Enhancing College Education: An AI-driven Adaptive Learning Platform (ALP) for Customized Course Experiences

Hanan Altaleb[1]

Samar Mouti[2]

Shajeela Beegom[3]

Information Technology Department
Khawarizmi International College
Abu Dhabi, United Arab Emirates

K-919660205@khawarizmi.com[1]

samarmouti@khawarizmi.com[2]

shajeela.beegom@khawarizmi.com[3]

2023 9th International Conference on Optimization and Applications (ICOA) | 979-8-3503-1254-6/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICOA58279.2023.10308834

Abstract—The main aim of this paper is to design an Adaptive Learning Platform (ALP) based on Artificial intelligence (AI) algorithm for students taking college courses. In this platform, course content and methods are customized to meet the needs of individual students based on AI algorithms. ALP aims to provide efficient, effective, and customized course learning paths based on student level. This platform enhances student performance in the course learning outcome by adjusting the instructional content, pace, and delivery methods to the learner's individual needs and preferences. ALP uses Reinforcement Learning (RL) algorithm and data analytics techniques to analyze the student's learning behaviors, strengths, and weaknesses, and provide personalized recommendations on how to improve their learning outcomes.

Keywords—Adaptive Platform, Artificial intelligence, Reinforcement Learning

I. INTRODUCTION

Adaptive Learning Platform (ALP) provides a personalized and engaging educational platform by adapting to individual learners, fostering engagement, and utilizing data-driven insights, the ALP enhances learning outcomes and empowers learners to achieve their full potential. ALP offers personalized learning to each user. The platform analyzes the unique characteristics and learning styles of individual learners, allowing it to deliver customized content, activities, and assessments that cater to their specific needs and goals. ALP improves learning outcomes by providing personalized feedback and guidance, the platform will support learners in their educational journey. It will track their progress, identify areas where they excel or struggle, and offer targeted resources and recommendations to help them succeed. Data-driven insights play a crucial role in the ALP project by analyzing learner data, the platform will provide valuable insights into their progress, preferences, and learning patterns. These insights will inform the ongoing development and improvement of the platform, helping to make it more effective and relevant.

II. LITERATURE Review

Recent challenges are addressed and proposed potential directions for developing effective knowledge assessment techniques in future adaptive learning ecosystems [1]. A plan and fabrication approach of adaptable learning is presented, which includes customized accessibility provision, learning multiple objects given with patterns, and retaining the information for every learner for recurring learning. The authors also analyze and adjust the archetype of a learning management system (LMS) to meet individual prerequisites and respond to exploration requests. Personalized learning is proposed for evaluation, where applications execute the personalization systems [2]. The research highlights the scarcity of didactic literature on adaptive learning and emphasizes the necessity to integrate domain expert knowledge. This review complements theoretical evaluations in the analytics literature of this field. While previous theoretical analyses focused primarily on methodological challenges, techniques, and a relatively narrower application of adaptive learning, this study aims to expand the application areas by employing diverse implementation techniques and conducting a qualitative appraisal of literature, along with a systematic selection and analysis of research studies [3]. The research concludes, through statistical analysis and data representation techniques, that adaptive learning systems must consider affective factors in addition to learners' cognitive abilities. The alignment between various components in adaptive systems can influence how learners access the system and, more importantly, their performance. The research also emphasizes the importance of visualization in revealing interesting discoveries that might otherwise be overlooked [4]. The concept presented is to offer courses that align with the way learners can successfully complete their learning journey. To achieve this objective, the model is realized via an adaptive e-learning system that utilizes the traces left by users' interactions with their learning environment. Depending on the learners' profile, the system automatically determines the path and recommends the appropriate courses using the ant colony algorithm [5]. The system utilizes multimodal sensor data and machine learning to identify three affective states (engagement, frustration,

979-8-3503-1254-6/23/$31.00 ©2023 IEEE

Authorized licensed use limited to: b-on: UNIVERSIDADE DO PORTO. Downloaded on November 03,2024 at 17:24:36 UTC from IEEE Xplore. Restrictions apply.

boredom) linked to learning and then determines how to present learning content to maintain the learner in an optimal affective state and maximize the rate of learning. The findings suggest that when learning activities are tailored to the learner's emotional state, it increases their engagement, which in turn promotes learning. However, longer exposure is needed to determine the effect of the system on learning [6]. The effectiveness of the proposed system is assessed through extensive experiments utilizing modified models of ResNets and MobileNetV2 on CIFAR-100 and ImageNet datasets. The outcomes indicate that the proposed distributed model has increased accuracy and reduced energy consumption in comparison to standard models, indicating its adaptability [7]. The method demonstrated superior prediction accuracy over ordinary random forest models and six other machine-learning-based regression models on the acquired throughput data. The improved throughput prediction accuracy of the method RF-LS-BPT demonstrates the significance of hyperparameter tuning in developing precise and reliable machine-learning-based regression models. The model has potential applications in throughput estimation and modeling in 5G and beyond 5G wireless communication systems [8]. The method integrates various tasks, including feature learning, attribute prediction, relation mining, image matching, image understanding, image summarization, and label expansion, into a single unified framework. Through extensive qualitative and quantitative experiments, the results demonstrate that the proposed method outperforms several state-of-the-art techniques [9-11].

III. Design of an Adaptive Learning Platform (ALP)

The steps of designing an Adaptive Learning Platform (ALP) that provides effective and personalized learning experiences for students are:

1. Define the platform's learning objectives: Start by clearly outlining the specific objectives of the platform. Identify what learners should be able to achieve or learn after using the platform. Consider the target audience, the subjects or topics to be covered, and the desired outcomes.
2. Create educational content: Develop or gather resources that align with the learning goals. These resources can include written materials, videos, interactive activities, and assessments. Ensure that the content is engaging, diverse, and suitable for personalized learning.
3. Understand the learners: Establish individual profiles for each student to understand their preferences and requirements. This information can be used to customize the learning environment for each person, taking into account factors like their educational level and preferred learning methods.
4. Assess prior knowledge: Administer exams or quizzes at the beginning to assess the existing knowledge of the learners. This helps the platform identify the starting point of each user and adjust the learning path accordingly.
5. Utilize adaptive algorithms: Develop intelligent algorithms that consider each learner's profile and assessment results. These algorithms dynamically modify the content and learning strategy to meet the specific needs of each student. Factors like learning style, difficulty level, progress, and feedback are taken into consideration.
6. Personalize the learning path: Create a unique learning path for each student based on the recommendations of the algorithm. This path should provide appropriate challenges and deliver information and exercises suitable for the learner's current level of knowledge.
7. Provide feedback and evaluations: Regularly assess and evaluate the progress of students. Offer feedback to help them understand their development and areas for improvement.
8. Monitor performance and analyze data: Keep track of students' progress and analyze data to gain insights and draw meaningful conclusions. This information assists in making informed judgments about the platform's effectiveness and potential for further development.
9. Continuously improve: Utilize data analysis and gather feedback from students to make ongoing improvements to the platform. Regularly update and enhance its features, functions, and content to maintain its usefulness and effectiveness.
10. Support instructors and students: Provide technical support and guidance to students, instructors, and administrators who are using the platform. Swiftly respond to their inquiries and address any concerns to ensure a successful learning experience.
11. Integrate with other systems: Establish connections between the ALP and other educational programs or resources, such as learning management systems or analytics platforms. This integration enables the sharing of data and facilitates a more comprehensive learning ecosystem.
12. Evaluate ALP performance and make necessary adjustments. This can be done by gathering feedback from users, conducting surveys, and analyzing data to identify areas that require improvement. Based on these evaluations, make appropriate changes to enhance the platform and ensure it continues to meet the needs of the learners.

Figure. 1 shows the steps of designing an Adaptive Learning Platform (ALP).

Authorized licensed use limited to: b-on: UNIVERSIDADE DO PORTO. Downloaded on November 03,2024 at 17:24:36 UTC from IEEE Xplore. Restrictions apply.

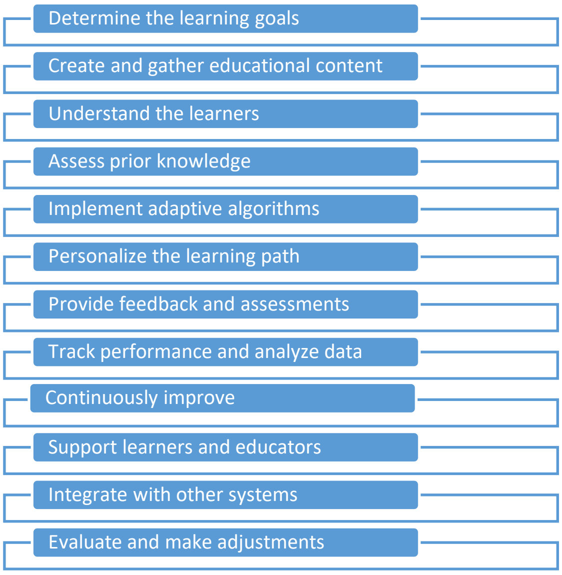


Fig. 1. Steps of designing an Adaptive Learning Platform (ALP).

IV. REINFORCEMENT LEARNING (RL) ALGORITHM

The ideal adaptable algorithm for an Adaptable Learning Platform (ALP) that strives to give individualized experiences and enhance learning outcomes based on individual requirements, goals, and learning preferences would be Reinforcement Learning (RL). Reinforcement Learning (RL) is especially well-suited to interactive systems in which an agent (in this example, the ALP) learns from its environment (the user) through trial and error. RL can modify the ALP's behavior according to input in the form of rewards or punishments, aiming for long-term cumulative rewards (learning outcomes). RL enables the ALP to tailor the learning experience to each individual user. The ALP may change its recommendations, activities, and assessments by gathering data on user attributes, learning preferences, goals, and evaluations to cater to the specific needs of each user. RL enables the ALP to learn and improve continually depending on user interactions. It may change its regulations and recommendations based on prior encounters, making the learning experience more customized and effective over time. The optimization of long-term cumulative rewards drives RL. In the context of ALP, this means that the ALP may focus on enhancing learning outcomes and assisting users in realizing their full potential by adjusting its techniques to optimize academic goal attainment. Gamification components may be incorporated into the ALP, offering prizes and incentives to keep users engaged and motivated in their learning path. By altering the difficulty level or tempo of activities dynamically. Figure. 2 shows the components of the RL:

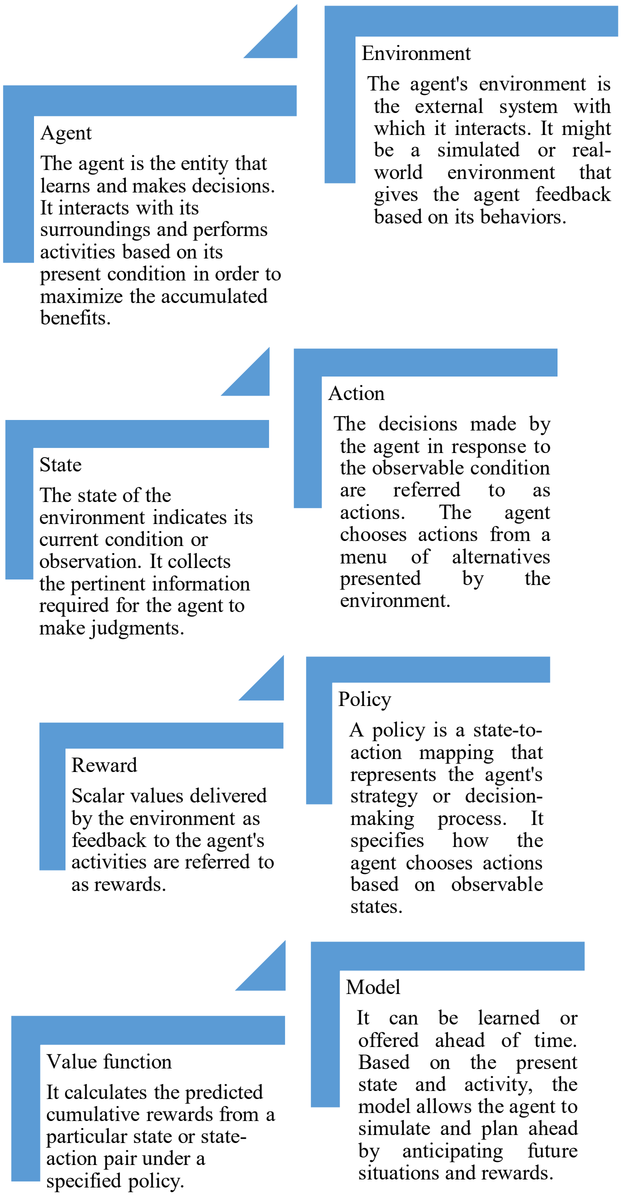


Fig. 2. Components of the RL

These elements interact to produce a feedback loop in which the agent observes the state, acts, receives rewards, and changes its policy or value.

V. ADAPTIVE LEARNING PLATFORM (ALP) IMPLEMENTATION

ALP was implemented using the Flask application in Visual Studio and programmed with Java script language for front-end implementation and Python for backend implementation. The system interacts with the SQLite database. It provides several routes for different functionalities related to user authentication, quizzes, notifications, goals, and progress tracking. Figure.3. shows the Login Page of the ALP. Users can change their passwords once every 60 days by providing their contact information so that they can be sent a verification code and reset their password.

Authorized licensed use limited to: b-on: UNIVERSIDADE DO PORTO. Downloaded on November 03,2024 at 17:24:36 UTC from IEEE Xplore. Restrictions apply.

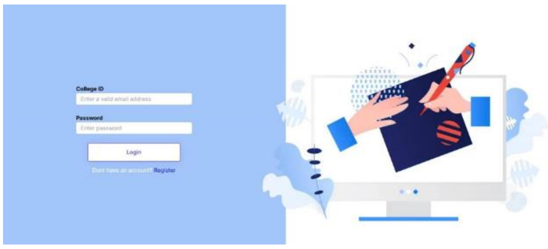


Fig. 3. Login Page of the ALP

Flask was imported to interact with an SQLite database. It provides several routes for different functionalities related to user authentication, quizzes, notifications, goals, and progress tracking as shown in the following code segment:

```python
app.py > get_language_analytic
import sqlite3
from flask import Flask, render_template, request, Response, jsonify
from flask_cors import CORS, cross_origin
import json
import hashlib, uuid
from datetime import datetime, timedelta

from helper import helper_check_score, helper_get_quiz, helper_user
salt = "my_secret_salt_kic"
app = Flask(__name__)
cors = CORS(app)

def get_db_connection():
    conn = sqlite3.connect('database.db')
    return conn


@app.route('/')
def index():
    conn = get_db_connection()
    posts = conn.execute('SELECT * FROM users').fetchall()
    conn.close()
    print(posts)
    return posts

@app.route('/posts', methods = ['GET'])
def posts():
    print(request.get_json()["test"])
    conn = get_db_connection()
    posts = conn.execute('SELECT * FROM posts').fetchall()
    conn.close()
```

get_db_connection() function establishes a connection to the SQLite database and returns the connection object. index() function handles the root route ("/") and retrieves all the rows from the "users" table in the database. It returns the fetched rows as a response. posts() function handles the "/posts" route and retrieves all the rows from the "posts" table in the database. It expects a JSON payload in the request and prints its "test" value. It returns the fetched rows as a response. sqlite3, hashlib and uuid were imported. ALP Connected to the SQLite database using the sqlite3.connect() method. This creates a connection object. Read the schema from the file schema.sql and execute it using the executescript() method of the connection object. This creates the necessary tables in the database as shown in the code segment below:

```python
init_db.py > ...
import sqlite3
import hashlib, uuid
salt = "my_secret_salt_kic"
connection = sqlite3.connect('database.db')


with open('schema.sql') as f:
    connection.executescript(f.read())

cur = connection.cursor()

cur.execute("INSERT INTO users (email, college_id, password) VALUES (?, ?, ?)",
            ('admin@admin.com', 'k-123456', hashlib.sha512(("password" + salt).encode('utf-8')).hexdigest())
            )



cur.execute("INSERT INTO courses (name) VALUES (?)",('Python',))
cur.execute("INSERT INTO courses (name) VALUES (?)",('C#',))
cur.execute("INSERT INTO courses (name) VALUES (?)",('CSS',))
cur.execute("INSERT INTO courses (name) VALUES (?)",('Dart',))
cur.execute("INSERT INTO courses (name) VALUES (?)",('Flutter',))
cur.execute("INSERT INTO courses (name) VALUES (?)",('Html',))
cur.execute("INSERT INTO courses (name) VALUES (?)",('Java',))
cur.execute("INSERT INTO courses (name) VALUES (?)",('Javascript',))
cur.execute("INSERT INTO courses (name) VALUES (?)",('Php',))
cur.execute("INSERT INTO courses (name) VALUES (?)",('Ruby',))


# placement
cur.execute("INSERT INTO exams (exam_title, type, level, end_date) VALUES (?, ?, ?, ?)",
            ('Placement Exam'
```

Users can log in to their accounts by providing their registered login ID and password, so they are able to use course features of the platform. Users can download, upload, update, and modify their interface if they want to do. Figure. 4. shows the Course Page of the ALP

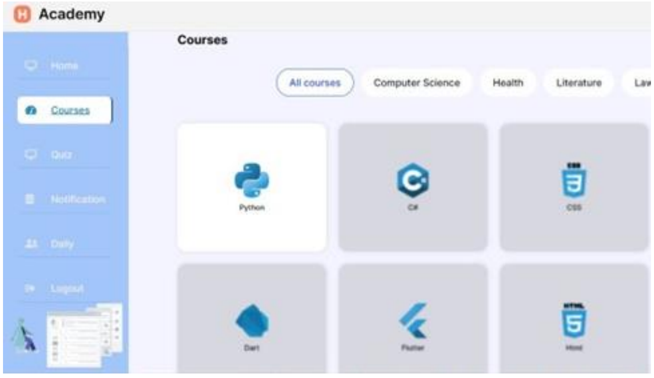
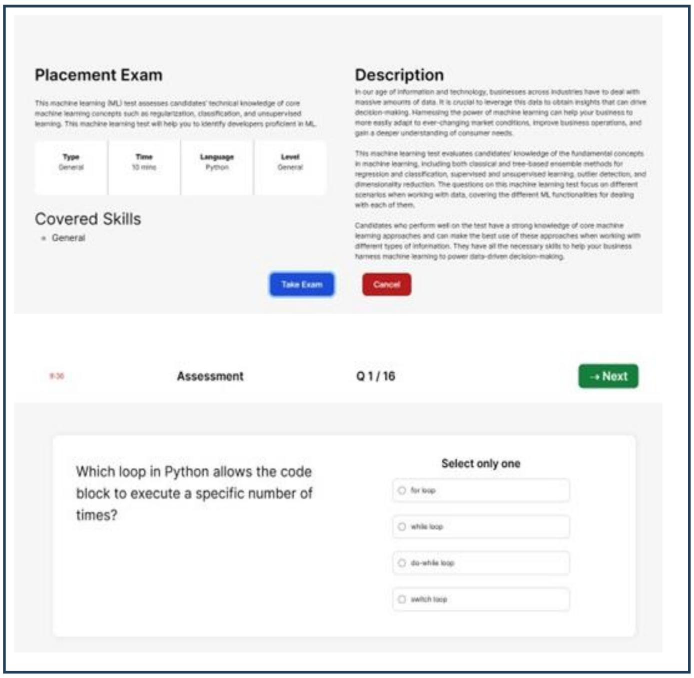


Fig. 4. Course Page of the ALP

For each student, there will be a unique learning path based on their data that will appear the first time they log in. Figure. 5. exhibits the Placement Test page.



Authorized licensed use limited to: b-on: UNIVERSIDADE DO PORTO. Downloaded on November 03,2024 at 17:24:36 UTC from IEEE Xplore. Restrictions apply.

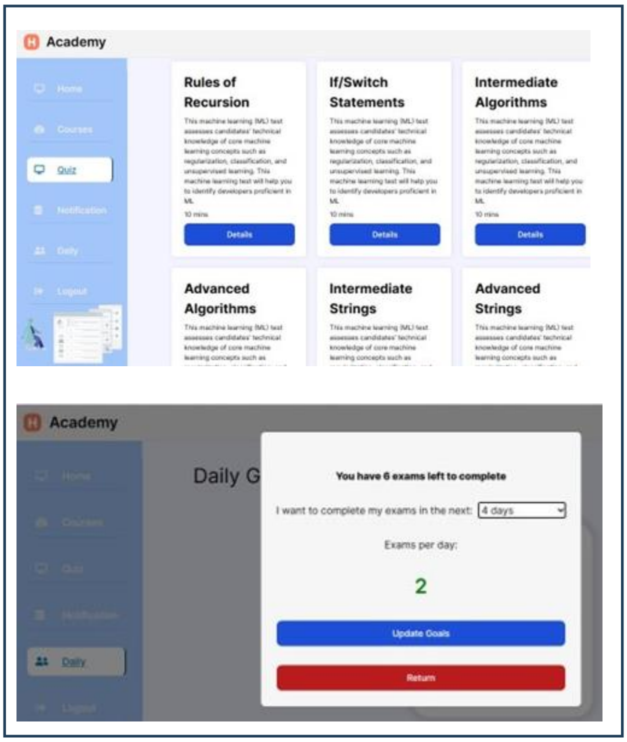


Fig. 5. Placement Test and Assessment Page

Users will receive notifications for completed assessments and advancing in their learning path. Figure. 6 shows the notification page of ALP. Users can see the Home page in Figure.7.

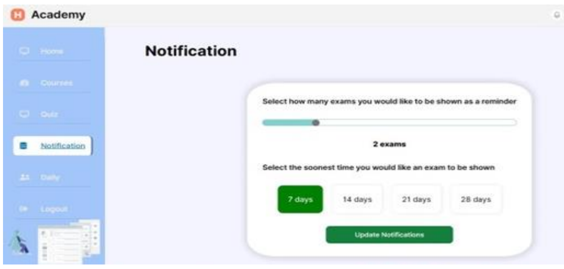


Fig. 6. Notification page

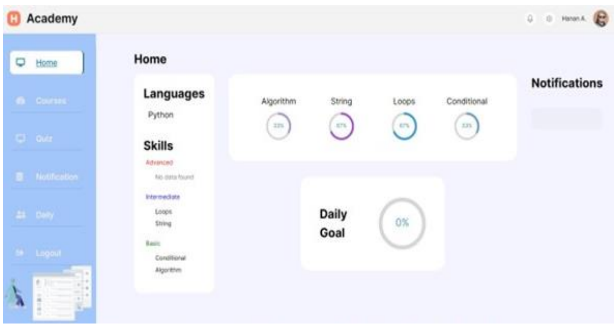


Fig.7. Home Page

VI. Conclusion

Adaptive Learning Platform (ALP) recognizes the particular requirements of every student, encourages engagement, and makes use of data-driven insights to improve learning outcomes and enable people to realize their full potential. Every user of ALP receives a totally customized experience. The platform provides personalized information, exercises, and evaluations that are catered to each user's particular requirements and goals by carefully assessing their traits and learning preferences. This degree of personalization makes sure that students get exactly what they need to succeed in their academic endeavors.

References

[1] S. Wang et al., "When adaptive learning is effective learning: comparison of an adaptive learning system to teacher-led instruction," Interactive Learning Environments, vol. 31, no. 2, pp. 793-803, 2023.

[2] D. Kem, "Personalised and adaptive learning: Emerging learning platforms in the era of digital and smart learning," International Journal of Social Science and Human Research, vol. 5, no. 2, pp. 385-391, 2022.

[3] J. L. R. Muñoz et al., "Systematic Review of Adaptive Learning Technology for Learning in Higher Education," Eurasian Journal of Educational Research, vol. 98, no. 98, pp. 221-233, 2022.

[4] M. Liu et al., "Using data to understand how to better design adaptive learning," Technology, Knowledge and Learning, vol. 22, pp. 271-298, 2017.

[5] E. M. Smaili et al., "Towards sustainable e-learning systems using an adaptive learning approach," in Emerging Trends in ICT for Sustainable Development: The Proceedings of NICE2020 International Conference, Cham: Springer International Publishing, 2021, pp. 365-372.

[6] P. J. Standen et al., "An evaluation of an adaptive learning system based on multimodal affect recognition for learners with intellectual disabilities," British Journal of Educational Technology, vol. 51, no. 5, pp. 1748-1765, 2020.

[7] Y. Long et al., "Complexity-aware adaptive training and inference for edge-cloud distributed AI systems," in 2021 IEEE 41st International Conference on Distributed Computing Systems (ICDCS), IEEE, 2021, pp. 573-583.

[8] J. Isabona et al., "Machine learning-based boosted regression ensemble combined with hyperparameter tuning for optimal adaptive learning," Sensors, vol. 22, no. 10, pp. 3776, 2022.

[9] B. Xue et al., "Complex ISAR target recognition using deep adaptive learning," Engineering Applications of Artificial Intelligence, vol. 97, pp. 1

[10] M. Zemzami, N. Elhami, M. Itmi, and N. Hmina, "An evolutionary hybrid algorithm for complex optimization problems," International Journal of Advanced Trends in Computer Science and Engineering, Article vol. 8, no. 2, pp. 126-133, 2019, doi: 10.30534/ijatcse/2019/05822019.

[11] I. Sahmi, A. Abdellaoui, T. Mazri, and N. Hmina, "MQTT-PRESENT: Approach to secure internet of things applications using MQTT protocol," International Journal of Electrical and Computer Engineering, Article vol. 11, no. 5, pp. 4577-4586, 2021, doi: 10.11591/ijece.v11i5.pp4577-4586.

Authorized licensed use limited to: b-on: UNIVERSIDADE DO PORTO. Downloaded on November 03,2024 at 17:24:36 UTC from IEEE Xplore. Restrictions apply.