Towards an Adaptive e-Learning System Based on Deep Learner Profile, Machine Learning Approach, and Reinforcement Learning

Riad Mustapha, Gouraguine Soukaina, Qbadou Mohammed, Aoula Es-Sâadia
Mathematics and Computer Science Department, ENSET of Mohammedia, Mohammedia, MOROCCO

***Abstract*—Now-a-days, the great challenge of adaptive e-learning systems is to recommend an individualized learning scenario according to the specific needs of learners. Therefore, the perfect adaptive e-learning system is the one that is based on a deep learner profile to recommend the most appropriate learning objects for that learner. Yet, the majority of existing adaptive e-learning systems do not give high importance to the adequacy of the real learner profile and its update with the one taken into account in the learning path recommendation. In this paper, we proposed an intelligent adaptive e-learning system, based on machine learning and reinforcement learning. The objectives of this system are the creation of a deep profile of a given learner, via the implementation of K-means and linear regression, and the recommendation of adaptive learning paths according to this deep profile, by implementing the Q-learning algorithm. The proposed system is decomposed into three principal modules, data preprocessing module, learner deep profile creation module, and learning path recommendation module. These three modules interact with each other to provide a personalized adaptation according to the learner's deep profile. The results obtained indicate that taking into account the learner's deep profile improves the quality of learning for learners.**

***Keywords*—Adaptive e-learning system; deep learner profile; reinforcement learning; Q-learning; k-means; linear regression; learning path recommendation; learning object**

I. Introduction

In traditional learning systems ("all to all" system), the learning content was determined without taking into consideration the specific needs and characteristics of the learners. As a result, all learners were learning the same learning content, which did not ensure the effectiveness of the learning activity. In the last decade, due to technological developments, various approaches to teaching and learning have come onto the scene, accelerated by e-learning in particular during the covid19 pandemic. However, finding the appropriate learning path and content for a given learner is a very interesting question for achieving learning goals, especially in today's education and teaching systems. It is in this context that intelligent tutoring systems are developed to enable learners to locate educational resources that meet their needs and concerns [1].

Adaptive learning presents a new approach to teaching and learning compared to traditional learning. This new approach allows learners to learn at any time and any place, taking into account the needs and characteristics of learners who are usually heterogeneous to achieve a specific skill within a certain time [2], [3]. In fact, learners have different learning profiles in terms of learning speed, knowledge, preferences, intellectual abilities, learning styles, etc., therefore, have different learning paths [4], [5], [6]. In this perspective, many studies have been realized during the last decade on the personalization of learning with the help of e-learning systems. However, most of these systems do not have methods to perfectly represent the learner's profile (deep profile), on which the system can propose and intelligently adapt the learning path that corresponds to this learner at the time of the execution of a learning activity.

In this paper, on the one hand, we focus on the creation of the deep learner profile from raw datasets on this learner, by combining two machine-learning algorithms: K-means classification and linear regression after executing the data preprocessing technique. On the other hand, the researchers focused on the adaptation and recommendation of learning paths to the learner according to their created deep profile. Next step is the implementation of the developed algorithm Q-learning of the reinforcement learning approach at the time of the execution of a learning activity, Fig. 6, and this via an intelligent and automatic choice of the most appropriate learning objects.

The plan followed in this paper includes, in the second section, a literature review is conducted on adaptive e-learning systems. Then, in the third section, the researchers described the architecture of the proposed approach, while defining the main concepts of the algorithms used to create the deep profile of the learner and to generate the learning path most adapted to this type of profile. The fourth section presents an example of the implementation and results of the application of the algorithms. Finally, the fifth section concludes the present paper and proposes suggestions and perspectives for future work.

II. Related Work

In this section, we have briefly reviewed work related to adaptive systems in e-learning environments and the machine-learning approaches that are used to identify and adapt learning objects to learners.

Recently, adaptive e-learning systems have been frequently developed and used to identify the most appropriate learning objects for learners' profiles. [7-9]. The vast volume of these learning objects presents different opportunities, though, presents constraints for learners to locate the most adequate learning objects for their profiles [2]. The majority of these systems use techniques just based on the learner's learning style and knowledge level to generate a personalized learning context [10]. These techniques are generally underpinned by the Felder-Silverman model [11] to determine this learning style and knowledge level. A few of these systems are discussed in the next paragraph.

Nafea et al. [12] proposed a novel and effective recommender algorithm that recommends personalized learning objects, this algorithm based on the student learning styles. In this system, various similarity metrics are considered in an experimental study to investigate the best similarity metrics to use in a recommender system for learning objects. Vedavathi et al. [13] created a hybrid system that generally uses the learning styles, and knowledge level of the learners to select the relevant learning objects. This system uses a two-step process to generate adaptation via the recommendation of the most appropriate learning objects. First, it categorizes learners based on their learning style, and knowledge level. Then, it looks for learning objects that match his request and that are of interest to similar learners. In the same context, P. Dwivedi et al. [14] and Alshammari et al. [15] have built a similar hybrid adaptation system that clusters learners based on their similarities and proposed the most appropriate learning objects to them. This system is based on learners' history activity, learning styles, and knowledge levels to create learner profiles. Then, it groups learners by using the Nearest Neighbor algorithm (KNN). Consequently, it provides adaptations according to the profile of the group of learners obtained, rather than to individuals.

M. Boussakssou et al. [16] presented an adaptation model based on reinforcement learning. This system takes merely the learning style to adapt and suggest the learning path to the learners' needs. Similarly, H. El Fazazi et al. [17] proposed an adaptive e-learning system design based on the multi-agent system approach and reinforcement learning to recommend an adaptive learning path for a learner with the following profile: intermediate knowledge level, verbal learning style, and hearing impairment. This system tries to recommend a list of learning objects appropriate for this learner profile.

Moreover, W. Intayoad et al. [18] proposed a method based on reinforcement learning, more precisely the State-Action-Reward-State-Action algorithm (SARSA). This method is able to explore the environment to obtain information and exploit it to recommend appropriate learning objects to learners in an e-learning system.

Based on previous studies, it was observed that the proposed learning systems do not have powerful techniques in terms of the quality of learner classification (deep profile creation), which allow to significantly represent the learner and provide the learning system with pertinent information to adapt the learning to the learner's profile. Generally, these systems just consider the learning style and knowledge level of the learner to generate the adaptation.

Therefore, the researchers proposed this system to create a personalized learning experience. That is, it can analyze the learner's learning style, preferences, and abilities, etc. to establish a customized learning path for them. As a result, it can adjust the difficulty level of the content, provide feedback, and offer additional resources based on the learner's performance.

In this study, this system takes into account not only learning style and knowledge level but also other types of profiles (preference profile, knowledge profile, feature profile, etc.), as well as the learner's learning objectives, via the application of machine learning algorithms. This deep profile created will be used to adapt the learning to the specific needs and characteristics of the learner in question, using reinforcement-learning approach. This system will be able to search and select the most appropriate learning objects for this depth profile, thus providing each learner with a learning path that is the most advantageous and adequate.

III. METHODOLOGY

The approach proposed in this paper takes into account the deep learner's profile by using two approaches; namely machine learning and reinforcement learning to intelligently adapt the content of the learning activity to the individual needs of the learners. This latter is done by the recommendation of the most appropriate list of learning objects according to the learner's deep profile. At first, after the preprocessing phase, we used the K-means algorithm and linear regression on the resulting datasets to identify the deep profile of a given learner were used. Then the Q-learning algorithm was applied to generate the learning path of each learner according to his or her deep profile. Our system is composed of three principal modules; data pre-processing module, learner deep profile creation module, and learning path recommendation module. These three modules interact with each other to provide a personalized adaptation according to the learner's deep profile.

A. Overall Process

In this section, the general process of the study was presented in Fig. 1. The important step at the beginning of the pipeline is to initiate the data preprocessing process.

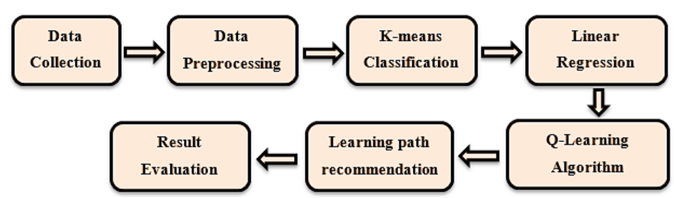

Fig. 1. The overall process for learning path recommendation.

- Data collection: A collection of learner information and characteristics to build a raw dataset containing a volume of pertinent learner information of learners.
- Data preprocessing: This technique consists of removing all redundant, non-pertinent, or less important attributes, extracting the information, and transforming the raw datasets into a useful and efficient format, which allows us to proceed to the next step. The extracted information is relevant and truly represents the learner's deep profile in terms of personal data,

learning style, and prerequisites, etc., and help in their classification. Feature extraction is a difficult and usually time-consuming process.

- K-means classification: Once the datasets are well defined, K-means is used for the classification and generation of homogeneous (similar) learner clusters.
- Linear regression: Once the clusters are generated by K-means, the linear regression algorithm is used to distinguish the different data that generate the same cluster i.e. help K-means to represent the data correctly in each cluster.
- Q-learning algorithm: Once the classification of the learners is completed by using K-means and linear regression, the Q-learning algorithm is used to search and determine the optimal learning path taken by the learner according to his or her deepened profile.
- Learning path recommendation: This step consists of recommending the most appropriate learning objects in real time to the learner based on their deepened profile.
- Result evaluation: The proposed system provides excellent results in terms of precision and quality obtained.

B. Description of the Learning Path Recommendation Process

In examining previous studies on the topic of adaptive e-learning systems development, it has been observed that they are generally based on the Felder-Silverman model (FSLSM) technique [19], to identify the learning style and knowledge level, as the learner profile. However, these types of profiles do not always represent the real learner in terms of specific learner needs and characteristics. To address this need, an intelligent mechanism based on artificial intelligence (AI) was proposed in this study to create the deep profile of a given learner from raw datasets on this learner, by applying in cascade the two machine learning algorithms: K-means and linear regression. Finally, recommending the most appropriate learning objects in real-time to this learner as the most beneficial adaptive learning path to this deep profile via the application of the Q-learning algorithm. The figure below Fig. 2, describes the developed system:

The figure describes the proposed system that integrates Artificial Intelligence (AI) for the creation of the deep profile of the learner connected to the system and for the adaptation of learning paths to the needs and characteristics of learners. In fact, the creation of the deep profile is a very interesting step in the context of recommending adequate learning objects to the learner, because if the created profile does not represent correctly the learner, then the proposed learning path will not be well adapted to him.

In this system, the key modules used are: **1)** the data pre-processing module, which consists of reducing the dimension of the datasets vectors, using techniques associated with dimension reduction, **2)** the module for creating the learner's deep profile from a volume of raw data on the learner in question, and **3)** the learning path recommendation module. These three modules interact with each other to provide personalized adaptation based on the deep profile created in the Module 2 phase. This system will try to find the appropriate learning objects for this deepened profile.

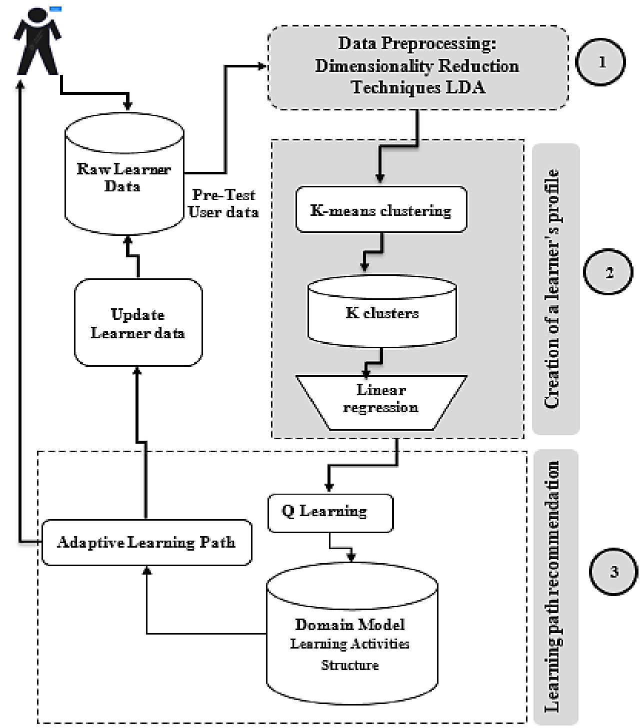

Fig. 2. The process of constructing the adaptive learning path developed.

A. Data Preprocessing

In a datasets, not all attributes are necessarily important. Some may be redundant, others may be not pertinent, etc. Ignoring or removing these non-pertinent or less important attributes reduces the load on the machine learning algorithms and improves the quality of the results obtained. In this context, the technique of dimension reduction plays a very interesting role, in reducing the dimension of high-dimensional datasets vectors [20]. Among these techniques, the best known according to the literature is Linear Discriminant Analysis (LDA), as a supervised algorithm. The aim is to project the features of a higher-dimensional space into a lower-dimensional space. This technique is based on two criteria to create another axis on which to project:

1) Maximize the distance between the means of the two classes;

2) Minimize the variation within each class;

Assuming two classes and d-dimensional elements such as x_1, x_2.., x_n, where: n1 is an individual from class C_1 and n_2 from class C_2. If xi is the data point, then its projection on the line represented by the unit vector v can be written as $v^T xi$:

$$\breve{u}_1 = \frac{1}{n_1}\sum_{xi \in C1}^{n1} v^T x_i = v^T u_1 \qquad (1)$$

The same goes for $\breve{u}_2$,

$$\breve{u}_2 = v^T u_2 \quad (2)$$

Then the dispersion for the elements of C1 is:

$$\tilde{s}_1^2 = \sum_{yi\in C1}(y_i - u_1)^2 \quad (3)$$

The same goes for $\tilde{s}_2^2$,

$$\tilde{s}_2^2 = \sum_{yi\in C2}(y_i - u_2)^2 \quad (4)$$

B. Creating the Deep Learner Profile

The adaptive e-learning system has been developed to provide a personalized learning path in the experience of executing a pedagogical activity. However, this adaptation must be based on the learner's deep profile that is created when the learner as soon as it is connected to the system, and that will be updated over time by the learner's feedback with the system. The use of deep profiles in Computer Environments for Human Learning (CEHL), and especially in distance learning platforms, is one of the most important ways to adapt learning to the specificities of learners. It plays a very interesting role in the individualization of learning paths. In fact, it focuses on learners' preferences, such as a learner's preference for video over text, or his or her preference for teaching to begin with an example and not a theoretical introduction. But also the features of the learners which contain important information allowing to describe the learner not only from the point of view of his demographic features such as age or gender but also from the point of view of his possible disabilities (e.g. hearing 85%; sight 30%, etc.), preferred time for learning, preferred language, culture, country, hobbies, preferred type of media, prerequisites, the learner's learning objectives, etc. which can be described finely.

The module in charge of creating the learner's deep profile within the system is decomposed into two sub-modules:

- The K-means classification algorithm;
- The Linear regression algorithm.

1) K-means classification algorithm: In the literature, there are many classification algorithms including K-means, KNN, SVM, etc. In this paper, the K-means Algorithm, which is one of the most popular algorithms due to its simplicity and intuitive interpretation [21] was adopted. It can be defined as the process of organizing objects in a dataset into clusters, such that objects in the same cluster have a high degree of similarity, while those belonging to different groups have a high degree of dissimilarity.

The key step for any unsupervised algorithm is to identify the optimal number of clusters (optimal K) into which the data can be grouped. The Elbow method [22], is one of the most popular methods for identifying this optimal value of K. i.e. the optimal k is the point after which the distortion/inertia starts to decrease linearly, where distortion is computed as the average of the squared distances of the cluster centers of the sample and inertia is the sum of the squared distances of the elements to their nearest center of gravity. It consists in calculating the variance of the different cluster volumes considered, and then placing the variances obtained on a graph:

$$intra - Cluster\ Variance = \sum\sum Distance\ (x, centroid)^2 \quad (5)$$

After calculating the optimal value of k, by the formula (5), we proceed to the running of the K-means algorithm, explained by the mathematical formula below:

$$\arg\min_{C}\sum_{i=1}^{k}\sum_{x\in Ci}\left||x - \mu i|\right|^2 = \arg\min_{C}\sum_{i=1}^{k}|C_i|\ \mathrm{Var}\ C_i$$

Where µi s is the average of the points in C_i. For its implementation, in the system, the following steps are respected:

- Step 0: Select optimal K calculated by the formula (5);
- Step 1: Select K random points in datasets as initial group centers;
- Step 2: Create K Clusters by associating each data point with its nearest center, according to the Euclidean distance defined by the formula:

$$d(x,y) = \sum_{i=1}^{n}(xi - yi)^2 \quad (7)$$

- Step 3: Recalculate the center of gravity of each cluster, as the average of all data points in that cluster;
- Step 4: Repeat steps 2 and 3 until the centers of gravity no longer change;

By examining the results of previous studies on the subject of data classification, the researchers find that after the execution of the K-means algorithm on the datasets, there is a problem in the distributions of the vectors of each cluster, i.e. different data can be represented by similar clusters, which results in a false classification of the data. In this paper and to overcome this problem, the researchers suggest to add another criterion to distinguish the different data that give the same cluster, by using in cascade the linear regression algorithm on each cluster obtained by K-means, to approximate the data and characteristics that make up the learner's deep profile, to improve the quality of classification, precision and error reduction.

2) Linear regression algorithm: In the literature, linear regression is classified among the multivariate analysis methods that deal with quantitative data, with the objective to find a linear relationship between a quantitative variable Y and one or more also quantitative variables X, i.e. to find the line that passes "as close as possible" to all the points of the cloud. This relationship can be expressed mathematically by the formula below:

$$yi = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} + \varepsilon_i = X_i^T\beta + \varepsilon_i \quad (8)$$

Where T denotes the transposition, so that $X_i^T\beta$ is the internal product between the vectors Xi and β.

C. Learning Path Recommendation

1) Domain model of the proposed approach: In the adaptive e-learning system, the pedagogical content is divided into many pedagogical activities, each of which is composed of several chapters. A concept tree represents each chapter. The concept tree contains a list of learning objects (LOs) that represent external representations, such as images, videos, examples, exercises, etc. This learning content is generally stocked in a database as a tree of 4 levels called the domain model. The figure below Fig. 3, describes the overall domain model structure of our system:

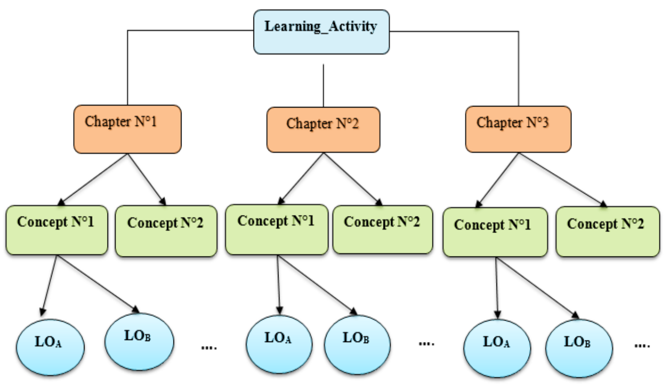

Fig. 3. Learning activity organization of the proposed approach.

The learning activity organization of the proposed approach is mapped into four hierarchical levels:

- Level 1: represents the root of the domain model structure, which presents the learning activity to be taught.
- Level 2: represents learning activity chapters, which particularly address a learning activity element to be taught.
- Level 3: represents the concepts of each chapter of the learning activity taught and is generally identified by a level: advanced, intermediate, or beginner.
- Level 4: represents the different types of learning objects (LOs) that represent external representations, such as images, videos, examples, exercises, etc. Each LO is characterized by several properties such as autonomy; adaptation; indexing; accessibility; durability, and reusability as the most important property. Each LO must be provided with a description, usually called a metadata file, allowing it to be easily found. This file uses emerging technologies associated with the development of learning objects, such as the semantic web and ontologies.

The figure below Fig. 4, describes the metadata file of each domain model level of the proposed system, based on the ontology technology:

In a traditional learning system, learners use a linear path of learning objects {LO_A, LO_B, LO_C.., LOz} regardless of their preferences, knowledge level, etc. The figure below Fig. 5, shows an example of the sequence of learning objects as a learning path for five learners {L_1, L_2, L_3, L_4, L_5} when executing a learning activity in this system.

However, in a personalized adaptive e-learning system, learners use a non-linear learning path to build the optimal sequence of learning objects according to their needs and characteristics. The figure below Fig. 6, shows the adaptive learning path for the same learners {L_1, L_2, L_3, L_4, L_5} when running the same pelagic activity.

In this learning path, the system can ignore some learning objects like LO_E, LO_F, $_{LOG,}$ and LO_H, since they do not correspond to any profile of the five learners in question.

2) Adaptation and recommendation of learning paths: Many algorithms are available for Reinforcement Learning (RL), which uses Q-function as a learning strategy like Opponent Modeling, Q-learning, and Ascending Gradient. In this paper, the Q-learning algorithm was adopted to recommend the most appropriate learning path for a given learner. It is more efficient than other algorithms in terms of precision and quality of results. Thus, it converges towards an optimal strategy, i.e. it leads to maximizing the total reward of the successive steps. The figure below Fig. 7, presents the overall process of reinforcement learning.

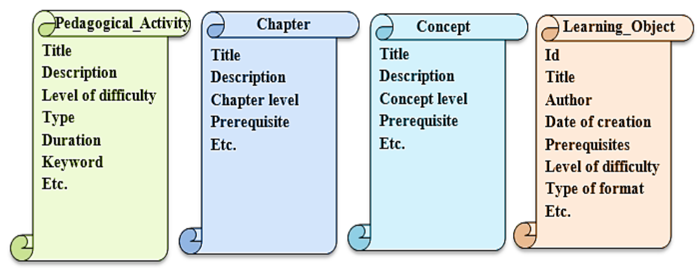

Fig. 4. Metadata file of the domain model levels.

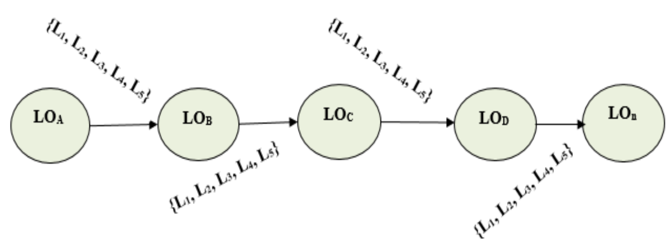

Fig. 5. Learning path in a traditional system.

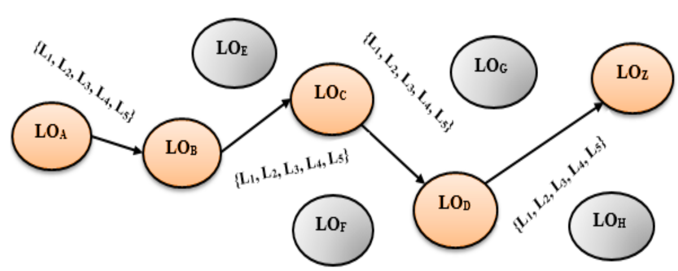

Fig. 6. Learning paths in an adaptive e-learning system.

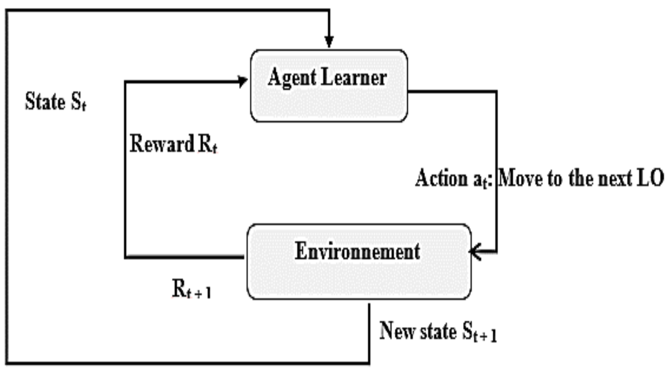

Fig. 7. The overall process of reinforcement learning.

In the proposed system, the high number of users will allow us to consider each learner as an agent. The agent can move from one state S (LO_A) to another St+1 (LO_B) by choosing an action A. This transition gives the agent a reward/sanction on which we calculate the total gain to define its optimal adaptive learning path. This gain is estimated and calculated by a Q-Value () which evaluates the quality of the combination of each state-action pair Q(s, a) over the long term and updates its table. Mathematically the Q-Value() function is expressed as follows.

$$V(S) = [R_{t+1} + \gamma\, R_{t+2} + \gamma^2\, R_{t+3} + \cdots] \left[\sum_{k=0}^{\infty} \gamma^k\, R_{t+k+1} | S_t \;= s\right]$$

Where the parameter γ is the discount rate between 0 and 1, k is the number of states, and R is the rewards. The agent learns from experience due to exploration, often called an episode [23]. The figure below Fig. 8, presents the overall process of learning by the Q-learning algorithm developed:

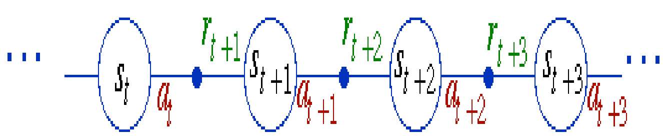

Fig. 8. Overall process of learning by the Q-learning algorithm.

The general algorithmic structure of this algorithm is detailed in Fig. 9.

Where α is the learning rate between 0 and 1. It determines how well the new calculated information will overcome the old, If α = 0, the agent (learner) learns nothing, but if α = 1, the agent still ignores everything it has learned so far and will only take into account the last information. In a deterministic environment, the learning speed $\alpha_t(s,a) = 1$ is optimal; γ is the discount factor between 0 and 1. It determines the importance of future rewards. A factor γ = 0 would make the agent (learner) myopic by considering only current rewards, but a factor γ near 1 would also include more distant rewards and the value of Q can diverge; s' is the new state; s is the previous state; a is the chosen action and R is the reward received by the agent (learner), and Q [v_1, v_2,.., v_p] is Q-table.

Algorithm 1: Heading

```
Input
    α ;
    γ ;
Output
    Q [v1, v2, ..., vp];
Initialize
      Q [s, a] for any non-terminal state s, any action a
      arbitrarily;
      Q [terminal state, a] = 0;
Repeat
  | // start of an episode
  | s : = terminal state
  |   Repeat
  |   // for each episode
  |    Choose an action a from s using the policy specified
  |    by Q;
  |    Execute the action a;
  |    Observe the reward r and the new state s' ;
  |    Update the values of Q
  |     | Q(s,a) =  α * (r + γ * maxQ(s') - Q(s,a))
  |     |  Define the next state as the current
  | Until s is the terminal state
  | End
End
```

Fig. 9. Q-function pseudo-code for e-learning adaptation and recommendation system.

IV. Results and Discussion

A. Implementation Example

In this part, the researchers start with a data preprocessing module to prepare datasets which will be used to create a deep profile of the learner connected to the system. By doing this, the deep profile is processed by using the developed Q-learning algorithm to find the ideal learning path for this learner. The experiments are conducted on a Dell computer with a CORE i5 processor using the Collaboratory platform and Python for result generation.

1) Data preprocessing: The datasets used in the experiments contain different raw information about 100 learners {L_0, L_1,.., L_{99}}, with 23 characteristics {Age, Gender, Handicap, Country, Language, Learning style; Knowledge level; etc.} numbered from 0 to 22. The specifications of the datasets are summarized in the figure below, Fig. 10.

In the first experiment, the researchers start to reduce the dimension of the vectors of the datasets, by ignoring the less important or unimportant attributes to improve the quality of the results obtained, by applying the LDA technique, while focusing on the criteria of "Minimizing the variation" within each cluster. Firstly. the variance of all the attributes of the datasets is calculated using the following formula:

$$V = \frac{1}{n} \sum_i^n (x - \overline{x})^2 \tag{9}$$

Console IPython

Console 2/A

	Age	Gender	Handicap	Culture	...	Obj2N	Obj3	Obj3F	Obj3N
0	18	1	1	2	...	2	8	5	3
1	19	1	1	3	...	17	8	5	2
2	22	2	0	1	...	5	8	3	9
3	20	1	1	2	...	14	7	2	14
4	22	2	1	1	...	7	7	3	17
5	19	1	1	1	...	0	9	1	11
6	22	1	0	2	...	2	8	3	11
7	16	1	0	4	...	16	9	3	3
8	17	1	0	2	...	10	7	4	2
9	22	1	1	1	...	7	9	3	0
10	18	1	0	2	...	6	8	3	17
11	18	1	0	3	...	8	7	1	6
12	21	1	1	1	...	17	7	3	8
13	18	2	1	2	...	14	9	4	0
14	19	1	1	3	...	16	9	3	18
15	19	2	1	3	...	6	9	5	15
16	22	1	0	4	...	12	8	5	7
17	22	1	1	3	...	10	9	1	16
18	21	1	1	2	...	0	8	5	5
19	19	2	1	1	...	10	9	5	14
20	18	1	1	2	...	0	7	5	13
21	18	2	1	1	...	4	7	3	10
22	18	1	0	1	...	8	7	4	8
23	17	1	1	3	...	14	9	5	2
24	21	2	0	3	...	20	9	1	16
25	20	2	1	1	...	20	9	4	19
26	22	2	0	3	...	9	7	1	5
27	21	2	0	3	...	7	8	5	20
28	19	1	0	4	...	5	8	4	10
29	20	1	1	4	...	1	8	2	15
..	...	...	...	...	...	...	...	...	...
70	21	1	0	1	...	10	7	5	12
71	17	2	0	2	...	10	9	3	6
72	22	2	1	3	...	15	7	4	10
73	19	2	0	3	...	2	7	4	11

Fig. 10. Raw Datasets of learners.

Then, the researchers try to find the attribute whose variance is the smallest compared to the others (the less significant attribute). The figure below, Fig. 11, shows the result of calculating the variance of all the attributes of the datasets:

```
[ 3.83545455  0.24434343  0.25242424  1.3069697  11.92515152  0.67232323
  4.91909091  0.25090909  0.25        0.25090909  0.25090909  0.64
  2.19555556  2.01373737  0.79707071  1.87585859 31.29292929  0.64
  1.92515152 33.90494949  0.64434343  2.22222222 43.73848485]
```

Fig. 11. Result of variance calculation.

By examining the figure above, the researchers notice that the minimum variance is 0.24434343. This variance corresponds to the 2^{nd} attribute of the datasets whose index is 1 (gender). In fact, it was imperative to delete this attribute because it is not significant in terms of importance. To evaluate the performance of the approach, the datasets must be divided into two sections: the training set (80%) and the test set (20%)

B. *Creating a Learner Deep Profile*

1) Application of K-means Algorithm: In the second experiment, and once the data preprocessing step is over, the researchers try to classify all learners into homogeneous (similar) groups. Initially, a sample of 100 learners {L_0, L_1,.., L_{99}} was used. In this case, the system categorized the learners into three clusters (0, 1, and 2). The figure below, Fig. 12, shows the list of learners in each Cluster.

Console 15/A

```
Learners liste of cluster 0 :

[1, 3, 6, 7, 11, 12, 16, 19, 21, 22, 23, 29, 30, 31, 32, 36, 37, 50, 53, 56, 57, 60, 65, 74, 78, 84, 89,
95, 96]

Learners liste of cluster 1 :

[2, 8, 9, 10, 13, 14, 17, 24, 27, 33, 34, 35, 39, 40, 41, 42, 44, 48, 54, 59, 61, 63, 66, 67, 69, 72,
76, 77, 87, 88, 90, 93, 94]

Learners liste of cluster 2 :

[4, 5, 15, 18, 20, 25, 26, 28, 38, 43, 45, 46, 47, 49, 51, 52, 55, 58, 62, 64, 68, 70, 71, 73, 75, 79,
80, 81, 82, 83, 85, 86, 91, 92, 97, 98, 99, 100]
```

Fig. 12. Lists of learners in each group.

The results in the form of a 2D histogram graph are presented in the figure below, Fig. 13 as follows: we have 33 learners in Cluster 0 and 38 learners in Cluster 1, and 29 learners in Cluster 2:

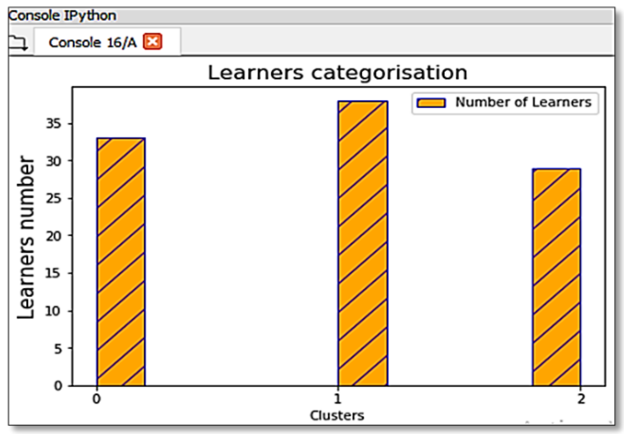

Fig. 13. Classification of learners into cluster.

The results in the form of a 3D sector graph are presented in the figure below, Fig. 14 and as follows: we have 33% learners in Cluster 0 and 38% learners in Cluster 1 and, 29% learners in Cluster 2.

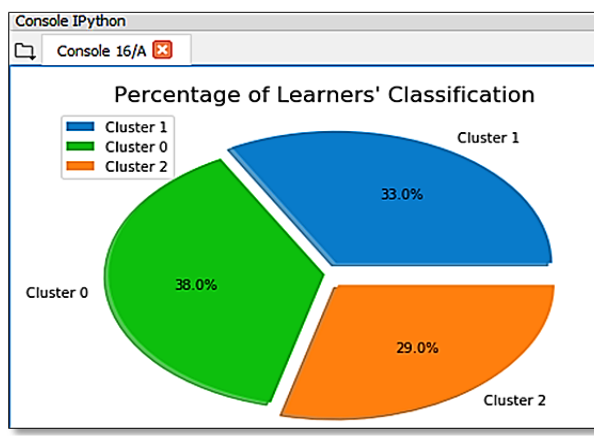

Fig. 14. Classification of learners into clusters.

In the third experiment, the results of classifying learners concerning the two attributes age and motivation for each cluster are presented in the figure below, Fig. 15:

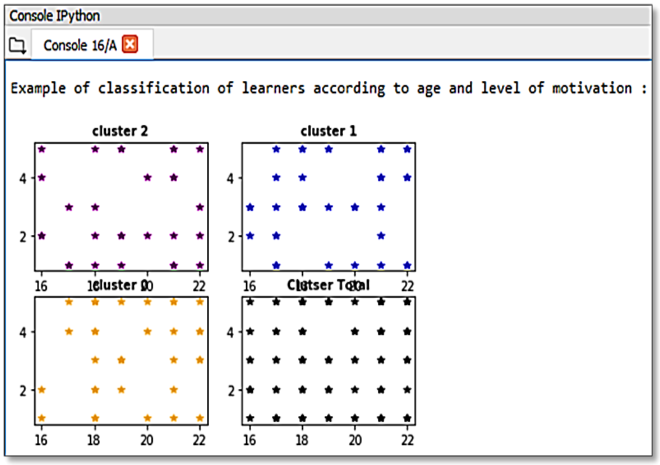

Fig. 15. Classification of learners according to age and motivation level.

The results above show the distribution of learners in the three clusters (Cluster 0, Cluster 1, and Cluster 2) according to age and the degree of motivation among learners. While in the cluster, whose name is “All_Cluster” presents the union of the three clusters. To identify the learners in each cluster to reduce the error, the researchers propose to run the linear regression algorithm in cascade on the clusters generated by the K-mean classification algorithm.

2) Application of the linear regression algorithm: By examining the results obtained previously on the classification of the learners according to their age and their degree of motivation, via the application of the K-means algorithm, and to improve the quality of classification, the performance of calculations, the precision, and the reduction of the error, the researchers propose to apply in cascade the linear regression algorithm on each cluster to identify it well to distinguish it compared to the other clusters First, the researchers start to calculate the slope and the intercept of each cluster. The results are presented in the figure below, Fig. 16.

These values allow us to draw linear regression lines for each cluster. This method allows us to identify the learners of each cluster and to build homogeneous clusters in terms of the classification quality of the learners according to their deep profiles. The figure below, Fig. 17, graphically presents a comparison of the linear regression of the three clusters:

The results in the Fig. 17 show that there is a relationship between the “age” and the “degree of motivation” of learners during the execution of an educational activity. Where the researchers notice that the degree of motivation of the learners is remarkable when the age > = 15 the level of motivation of the learners equals 2.5 for cluster 0 and cluster 2, on the other hand, the level of motivation of the learners equals 3 for cluster 1. Moreover, when the age = 25, the motivation among the learners reaches the peak: 4 for Cluster 1 and 3.8 for Group 0, and 2.8 for Cluster 2.

C. Learning Path Recommendation

In this example, the researchers show in the form of a graph Fig. 18 how to define an adaptive e-learning activity for a given learner, using "Bloom's taxonomy" [16], [24], [25] the researchers give a list of states (LOs) that describe the learning activity: Introduction; Chapter N° 1; Examples, Exercises, and Exam. Bloom's taxonomy" helps us to give each transition a reward (at the basis of a scale from 0 to 10) according to its difficulty level, where each level depends on the previous one:

Stay = + 00; Next_level = + 0,5; High_level = + 0,5; Exercising = + 0,4; Look_Examlpes = + 0,2; Previous = + 0,1 and Passing_Quiz = + 0,3.

```
Console IPython
Console 2/A

Regression of the cluster 0

slope 0: [[0.11745573]]
intercept 0: [0.86375136]

Regression of the cluster 1

slope 1: [[0.07981097]]
intercept 1: [1.68836965]

Regression of the cluster 2

slope 2: [[0.0499114]]
intercept 2: [1.90342587]
```

Fig. 16. Slope calculation and cluster interception.

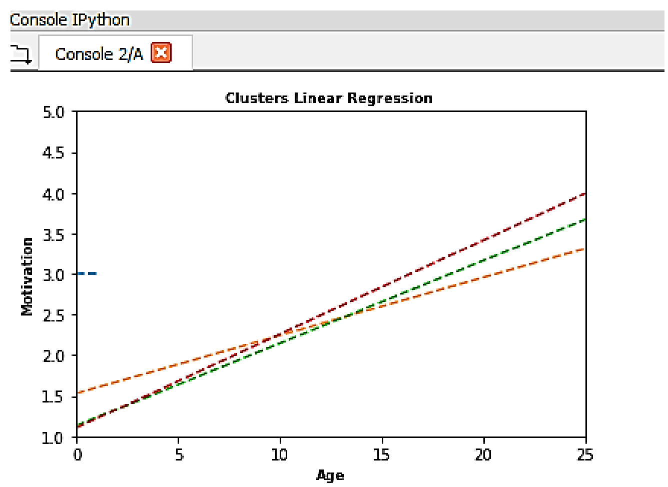

Fig. 17. Cluster linear regression.

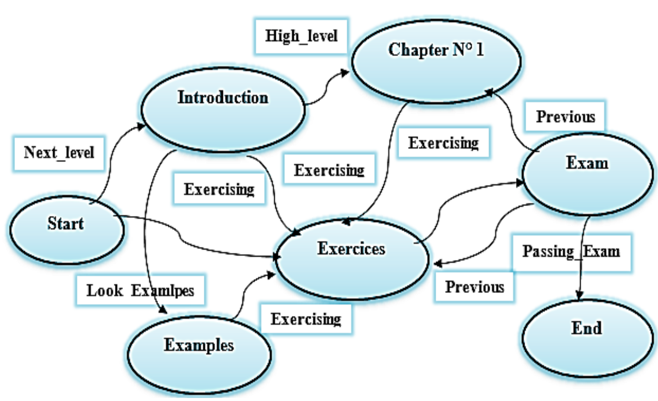

Fig. 18. Example of a state-action combination graph for the execution of an educational activity.

Applying "Bloom's taxonomy" to our example above, the researchers deduce the reward matrix of state-action pairs on the set of related (directly connected) states:

$$Mij \begin{pmatrix} +00 & +0,5 & +00 & +00 & +0,4 & +00 \\ +00 & +00 & +0,5 & +0,2 & +0,4 & +00 \\ +00 & +00 & +00 & +00 & +0,4 & +00 \\ +00 & +00 & +00 & +00 & +0,4 & +00 \\ +00 & +00 & +00 & +00 & +00 & +0,3 \\ +00 & +00 & +0,1 & +00 & +0,1 & +00 \end{pmatrix}$$

Reward matrix of state-action pairs

Where, the rows i of the matrix M_{ij} represent the states, while the columns j of the matrix M_{ij} represent the actions. To determine the optimal adaptive learning path to be taken by the learner for this learning activity cited in Fig. 17, based on their deep profile, the researchers must first generate the Q-value of the Q-learning algorithm as follows:

```
1 [0 ; 1,157229998544553 ; 0 ; 0 ; 0,0999999999991 ; 0]
2 [0 ; 0, 0,98899999588963 ; 0,6699999971061 ; 0,6999999955544 ; 0,29999999997890]
3 [0 ; 0 ; 0 ; 0,626999999999999 ; 0 ; 0,299999999999999]
4 [0 ; 0 ; 0 ; 0 ; 0,6999999999999 ; 0]
5 [0 ; 0 ; 0 ; 0 ; 0 ; 0,299999999999999]
6 [0 ; 0 ; 0 ; 0 ; 0 ; 0]
```

Fig. 19. Q-table matrix of the Q-value function for the action-state.

The matrix above presents the results obtained by running the Q-learning algorithm on our system, where rows i of the Q-table represent states, while columns j represent actions. They show how the Q-learning algorithm proposes the most optimal learning path for a given learner. The learner can choose to start with any learning object at the time of the learning activity, after which the system will recommend the best learning path, by choosing and organizing the sequence of learning objects appropriate in real time to their deeper profile.

If the agent (the learner) were to use the policy described in the Q-table above to find the most appropriate learning path for her deep profile, then:

- From state zero (Start), the action with the maximum value is Next_level, so that's what he will do, achieving state one (learning object Introduction).
- From state one (Introduction), the action with the maximum value is High_level, This leads the learner to state two (learning object Chapter N°1).
- From state two (Chapter N°1), the action with the maximum value is High_level, This leads the learner to state four (learning object Exercises).
- From state four (Exercises), the action with the maximum value is Passing_Exam, This leads the learner to state five (learning object Exam).

Based on these transitions, we can deduce the adaptive learning path for this learner:

Start ➔ Introduction ➔ Chapter N° 1 ➔ Exercises ➔ Exam➔ End.

V. Conclusion

Deep learner profiles are very interesting objects that can contribute to the success of adaptive e-learning systems. These deep profiles must be able to contain different types of information and characteristics about the learner, to take into account the different facets of their learning. Therefore, generic models are needed to properly represent learners. To address this need, the researchers have presented in this paper a new intelligent approach based on K-means, linear regression, and algorithms Q-learning. The system is designed to create the most detailed and best structured in-depth profile for a given learner, and then recommend the most appropriate learning path for that learner. The proposed system ensures that it can be used by various learning management systems (LMS) in these various contexts.

As a perspective, on the one hand, the researchers plan to propose a technique for the choice of optimal k in the learner classification phase and on the other hand to experiment with the approach while adding as many states and actions as possible for each state (learning object) to allow the discovery of the optimal action for each learner. This can of course lead to complexity and convergence constraints, especially if the system is to be put online. For this, the use of deep reinforcement learning (Deep Q-learning) will be a good idea for issues of complexity and optimization of the system.

This study has potential limitations, since Reinforcement learning has various drawbacks while appearing to be a very potent and useful technique, it needs to store values for each state; it frequently uses too much RAM. That is, it might become a memory-intensive process. Moreover, due to the small sample size, the results may not be representative of the entire population. Therefore, it may limit the validity and generalizability of the findings. A larger sample size would be needed to increase the statistical power and improve the representativeness of the sample since there would be the chance of having different deep profiles

References

[1] M. Laaziri, S. Khoulji, K. Benmoussa, and K. M. Larbi, "Outlining an Intelligent Tutoring System for a University Cooperation Information System," Engineering, Technology & Applied Science Research, vol. 8, no. 5, pp. 3427–3431, Oct. 2018, https://doi.org/10.48084/etasr.2158.

[2] Manal Abdullah, Reem M. Bashmail, Wafaa H. Daffa, Mona Alzahrani and Malak Sadik, The Impact of Learning Styles on Learner's Performance in E-Learning Environment. (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 6, No. 9, 2015.

[3] H.M. Truong, : Integrating learning styles and adaptive e-learning system: current developments, problems and opportunities, Comput. Hum. Behav., vol. 55, pp. 1193 (2016).

[4] Hoang Tieu Binh and Bui The Duy.: Predicting Students' performance based on Learning Style by using Artificial Neural Networks. IEEE International Conference on Knowledge and Systems Engineering(KSE) (2017).

[5] H. Imran, M. Belghis-Zadeh, T. Chang, T.S Graf. PLORS: a personalized learning object recommender system. Vietnam J.Comput. Sci. vol. 3, no. 1, pp. 3–13, 2016, doi: 10.1007/s40595-015-0049-6.

[6] Manal Abdulaziz Abdullah.: Learning style classification based on student's behavior in moodle learning management system. Transactions on Machine Learning and Artificial Intelligence, 3(1):28 (2015) 045815.

[7] Madani, Y., Bengourram, J., Erritali, M., Hssina, B., & Birjali, M. (2017). Adaptive e-learning using genetic algorithm and sentiments

analysis in a big data system. International Journal of Advanced Computer Science And Applications, 8(8), 394403.

[8] Rajesh C. Panicker, Akash Kumar, Dipti Srinivasan and Deepu John. Adaptive Learning and Analytics in Engineering Education; 2018 IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE). 978-1-5386-6522-0/18/$31.00 ©2018 IEEE 4-7 December 2018, Wollongong, NSW, Australia.

[9] Outmane Bourkoukou, and Essaid El Bachari . A Big-Data Oriented Recommendation Method in E-Learning Environment. Article in International Journal of Emerging Technologies in Learning (IJET) · June 2022 . doi: 10.3991/ijet.v17i10.27861.

[10] M. T. Alshammari and A. Qtaish, "Effective Adaptive E-Learning Systems According to Learning Style and Knowledge Level," Journal of Information Technology Education: Research, vol. 18, pp. 529–547,Nov. 2019.

[11] Ikawati, Y., Al Rasyid, M. U. H., & Winarno, I. (2020, September). Student behavior analysis to detect learning styles in Moodle learning management system. In 2020 International Electronics Symposium (IES) (pp. 501-506). IEEE. doi: 10.1109/IES50839.2020.9231567.

[12] Nafea, S. M., Siewe, F., & He, Y. (2019, February). A novel algorithm for course learning object recommendation based on student learning styles. In 2019 International Conference on Innovative Trends in Computer Engineering (ITCE) (pp. 192-201). IEEE. doi: 10.1109/ITCE.2019.8646355.

[13] N. Vedavathi, K.M. Anil. An efficient e-learning recommendation system for user preferences using hybrid optimization algorithm. Soft Comput, 25, 9377–9388, 2021, doi: 10.1007/s00500-021-05753-x.

[14] Dwivedi, P., Kant, V., & Bharadwaj, K. K. (2018). Learning path recommendation based on modified variable length genetic algorithm. Education and information technologies, 23, 819-836.

[15] M. T. Alshammari and A. Qtaish, "Effective Adaptive E-Learning Systems According to Learning Style and Knowledge Level," Journal of Information Technology Education: Research, vol. 18, pp. 529–547, Nov. 2019.

[16] M. Boussakssou, B. Hssina, and M. Erittali, "Towards an Adaptive Elearning System Based on Q-Learning Algorithm," Procedia Computer Science, vol. 170, pp. 1198–1203, Jan. 2020, https://doi.org/10.1016/j.procs.2020.03.028.

[17] H. El Fazazi, M. Elgarej, M. Qbadou, and K. Mansouri, "Design of an Adaptive e-Learning System based on Multi-Agent Approach and Reinforcement Learning", Engineering, Technology & Applied Science Research (Eng. Technol. Appl. Sci. Res.), vol. 11, no. 1, pp. 6637–6644, Feb. 2021.

[18] W. Intayoad, C. Kamyod, and P. Temdee, "Reinforcement Learning for Online Learning Recommendation System," in 2018 Global Wireless Summit (GWS), Chiang Rai, Thailand, Nov. 2018, pp. 167–170, https://doi.org/10.1109/GWS.2018.8686513.

[19] Graf, S., Viola, S. R., & Kinshuk, T. L. (2006, December). Representative characteristics of felder-silverman learning styles: An empirical model. In Proceedings of the IADIS International Conference on Cognition and Exploratory Learning in Digital Age (CELDA 2006), Barcelona, Spain (pp. 235-242).

[20] G. Thippa Reddy M. ; Praveen Kumar Reddy; Kuruva Lakshmanna; Rajesh Kaluri; Dharmendra Singh Rajput; Gautam Srivastava; Thar Baker "Analysis of Dimensionality Reduction Techniques on Big Data". IEEE, 16 March 2020, doi: 10.1109/ACCESS.2020.2980942.

[21] B. K. Ponukumati, P. Sinha, M. K. Maharana, A. V. P. Kumar, and A. Karthik, "An Intelligent Fault Detection and Classification Scheme for Distribution Lines Using Machine Learning", Eng. Technol. Appl. Sci. Res., vol. 12, no. 4, pp. 8972–8977, Aug. 2022.

[22] Cui, M. (2020). Introduction to the k-means clustering algorithm based on the elbow method. Accounting, Auditing and Finance, 1(1), 5-8. doi: 10.23977/accaf.2020.010102.

[23] Sutton, R. S., & Barto, A. G. (1998). Introduction to reinforcement learning (Vol. 2, No. 4). Cambridge: MIT press.

[24] Krathwohl, D. R. (2002). A revision of Bloom's taxonomy: An overview. Theory into practice, 41(4), 212-218. Doi:10.1207/s15430421tip4104_2.

[25] Anderson, L. W., & Krathwohl, D. R. (2021). A taxonomy for learning, teaching, and assessing: A revision of Bloom's taxonomy of educational objectives. Longman.