the administrator to manage access and databases. This includes the creation, modification, and deletion of user accounts, as well as the detailed management of roles and permissions for tutors and learners. Fig. 5 illustrates the main activities of the administrator (see Fig. 6).

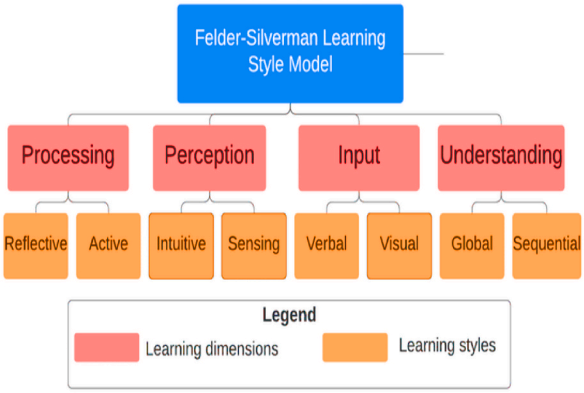

Fig. 3. Felder Silverman Learning Style Model [44].

Table 3
Description of Felder Silverman learning style model.

Dimension	Description	Learning styles
Input	Refers to the type of information a student prefers.	Visual: learners prefer visual presentations, including videos, pictures, and diagrams. Verbal: learners favor written and spoken explanations.
Perception	Concerns how students perceive and understand information.	Sensing: learners are inclined towards concrete, practical thinking, and focus on facts and procedures. Intuitive: learners are drawn to theories and abstract concepts, often thinking conceptually and innovatively.
Processing	Relates to how students process information.	Active: learners engage by experimenting and working collaboratively in groups. Reflective: learners prefer to think and analyze information independently or with a trusted.
Understanding	Deals with how students' progress towards understanding.	Sequential: learners prefer to understand in a logical, step-by-step manner, engaging in linear thinking. Global: learners tend to grasp concepts in large steps, through a holistic thinking process.

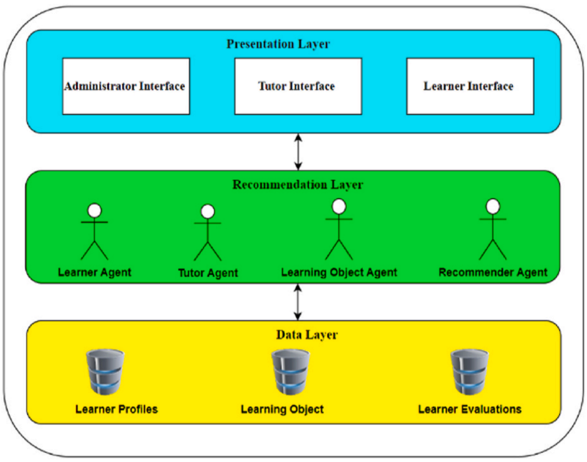

Fig. 4. Conceptuel model of proposed system.

4.1.2. Tutor interface

The tutor interface enables tutors to supervise courses, develop and modify learning objects to accommodate different learning styles, create tests to assess knowledge levels, and monitor learners' progress throughout the learning process. Fig. 5 shows the tutor's main activities.

4.1.3. Learner interface

The learner interface enables the management of the learner's account system, including registration and login procedures.

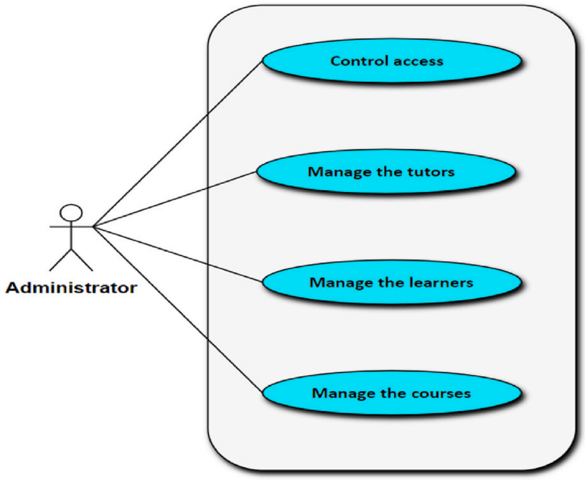

Fig. 5. Use case diagram for the administrator.

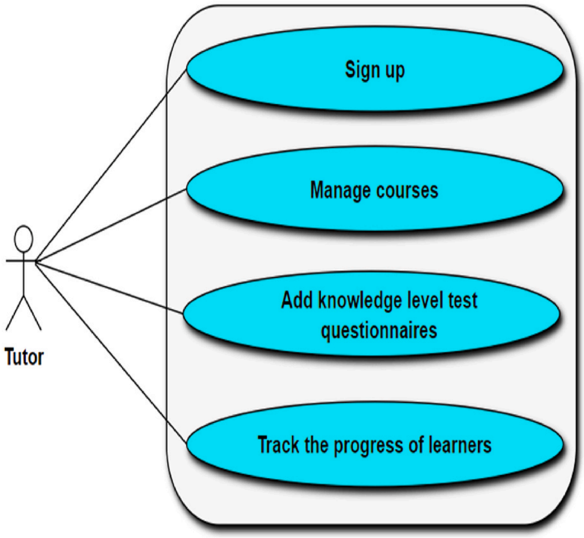

Fig. 6. Use case of the tutor.

Additionally, it plays a crucial role in presenting the learning objects recommended by the system and in collecting learner evaluations. Fig. 7 shows the learner's main activities.

4.2. Recommendation layer

The recommendation layer constitutes the core of the proposed system, encompassing four distinct agents: the learner agent, the tutor agent, the learning object agent, and the recommendation agent. Each agent is responsible for managing a specific aspect of the recommendation process, working collaboratively to form an integrated personalized recommendation system for e-learning.

4.2.1. Learner agent

The learner agent gathers various types of data from learners, including personal details, preferences, knowledge levels, learning styles, and interaction histories. This information is used to create individual learner profiles, which store essential data about each learner's preferences. These profiles enable the system to deliver personalized learning objects tailored to each learner's unique needs. It has two primary functions.

- **Learning Style Detection**

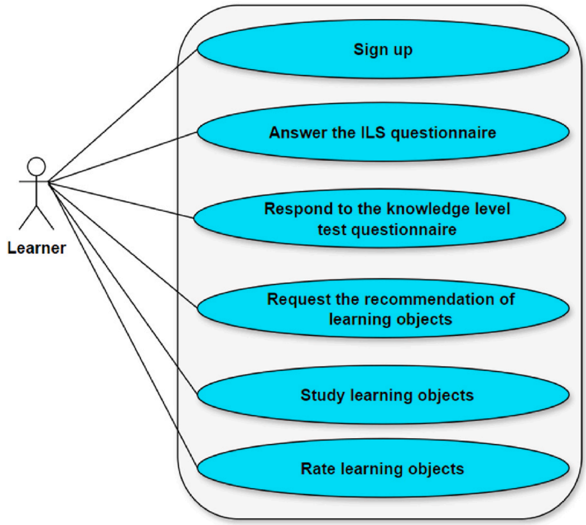

Fig. 7. Use Case of the learner.

Acquiring information about learners' learning styles is an important task, as it is the initial step in building a personalized e-learning recommendation system, which in turn is needed to develop a learner profile containing the learner's personal preferences. In this context, the challenge is how to obtain the learning styles of new learners when the system does not have sufficient information to generate high-quality personalized recommendations. This is commonly referred to as the "cold start" problem. To determine their learning style, learners must complete the 44-question Index of Learning Style (ILS)[1] questionnaire. The LA then analyzes the responses and identifies the learning style based on the FSLSM model. The proposed architecture for initializing learners' learning styles comprises the following steps, as shown in Fig. 8.

- **Knowledge Level Detection**

For each course, learners need to take a first test (see Fig. 9). The scores from this test help figure out if the student is just starting out (beginner), in the middle (intermediate), or good at it (advanced).

Fig. 8 illustrates how a new student interacts with the system. It specifically outlines how the different agents work together to determine the student's learning style and current level of knowledge.

4.2.2. Learning object agent

The Learning Object Agent (LOA) in the system plays a key role in managing LOs. It systematically classifies and arranges these LOs in line with the criteria set by the 2020 IEEE LOM standard and FSLSM. Once categorized, the LOA diligently saves them in a designated database. Fig. 10 shows the interactions among the agents to store the LOs in the database.

4.2.3. Recommendation agent

The Recommendation Agent (RA) plays a crucial role in the system by recommending LOs to students based on their individual learning styles and knowledge levels. After each recommendation, the RA plays a crucial role in enhancing the learning process by sending feedback and rating from the learners to the TA. Fig. 11 depicts the recommendation procedure.

4.2.4. Tutor agent

The Tutor Agent (TA) is responsible for managing tutor access into system and assist the tutors for monitoring the progress of students in their learning process. additionally, TA sends feedback from learners to tutors to fills any gaps in LOs and develops new LOs as needed. This loop of recommendation and feedback ensures that the learning experience is continuously optimized and tailored to the needs of each individual learner.

4.3. Data layer

The data layer facilitates the storage and retrieval of data within the proposed e-learning system. A NoSQL database (MongoDB), which houses all information related to student profiles, learning objects, and learning behavior patterns.

[1] Index of Learning Styles Questionnaire.

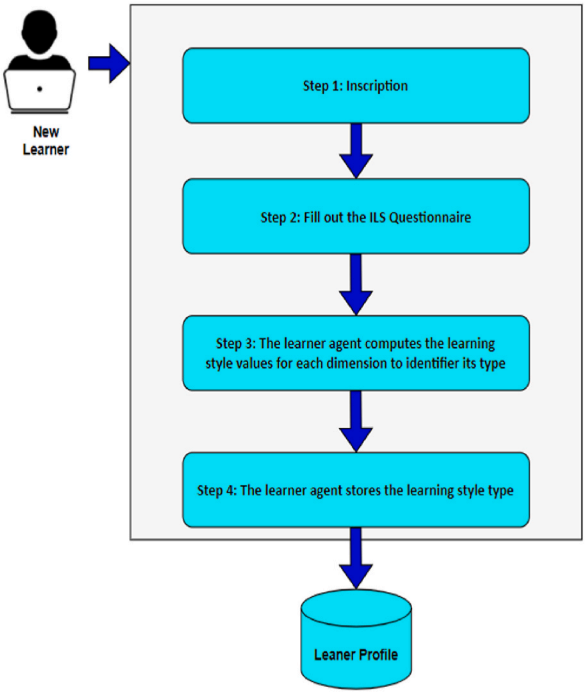

Fig. 8. Initialization architecture for Learner's learning styles.

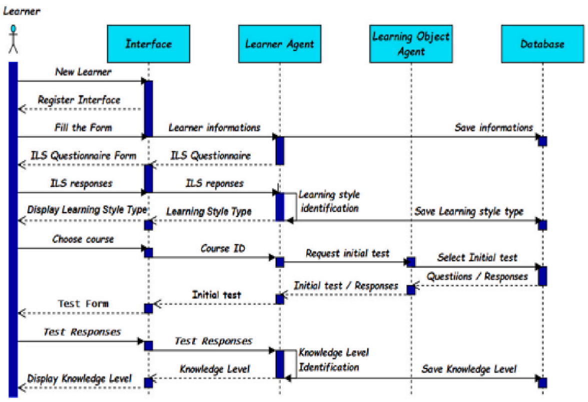

Fig. 9. Sequence diagram for Gathering learner information.

- Learner Profile Database: Contains a combination of static and dynamic information about the learners. Static data includes details such as full name, email address, and password, while dynamic data encompasses learning styles and knowledge levels of the learners.
- Learning Objects Database: Stores information about various formats of learning objects, organized using the IEEE LOM 2020 standard. This database is crucial for the effective management of digital resources, ensuring easy access, reuse, and standardized and efficient sharing of learning objects between tutors and learners.
- Learner Evaluations Database: Contains past evaluations that learners have assigned to the learning objects they have completed.

5. Experimental analysis and results

In this section, we start by describing the dataset we used for our study. Ater that, we explore and discuss the results we obtained from our experiments.

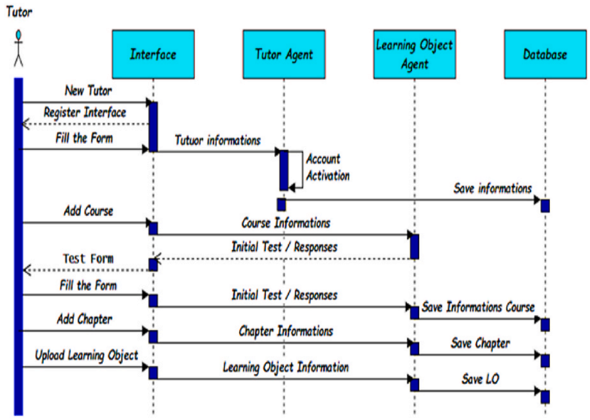

Fig. 10. Sequence diagram for storing learning object.

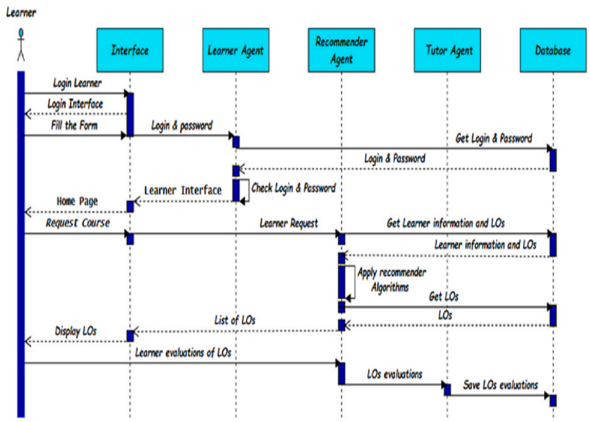

Fig. 11. Sequence diagram of the learning object recommendation process.

5.1. Dataset

In this research, a specialized dataset[2] is utilized. It comprises data on one hundred learners and ten courses. Each course is structured in three layers, forming a hierarchical network with three levels: Course, Chapter, and Learning Object (LO) as shown in Fig. 12. Therefore, the LOs are structured as the IEEE Standard LOM, Fig. 13 demonstrated the fields of LOM used in our MAS (underlined in black). Learners are characterized by both static and dynamic details. Static details, which do not play a role in recommendations, include identification number, name, gender, and age. On the other hand, dynamic details that contribute to recommendations include the learner's style of learning, preferred language, Domaine, and level of knowledge.

5.2. Results and discussions

Two deep learning-based algorithms that are appropriate for recommender systems have been developed and evaluated in order to suggest LOs to students: a Multi-Layer Perceptron (MLP) and a Convolutional Neural Network (CNN). The parameters for these algorithms are detailed in Tables 4 and 5, respectively.

We have allocated 70 % of the dataset for the training phase and 30 % for validation activities. Both the MLP and CNN algorithms have undergone a training regimen spanning 100 epochs with a consistent batch size of 32, employing the 'adam' algorithm for optimization. The ReLU activation function has been applied to the hidden layers, whereas the Softmax function has been adopted for the output layer. Several metrics are used to evaluate the MLP and CNN algorithms in this study, including accuracy, precision, recall, and F1 score. Equations (1)–(4) are used to calculate the following values: The True Positive (TP) occurs when the model accurately foresees a positive instance, and the True Negative (TN) arises when the model accurately anticipates a negative instance. Conversely, the False Positive (FP) emerges when the model inaccurately forecasts a positive outcome, and the False Negative (FN) manifest when

[2] Dataset/DatasetLearners.csv at main · medhen/Dataset · GitHub.

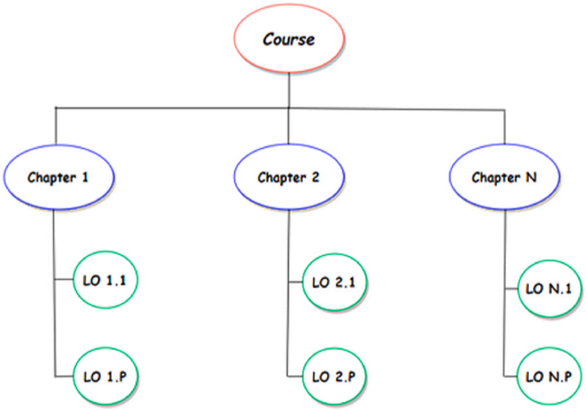

Fig. 12. Learning object Hierarchy.

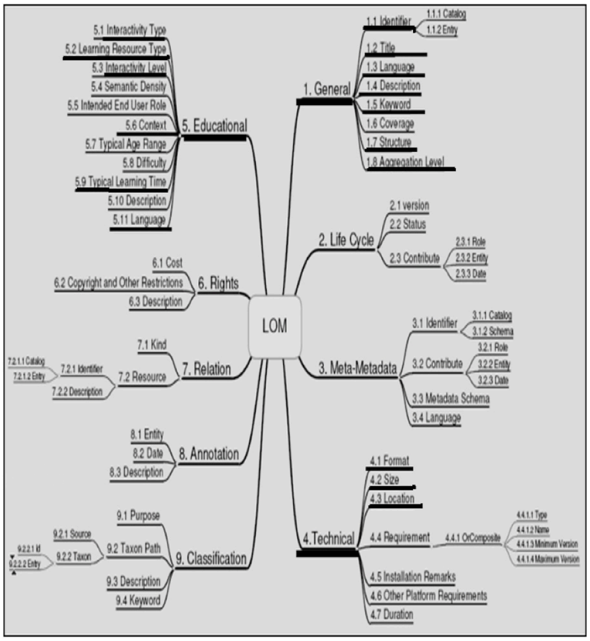

Fig. 13. Field of LOM used in Our MAS [47].

Table 4
MLP architecture.

Layer type	Input shape	Output shape
Input layer	[None, 11]	[None, 11]
Dense	[None, 11]	[None, 300]
Dense_1	[None, 300]	[None, 150]
Dropout	[None, 150]	[None, 150]
Dense_2	[None, 150]	[None, 10]

the model falls short in predicting a positive outcome. equations (1)–(4) present accuracy, precision, recall and F1-score used to evaluate our approach.

$$\text{Accuracy} = \frac{(\text{TP} + \text{TN})}{(\text{TP} + \text{TN} + \text{FP} + \text{FN})} \tag{1}$$

Table 5
CNN architecture.

Layer Type	Configuration	Input Shape	Output Shape
Input Layer		[None, 11]	[None, 11]
Reshape		[None, 11]	[None, 11, 1]
Conv1d	32 filters	[None, 11, 1]	[None, 11, 32]
Conv1d_1	64 filters	[None, 11, 32]	[None, 11, 64]
Averagepooling1d	Preserves sequence length	[None, 11, 64]	[None, 11, 64]
Flatten		[None, 11, 64]	[None, 704]
Dense	300 neurons	[None, 704]	[None, 300]
Dense_1	150 neurons	[None, 300]	[None, 150]
Dropout		[None, 150]	[None, 150]
Dense_2	10 neurons (output layer)	[None, 150]	[None, 10]

$$\mathrm{Precision} = \frac{\mathrm{TP}}{(\mathrm{TP} + \mathrm{FP})} \tag{2}$$

$$\mathrm{Recall} = \frac{\mathrm{TP}}{(\mathrm{TP} + \mathrm{FN})} \tag{3}$$

$$\mathrm{F1} - \mathrm{Score} = \frac{(2 \times \mathrm{Precision} \times \mathrm{Recall})}{(\mathrm{Precision} + \mathrm{Recall})} \tag{4}$$

Moreover, the Cross-validation (CV) is extensively employed for robust Deep Learning model evaluation. The average of the resulting scores, weighted by the number of data points in each test fold, is used as the final evaluation of model performance.

The algorithms are evaluated on various subsets of the dataset to confirm their performance over a range of data samples. The dataset is split up into five equal subsets for training.

The table (Table 6) presents the average results obtained for both algorithms, revealing that the best performance is achieved with CNN across all four metrics. The corresponding values are as follows: 93.33 % for accuracy and recall, 91.86 % for precision, and 90 % for the F1-score. This superiority in metrics implies that the CNN model is better at correctly identifying relevant learning objects while minimizing false positives and false negatives. In practical terms, higher precision means that learners receive fewer irrelevant recommendations, which enhances their learning experience by providing more targeted and useful resources. Higher recall ensures that most relevant resources are recommended, reducing the chance of missing valuable content. This balance, as reflected in the higher F1-score, indicates a robust model that can significantly improve learner outcomes by providing a more tailored and efficient learning experience. The cross-validation results for CNN are shown in Fig. 14. Especially, the algorithm performed very well in folds 0 and 3, scoring 95 % on all four metrics. Nevertheless, as Fig. 15 illustrates, the best MLP results were achieved in folds 3 and 4, with the following values: 90 % for accuracy and recall, 88.33 % for precision, and 89.66 % for the F1-score.

The accuracy for MLP and CNN models has reached the threshold of 90 %, and 93.33 %. These efficacy measures are illustrated in Figures Figs. 16 and 17, while the loss incurred by each algorithm is delineated in Figures Figs. 18 and 19.

5.3. Sensitivity analysis

The sensitivity analysis results indicate the following impacts on model accuracy.

➢ **Learning Styles (Features 0 to 7):** Each of these features exhibited a moderate impact on accuracy, with a decrease of approximately 0.05 when perturbed (see Fig. 20). This suggests that while these features are significant, their influence is less pronounced compared to other parameters.

➢ **Knowledge Level (Feature 8):** This feature displayed the most substantial impact, with a decrease in accuracy of 0.10. This finding underscores the critical role of the Knowledge Level in our model's predictions.

The conducted sensitivity analysis confirms that both the Learning Style and Knowledge Level features significantly affect the accuracy of our model. These findings validate the model's reliance on these parameters, reinforcing their importance in the context of existing domain knowledge. By systematically perturbing these features, we have elucidated their relative importance, thereby enhancing the robustness and reliability of our model's predictions.

Table 6
Algorithms comparison – Performance Metrics.

Metrics Algorithms	Accuracy	Precision	Recall	F1-Score
CNN	93.33 %	90.00 %	91.86 %	93.33 %
MLP	89.86 %	86.81 %	89.86 %	87.28 %

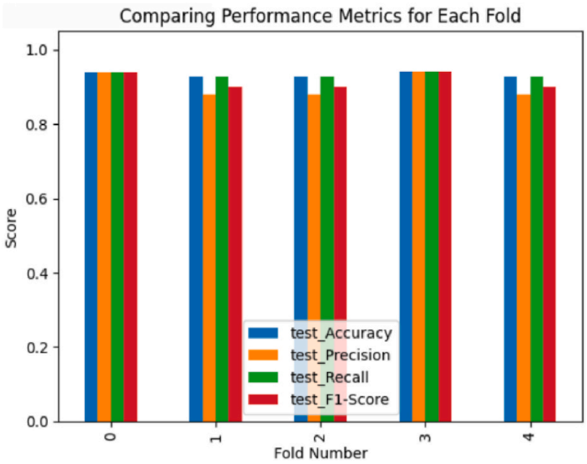

Fig. 14. CNN cross -validation performance.

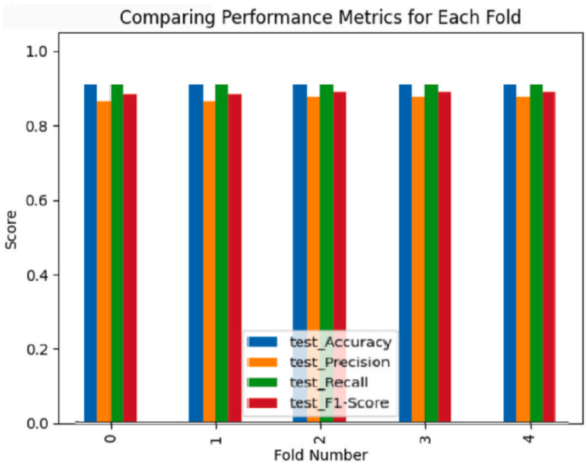

Fig. 15. MLP cross-validation performance.

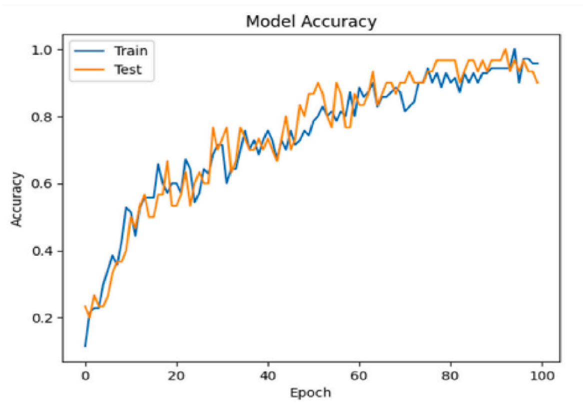

Fig. 16. CNN accuracy.

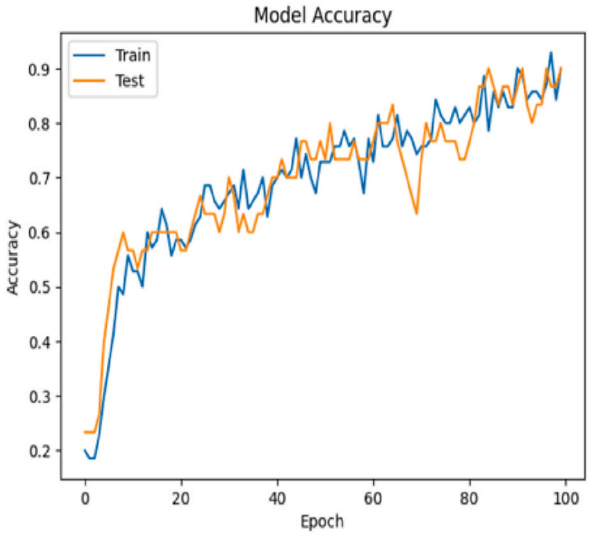

Fig. 17. Mlp accuracy.

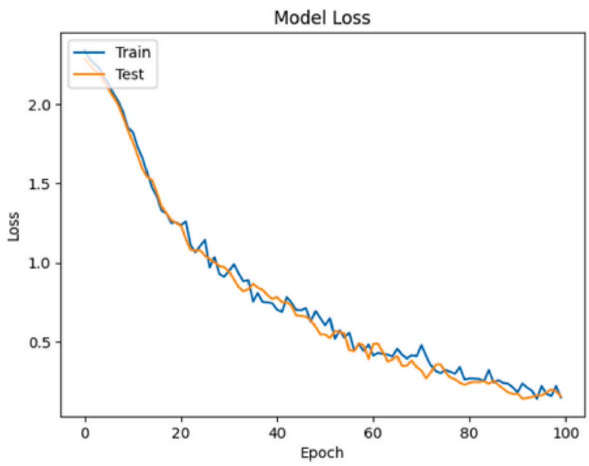

Fig. 18. CNN loss.

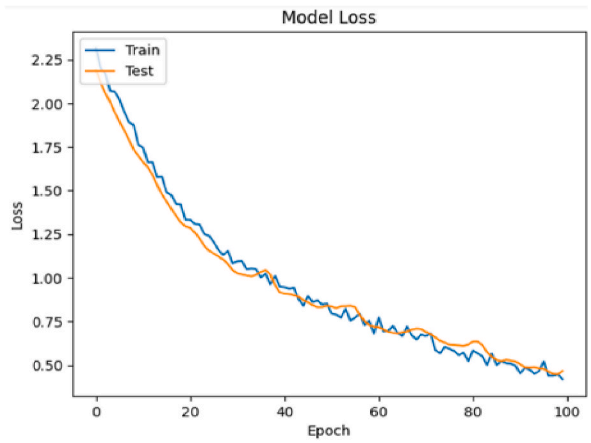

Fig. 19. Mlp loss.

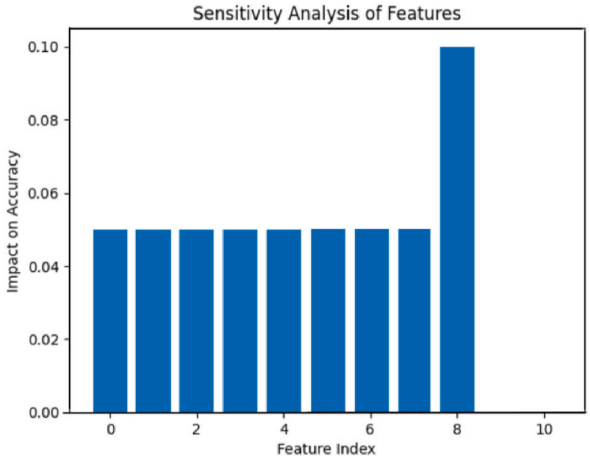

Fig. 20. Sensitivity analysis.

6. Conclusion

This study has introduced a MAS that combines the dynamic flexibility of multi-agent interactions with the personalized approach of deep learning to provide a customized e-learning experience. The proposed system, which includes learner, tutor, learning object, and recommendation agents, demonstrates the power of leveraging the FSLSM for discerning student learning styles and organizing learning objects as per the IEEE LOM standard. Through the application CNN and MLP, the system adeptly aligns educational resources with students' individual requirements, thereby enhancing engagement and efficiency in the learning process.

Our research contributes to the field of e-learning by offering a scalable, personalized recommendation system that can adapt to the continuously evolving educational technologies and methodologies. Future initiatives will aim to complete the development of the multi-agent system, integrate capabilities for real-time data analysis and expand the spectrum of learning styles catered to. This will include the advancement of automatic learning style detection, enhancing the system's ability to tailor content seamlessly.

Moreover, we plan to explore the incorporation of further machine learning strategies to elevate the precision of our recommendations.

CRediT authorship contribution statement

Ahmed Salem Mohamedhen: Writing – review & editing, Writing – original draft, Conceptualization. **Abdullah Alfazi:** Writing – review & editing, Writing – original draft, Visualization, Validation. **Nouha Arfaoui:** Writing – original draft, Methodology, Conceptualization. **Ridha Ejbali:** Writing – review & editing, Writing – original draft, Project administration, Methodology, Conceptualization. **Mohamedade Farouk Nanne:** Writing – review & editing, Writing – original draft.

7. Data availability statement

The dataset is available upon request.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] S. Souabi, A. Retbi, M.K.I. Khalidi Idrissi, S. Bennani, Recommendation systems on E-learning and social learning: a systematic review, EJEL 19 (2021) pp432–451, https://doi.org/10.34190/ejel.19.5.2482.
[2] P. Dixit, U. Pathak, E-Learning, in: Advances in Distance Learning in Times of Pandemic, first ed., Chapman and Hall/CRC, Boca Raton, 2023, pp. 273–297, https://doi.org/10.1201/9781003322252-11.
[3] A.M. Maatuk, E.K. Elberkawi, S. Aljawarneh, H. Rashaideh, H. Alharbi, The COVID-19 pandemic and E-learning: challenges and opportunities from the perspective of students and instructors, J. Comput. High Educ. 34 (2022) 21–38, https://doi.org/10.1007/s12528-021-09274-2.
[4] L. Salau, M. Hamada, R. Prasad, M. Hassan, A. Mahendran, Y. Watanobe, State-of-the-Art survey on deep learning-based recommender systems for E-learning, Appl. Sci. 12 (2022) 11996, https://doi.org/10.3390/app122311996.
[5] Q. Zhang, J. Lu, G. Zhang, Recommender systems in E-learning, Journal of Smart Environments and Green Computing 1 (2021) 76–89, https://doi.org/10.20517/jsegc.2020.06.
[6] S. Souabi, A. Retbi, M.K. Idrissi, S. Bennani, Towards an evolution of E-learning recommendation systems: from 2000 to nowadays, Int. J. Emerg. Technol. Learn. 16 (2021) 286, https://doi.org/10.3991/ijet.v16i06.18159.

[7] G. Adomavicius, A. Tuzhilin, Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions, IEEE Trans. Knowl. Data Eng. 17 (2005) 734–749, https://doi.org/10.1109/TKDE.2005.99.
[8] K. Ong, S.-C. Haw, K.-W. Ng, Deep learning based-recommendation system: an overview on models, datasets, evaluation metrics, and future trends, in: Proceedings of the 2019 2nd International Conference on Computational Intelligence and Intelligent Systems, Association for Computing Machinery, New York, NY, USA, 2020, pp. 6–11, https://doi.org/10.1145/3372422.3372444.
[9] F. Safarov, A. Kutlimuratov, A.B. Abdusalomov, R. Nasimov, Y.-I. Cho, Deep learning recommendations of E-education based on clustering and sequence, Electronics 12 (2023) 809, https://doi.org/10.3390/electronics12040809.
[10] C. Li, I. Ishak, H. Ibrahim, M. Zolkepli, F. Sidi, C. Li, Deep learning-based recommendation system: systematic review and classification, IEEE Access 11 (2023) 113790–113835, https://doi.org/10.1109/ACCESS.2023.3323353.
[11] M.U. Bokhari, S. Ahmad, Multi-agent based E-learning systems: a comparative study, in: Proceedings of the 2014 International Conference on Information and Communication Technology for Competitive Strategies, Association for Computing Machinery, New York, NY, USA, 2014, pp. 1–6, https://doi.org/10.1145/2677855.2677875.
[12] A.M. Aseere, D.E. Millard, E.H. Gerding, An agent based voting system for E-learning course selection involving complex preferences, in: 2011 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology, IEEE, Lyon, France, 2011, pp. 386–393, https://doi.org/10.1109/WI-IAT.2011.238.
[13] M. Nadrljanski, D. Vukic, D. Nadrljanski, Multi-agent systems in e-learning, in: 2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), IEEE, 2018, pp. 990–995, https://doi.org/10.23919/MIPRO.2018.8400181. Opatija.
[14] L. Yinggang, T. Xiangrong, Social recommendation system based on multi-agent deep reinforcement learning, in: 2022 IEEE 8th International Conference on Cloud Computing and Intelligent Systems (CCIS), IEEE, Chengdu, China, 2022, pp. 371–377, https://doi.org/10.1109/CCIS57298.2022.10016386.
[15] Z. Shahbazi, Y.-C. Byun, Agent-based recommendation in E-learning environment using knowledge discovery and machine learning approaches, Mathematics 10 (2022) 1192, https://doi.org/10.3390/math10071192.
[16] B. Ciloglugil, O. Alatli, M.M. Inceoglu, R.C. Erdur, A multi-agent based adaptive E-learning system, in: O. Gervasi, B. Murgante, S. Misra, C. Garau, I. Blečić, D. Taniar, B.O. Apduhan, A.M.A.C. Rocha, E. Tarantino, C.M. Torre (Eds.), Computational Science and its Applications – ICCSA 2021, Springer International Publishing, Cham, 2021, pp. 693–707, https://doi.org/10.1007/978-3-030-86970-0_48.
[17] M.U. Bokhari, S. Ahmad, Detailed analysis of existing Multi-agent based E-learning systems, in: 2014 International Conference on Computing for Sustainable Global Development (INDIACom), IEEE, New Delhi, India, 2014, pp. 388–393, https://doi.org/10.1109/IndiaCom.2014.6828165.
[18] T.O. Almeida, J.F. De Magalhães Netto, A.M.M. Lopes, Multi-agent system for recommending learning objects in E-learning environments, in: 2021 IEEE Frontiers in Education Conference (FIE), 2021, pp. 1–5, https://doi.org/10.1109/FIE49875.2021.9637258.
[19] M. Sabeima, M. Lamolle, M.F. Nanne, Towards personalized adaptive learning in e-learning recommender systems, Int. J. Adv. Comput. Sci. Appl. 13 (2022), https://doi.org/10.14569/IJACSA.2022.0130803.
[20] A.B. Mourão, J.F.M. Netto, SIMROAA multi-agent recommendation system for recommending accessible learning objects, in: 2019 IEEE Frontiers in Education Conference (FIE), 2019, pp. 1–9, https://doi.org/10.1109/FIE43999.2019.9028504.
[21] P. Rodríguez, N. Duque, D.A. Ovalle, Multi-agent system for knowledge-based recommendation of learning objects using metadata clustering, in: J. Bajo, K. Hallenborg, P. Pawlewski, V. Botti, N. Sánchez-Pi, N.D. Duque Méndez, F. Lopes, V. Julian (Eds.), Highlights of Practical Applications of Agents, Multi-Agent Systems, and Sustainability - the PAAMS Collection, Springer International Publishing, Cham, 2015, pp. 356–364, https://doi.org/10.1007/978-3-319-19033-4_31.
[22] M. Amane, K. Aissaoui, M. Berrada, A multi-agent and content-based course recommender system for university E-learning platforms, in: S. Motahhir, B. Bossoufi (Eds.), Digital Technologies and Applications, Springer International Publishing, Cham, 2021, pp. 663–672, https://doi.org/10.1007/978-3-030-73882-2_60.
[23] M. Dan, Design and research of agent-based personalized recommendation system, in: 2022 IEEE 2nd International Conference on Mobile Networks and Wireless Communications (ICMNWC), IEEE, 2022, pp. 1–4.
[24] F.N. Kivuva, E. Maina, R. Gitonga, Multi-agent adaptive e-learning system based on learning styles, Open Journal for Information Technology 4 (2021) 1.
[25] H.E. Fazazi, M. Elgarej, M. Qbadou, K. Mansouri, Design of an adaptive e-learning system based on multi-agent approach and reinforcement learning, Eng. Technol. Appl. Sci. Res. 11 (2021) 6637–6644, https://doi.org/10.48084/etasr.3905.
[26] O. Hamal, N.-E.E. Faddouli, M.H.A. Harouni, Design and implementation of the multi-agent system in education, World Journal on Educational Technology: Current Issues 13 (2021) 775–793, https://doi.org/10.18844/wjet.v13i4.6264.
[27] A. Ouatiq, K. El-Guemmat, K. Mansouri, M. Qbadou, A design of a multi-agent recommendation system using ontologies and rule-based reasoning: pandemic context, Int. J. Electr. Comput. Eng. (2022) 12, 2088-8708.
[28] P. Rodríguez, N. Duque, D.A. Ovalle, Multi-agent system for knowledge-based recommendation of learning objects using metadata clustering, in: J. Bajo, K. Hallenborg, P. Pawlewski, V. Botti, N. Sánchez-Pi, N.D. Duque Méndez, F. Lopes, V. Julian (Eds.), Highlights of Practical Applications of Agents, Multi-Agent Systems, and Sustainability - the PAAMS Collection, Springer International Publishing, Cham, 2015, pp. 356–364, https://doi.org/10.1007/978-3-319-19033-4_31.
[29] A.S. Mohamedhen, N. Arfaoui, R. Ejbali, M.F. Nanne, Learning object from emergence to nowadays: systematics literature review, J. Theor. Appl. Inf. Technol. 102 (2024). http://www.jatit.org/volumes/Vol102No4/14Vol102No4.pdf. (Accessed 5 March 2024).
[30] IEEE Standard for Learning Object Metadata, IEEE Std 1484 (2020) 1–50, https://doi.org/10.1109/IEEESTD.2020.9262118, 12.1-2020.
[31] D. Roy, S. Sarkar, S. Ghose, A comparative study of learning object metadata, learning material repositories, metadata annotation & an automatic metadata annotation tool, M. Joshi H. Boley 2 (2010).
[32] A.R. Vazquez, Y.A. Ostrovskaya, Analysis of open technological standards for learning objects, in: 2006 Fourth Latin American Web Congress, 2006, pp. 105–108, https://doi.org/10.1109/LA-WEB.2006.5.
[33] N.S. Raj, V.G. Renumol, A systematic literature review on adaptive content recommenders in personalized learning environments from 2015 to 2020, J. Comput. Educ. 9 (2022) 113–148, https://doi.org/10.1007/s40692-021-00199-4.
[34] U.C. Apoki, H.K.M. Al-Chalabi, G.C. Crisan, From digital learning resources to adaptive learning objects: an overview, in: D. Simian, L.F. Stoica (Eds.), Modelling and Development of Intelligent Systems, Springer International Publishing, Cham, 2020, pp. 18–32, https://doi.org/10.1007/978-3-030-39237-6_2.
[35] K. Benabbes, K. Housni, A. Zellou, H. Brahim, A. El Mezouary, Context and learning style aware recommender system for improving the E-learning environment, Int. J. Emerg. Technol. Learn. 18 (2023) 180–202, https://doi.org/10.3991/ijet.v18i09.38361.
[36] M.M. El-Bishouty, A. Aldraiweesh, U. Alturki, R. Tortorella, J. Yang, T.-W. Chang, S. Graf, Kinshuk, Use of Felder and Silverman learning style model for online course design, Education Tech Research Dev 67 (2019) 161–177, https://doi.org/10.1007/s11423-018-9634-6.
[37] P. Ramírez-Correa, J. Alfaro-Pérez, M. Gallardo, Identifying engineering undergraduates' learning style profiles using machine learning techniques, Appl. Sci. 11 (2021) 10505.
[38] R.M. Felder, L.K. Silverman, Learning and teaching styles in engineering education, Engineering Education 78 (1988) 674–681.
[39] N.D. Fleming, C. Mills, Not another inventory, rather a catalyst for reflection, To Improve the Academy 11 (1992) 137–155, https://doi.org/10.1002/j.2334-4822.1992.tb00213.x.
[40] D.A. Kolb, Experimental Learning: Experience as the Source of Learning and Development, Prentice-Hall, Englewood Cliffs, N.J, 1984.
[41] P. Honey, A. Mumford, The manual of learning styles, Peter Honey, Maidenhead, Berkshire (1982).
[42] A.B. Rashid, R.R.R. Ikram, Y. Thamilarasan, L. Salahuddin, N.F.A. Yusof, Z.B. Rashid, A student learning style auto-detection model in a learning management system, Eng. Technol. Appl. Sci. Res. 13 (2023) 11000–11005, https://doi.org/10.48084/etasr.5751.
[43] V. Thongchotchat, K. Sato, H. Suto, Recommender system utilizing learning style: systematic literature review, in: 2021 6th International Conference on Business and Industrial Research (ICBIR), 2021, pp. 184–187, https://doi.org/10.1109/ICBIR52339.2021.9465832.

[44] N. Zaric, R. Röpke, V. Lukarov, U. Schroeder, Gamified Learning Theory: the Moderating role of learners' learning tendencies, International Journal of Serious Games 8 (2021) 71–91, https://doi.org/10.17083/ijsg.v8i3.438.
[45] T. Liu, Q. Wu, L. Chang, T. Gu, A review of deep learning-based recommender system in e-learning environments, Artif. Intell. Rev. 55 (2022) 5953–5980, https://doi.org/10.1007/s10462-022-10135-2.
[46] S. Zhang, L. Yao, A. Sun, Y. Tay, Deep learning based recommender system: a survey and new perspectives, ACM Comput. Surv. 52 (2019), https://doi.org/10.1145/3285029, 5:1-5:38.
[47] A. Casali, C. Deco, A. Romano, G. Tomé, An assistant for loading learning object metadata: an ontology based approach, in: Proceedings of the Informing Science and Information Technology Education Conference, Informing Science Institute, 2013, pp. 77–87.