MATHEMATICAL MODELING AND COMPUTING, Vol. 11, No. 4, pp. 1003–1012 (2024)

Personalized education plan construction using neural networks

Kopylchak O., Kazymyra I., Mukan O., Bondar B.

Lviv Polytechnic National University,
12 S. Bandera Str., 79013, Lviv, Ukraine

(Received 8 March 2024; Revised 7 November 2024; Accepted 14 November 2024)

In the paper, a personalized education planning system that utilizes neural networks and artificial intelligence to adapt learning paths for individual learners dynamically is presented. The system employs neural networks to analyze learner profiles, preferences, and real-time performance data, enabling the generation of tailored study plans. Neural networks are integral in predicting learner needs by analyzing past performance, learning style, and engagement patterns, allowing the system to recommend appropriate learning modules and optimal study schedules. Additionally, the system adjusts learning plans in real time, balancing cognitive load and ensuring personalized pacing to prevent learner fatigue. By incorporating these advanced mechanisms, the system provides content recommendations and schedules that evolve continuously as learners progress. The adaptive nature of the system is further enhanced through neural networks' ability to optimize long-term learning strategies, ensuring that the right balance between challenge and support is maintained. The proposed system can be seamlessly integrated with Learning Management Systems (LMS), offering a scalable solution for personalized education. The paper highlights the effectiveness of neural networks in creating efficient, learner-centered study paths and improving educational outcomes through data-driven adaptation.

Keywords: *personalized education planning; neural networks in education; AI-driven learning systems; adaptive learning; personalized learning recommendation; cognitive load management; dynamic learning schedule; artificial intelligence in education; education technology; learner-centered education.*

2010 MSC: 68T05, 68T20, 97U70, 68T07, 97D40 **DOI:** 10.23939/mmc2024.04.1003

1. Introduction

Artificial Intelligence and neural networks have emerged as transformative technologies with the potential to address complex real-world problems across diverse fields. Their versatility and impact are evidenced by successful applications in healthcare [1, 2], manufacturing [3], agriculture [4], data science [5], cybersecurity [6], education [7, 8], etc.

The use of AI in education, especially for personalized learning and self-education, has gained significant attention in recent years. Various researchers have contributed to the development of intelligent tutoring systems and adaptive learning platforms designed to cater to individual learning needs. The field of Artificial Intelligence in Education (AIED) has evolved over the past 25 years, with researchers identifying key strengths and future opportunities for AI in education. Roll and Wylie (2016) highlight that AI technologies in education are at a crossroads, offering two distinct approaches for future research: an evolutionary path that builds on current classroom practices and collaboration with teachers, and a revolutionary path where AI technologies become embedded in students' everyday lives, supporting their learning goals in a personalized manner [9].

The evolutionary approach is primarily concerned with improving current practices, such as enhancing one-on-one tutoring experiences through systems that model effective teaching strategies. The revolutionary approach, on the other hand, emphasizes the integration of AI systems into self-directed learning, allowing learners to plan and execute their educational journeys with minimal external intervention. The implementation of AI in education has predominantly been through Intelligent Tutoring Systems (ITS), which offer personalized guidance and feedback to students based on their learning behaviors and performance data. For instance, the work by VanLehn (2011) demonstrates that ITS can replicate the effectiveness of human tutors by providing step-by-step feedback and tailored educational content [10].

Similarly, adaptive learning systems, as discussed by Hwang et al. (2021), have been designed to analyze students' cognitive and affective performance, adjusting learning plans accordingly to maximize learning efficiency [11]. These systems are particularly valuable in self-education, where learners often need real-time feedback to guide their progress without the constant presence of a teacher. Moreover, studies by Zawacki-Richter et al. (2018) have shown that AI-based systems can analyze large-scale data from various learners to make informed decisions about personalized content delivery [12].

This is critical in the context of self-education, where AI can offer custom learning paths based on a learner's previous performance, preferred learning style, and subject interests. As learners progress, the AI system refines its understanding of their strengths and weaknesses, enabling more accurate recommendations for further study. Neural networks, a subset of AI, have been increasingly applied to improve personalized learning planning. Neural networks excel at identifying patterns in data and can predict learners' future performance based on their past behaviors. This predictive capability is essential for developing personalized learning plans that adapt to the learner's evolving needs. Roll and Wylie (2016) point out that neural networks allow for dynamic adjustment of learning goals and can suggest optimal study paths, considering the subject matter's complexity and the learner's proficiency level [9].

In the context of self-education, neural networks play a pivotal role in creating adaptive learning environments. For example, some AI-driven platforms use neural networks to continuously assess a learner's progress and recommend the next steps in their education plan. These platforms can also identify when a learner is struggling with specific topics and adjust the difficulty or type of content to ensure mastery before moving on to more advanced material. The application of AI in self-education goes beyond just delivering content. AI systems can support learners in setting educational goals, tracking their progress, and adjusting study plans based on their performance. According to the systematic review by Hwang et al. (2021), AI can provide scaffolding for learners to engage in self-regulated learning, guiding them through complex problem-solving processes and fostering critical thinking skills [11].

This is particularly important for self-directed learners who may lack the external motivation and structure provided by traditional educational systems. Furthermore, as AI technologies advance, there is growing potential for creating more immersive and interactive learning experiences. For instance, AI-powered educational games and simulations offer learners the opportunity to explore subjects in a hands-on manner, making learning more engaging and effective. Research by Dzikovska et al. (2014) has shown that complex problem-solving environments supported by AI can significantly enhance learners' ability to apply theoretical knowledge to practical situations [13].

Despite the promising advancements in AI for personalized learning, several challenges must be addressed by researchers and educators. One primary concern is the ethical implications of using AI to collect and analyze vast amounts of student data. Issues related to data privacy and algorithmic bias must be carefully managed to ensure that AI systems are fair and transparent. Moreover, as Roll and Wylie (2016) suggest, integrating AI into education requires a careful balance between technology and pedagogy. It is essential to ensure that AI complements, rather than replaces, the human elements of education [9]. Looking ahead, the research suggests that AI, particularly through the use of neural networks, will continue to play a critical role in shaping the future of self-education. However, further research is needed to refine the algorithms driving personalized learning systems and ensure they can be seamlessly integrated into various educational contexts.

1.1. Materials

To integrate AI algorithms into personalized learning systems, researchers have explored various models and approaches that enhance learning outcomes and engagement. Key AI techniques applied in personalized learning include neural networks, machine learning, natural language processing, and cognitive neuropsychology-based models.

Neural networks and machine learning algorithms form the backbone of adaptive learning platforms, which tailor content to individual learners based on their performance data. Neural networks enable predictive models that forecast a learner's future needs by analyzing past behaviors, thus guiding the learning journey dynamically. Machine learning algorithms continuously refine the system's recommendations based on real-time interactions, allowing the platform to become more precise over time. For example, the AI-enabled intelligent assistant (AIIA) framework discussed by Sajja et al. (2024) leverages ML and NLP to offer personalized support, quizzes, and learning paths based on student inquiries and performance [14].

Natural language processing is essential for developing intelligent tutoring systems that provide personalized feedback and interaction. These systems can interpret student inputs, assess the comprehension level, and offer tailored responses, creating an interactive and engaging learning experience. This technology is particularly beneficial in e-learning platforms, where timely feedback and personalized content are crucial for self-directed learners [15, 16].

Adaptive learning platforms (ALPs) such as Carnegie Learning and DreamBox Learning apply AI to adjust the learning content dynamically to a student's progress and comprehension. These platforms rely on a combination of AI techniques, including learning analytics, to assess a learner's current knowledge and suggest optimal learning activities. The goal is to enhance learning efficiency by presenting material that aligns with the learner's evolving understanding [10–12].

Some adaptive systems are designed using principles from cognitive neuropsychology, which align AI-driven personalized learning with how the brain processes information. This approach improves student engagement and retention by adapting content based on the learner's cognitive state and providing feedback that is neurologically aligned with learning patterns [11, 12].

The aim of this work is to develop a flexible and adaptive personalized learning planning system that leverages neural networks and artificial intelligence to optimize the learning experience for individual learners. The system aims to dynamically generate and adjust personalized learning paths based on real-time learner data, ensuring that each learner receives tailored educational content, feedback, and pacing to improve learning efficiency and outcomes. This aim focuses on building a system that not only personalizes learning but also enhances the overall educational process by integrating AI-driven decision-making, allowing for continuous optimization and adaptation of learning strategies based on the learner's performance and preferences.

As personalized learning planning involves managing a complex set of variables and constraints, a Constraint Satisfaction Problem (CSP) approach is a natural choice to proceed with. CSP methods have been extensively studied and applied in various fields to address intricate problem-solving scenarios, and many advanced techniques are currently in development. For example, recent innovations such as Recurrent Transformer models and Graph Neural Networks (GNNs) have emerged as powerful tools in handling CSPs with complex, sequential, or relational data. GNNs, in particular, excel in representing variable dependencies in scheduling or adaptive learning systems by modeling entities (e.g., learning tasks or time slots) as nodes and constraints as edges, making them highly suited for our personalized learning application. Additionally, methods like SATNet, which integrates neural networks with traditional satisfiability (SAT) solvers, offer promising directions for combining data-driven insights with classic CSP techniques.

In our system, Graph Neural Networks are utilized for their efficacy in capturing relationships and dependencies in educational scheduling, allowing our personalized learning plan to adapt dynamically to evolving learner needs. Integrating other approaches, such as Recurrent Transformer models, could further enhance long-term learning strategies, ensuring efficient and context-aware recommendations for each student.

1.2. Problem statement

For a Personalized Learning Planning System (PLPS), we seek an automated system that, given certain input data regarding learner preferences, progress, and available resources, can generate a personalized learning plan. The system must ensure that the recommendations align with the learner's goals and adapt over time as learning progresses, while satisfying individual constraints.

The PLPS problem can be formally described as follows.

Given a list of:

1. Learners with unique profiles, including learning goals, cognitive abilities, and available learning hours.
2. Learning modules (or subjects) categorized by complexity, time commitment, and required prerequisites.
3. Resources such as available instructors, multimedia content, self-paced exercises, and assessments.
4. Neural network models for recommending content and tracking progress, personalized based on past interactions.

The PLPS problem consists of planning a learning pathway for each learner in such a way that:

1. Each learning module is tailored to the learner's pace and ability, with some modules offering flexibility in delivery mode (e.g., visual, textual, or interactive).
2. The recommended modules should not conflict with the learner's prerequisites; only one learning module requiring the same prerequisite should be scheduled at any given time.
3. Modules can be spaced out over time to balance the learner's workload and prevent cognitive overload. Consecutive learning sessions may be scheduled for intensive subjects or for project-based work.
4. A learning resource (e.g., an instructor or learning assistant) cannot be assigned to more than one learner at the same time for live, synchronous sessions.
5. The time commitment assigned to a learner per day should not exceed the learner's availability as defined in their profile.
6. Assessments or collaborative projects may be assigned to more than one learner, especially for subjects that benefit from group work or peer feedback.
7. Learning sessions must be distributed in a way that avoids overlap of learning goals in conflicting cognitive domains (i.e., technical and creative tasks should not be scheduled back-to-back unless designed to complement one another).

2. Design of PLPS

As personalized learning planning involves managing a complex set of variables and constraints, a Constraint Satisfaction Problem (CSP) approach is a natural choice to proceed with. Mathematically, a CSP is a triplet (X, D, C), where X is a non-empty finite set of variables, D is the set of domains for these variables, and C is the set of constraints restricting the values the variables can take simultaneously.

The Personalized Learning Planning System (PLPS) must schedule and organize different interlinked entities: the learners' goals and progress, the available learning resources, and the learning modules (subjects). This problem, while complex, can be broken into smaller, manageable tasks based on learning objectives, time availability, and content progression.

To optimize the process, we employ neural network-based recommendation systems to suggest content that aligns with the learner's current progress. In the first phase, a high-level learning plan (Master Learning Path, MLP) is created, providing a framework of learning modules that can be tailored to the learner's preferences. The system checks for prerequisites and suggests modules based on the learner's profile, learning pace, and available resources.

In the second phase, personalized adjustments are made through a fine-tuned planner (Personalized Learning Planner, PLP). The PLP adjusts the recommended learning path by analyzing real-time data on learner performance, recommending new resources, or changing the learning sequence based on progress and evolving interests.

2.1. Input for the learning planner

Before the system begins generating personalized learning pathways, it requires specific input data about the learner and the available resources. A detailed case study of personalized learning systems can illustrate these inputs. The inputs required include:

— Learner profile: including learning objectives, learning style (e.g., visual, interactive), cognitive load, and daily available study time.
— Learning modules: subjects or topics categorized by complexity, prerequisites, and estimated completion time.
— Available resources: instructors, self-paced digital content, and collaborative learning opportunities (e.g., group projects, peer feedback).
— System preferences: including scheduling flexibility, recommended pacing, and content delivery modes (text, video, exercises).
— External resources: information on relevant external sources like e-books, MOOCs, or subject matter experts.

Based on the learner's profile and preferences, the system processes these inputs to generate an adaptive, personalized learning path that evolves over time as new data (e.g., learner progress or feedback) is added.

Example Input Tables: Learner Profile: An example of learner data input is provided in Table 1, showcasing learning goals, cognitive load capacity, and daily study hours.

Table 1. Learner Profile.

Learner	Goal	Cognitive Load	Available Time	Learning Style	Preferred Pace
Learner 1	Python Proficiency	Medium	2 hrs/day	Visual	Steady
Learner 2	Data Science	High	4 hrs/day	Interactive	Fast

Table 2. Learning Modules.

Module	Prerequisite	Time to Complete	Type	Level
Python Basics	None	10 hrs	Theory	Beginner
Data Structures	Python Basics	15 hrs	Practical	Intermediate
Machine Learning	Data Structures	20 hrs	Theory + Practical	Advanced

2.2. Personalization layers

The learning pathway generation system operates in two distinct layers.

Master Learning Path (MLP): This system component generates an overarching learning pathway based on the learner's long-term goals, such as mastering a subject or completing a certification. The MLP organizes learning modules based on prerequisites, and provides a high-level overview of the subjects that need to be covered. The output is a learning blueprint that assigns tentative schedules and learning sequences.

Personalized Learning Planner (PLP): At this layer, more granular personalization occurs, adjusting the learner's daily or weekly learning tasks based on real-time data. The PLP makes on-the-fly adjustments based on learner feedback, progress, and performance metrics, ensuring the plan evolves as the learner advances. For example, if a learner struggles with a particular topic, the PLP may reschedule or provide supplementary material.

The design approach leverages the power of neural networks to improve the personalization process continuously. By balancing the complexity of the problem across these layers, we can provide a robust personalized learning plan that adapts dynamically to each learner's needs.

2.3. Personalized scheduling strategies

In personalized learning, scheduling must account for the learner's goals, time constraints, and cognitive load. The system checks feasibility based on available learning time, resources, and the difficulty of subjects. Below is the adapted version of scheduling strategies for Personalized Learning Planning:

— Feasibility check – If the learner's available study hours per week multiplied by the number of modules they can engage with per week is greater than or equal to the total workload required to achieve their goals, then a feasible learning plan can be generated. The workload includes time for consuming learning materials, engaging in interactive activities, and completing assessments. If scheduling cannot be achieved, adjustments can be made by reducing the learning intensity (fewer subjects) or distributing study hours more effectively across the week (e.g., allocating some tasks to evenings or weekends).

— Creation of master learning path (MLP) – Using the learner's available study hours and goals, the system allocates time slots to different subjects. Each subject is assigned a priority based on its prerequisite structure and importance to the learner's objectives. For subjects that involve collaboration or live interaction, fixed time slots are created first. Self-paced or asynchronous subjects are then scheduled flexibly around these fixed sessions. This master schedule becomes the blueprint for personalized learning. Subjects that cross multiple domains (interdisciplinary) are prioritized to avoid conflicts in cognitive load.

2.4. Types of constraints checked in personalized learning planning

For effective personalized planning, several constraints need to be checked, similar to traditional scheduling systems:

1. Learner time preference constraint: Learning sessions should match the learner's availability and preferences (e.g., morning, evening, or spread throughout the day).
2. Subject complexity constraint: Some subjects may require extended focus or consecutive sessions (e.g., practical labs or project work), while others can be divided into shorter, self-paced units.
3. Resource availability constraint: For interactive subjects involving instructors or group work, resources (instructors or peers) should not be overbooked at the same time.
4. Learning load constraint: The system ensures that the total number of learning hours per day or week does not exceed the learner's cognitive load capacity, preventing burnout.

2.5. Output of the personalized learning planner (PLP)

The output of the Personalized Learning Planner (PLP) will be a list similar to the following, where each subject is assigned its study time slot based on the learner's preferences and needs.

Table 3. Personalized Learning Path Output.

Subject Code	Module	Assigned Learning Time Slots				
ML-01	Machine Learning	Session 1	Session 2	Session 3	Session 4	Session 5
		Morning Slot	Evening Slot	Free Study	Practice	Assessment
	—	—	—	—	—	—

Using this schedule, the learner can follow their personalized study plan, which adapts dynamically as they progress through their modules.

2.6. Creation of personalized learning schedule

The Personalized Learning Planner (PLP) scheduler will handle the final stage by assigning study sessions, learning modules, or interactive activities to specific time slots based on the learner's preferences and availability. Since the learner's time is allocated flexibly in earlier stages, the scheduler will ensure that there are no conflicts among subjects or tasks requiring similar cognitive efforts.

The following constraints need to be taken into account:

1. Learner preferences for study time (morning/evening or specific days).
2. More than one learner can participate in collaborative sessions (such as group projects or peer assessments).
3. A learner cannot be assigned two conflicting cognitive tasks at the same time (for instance, solving complex mathematical problems and working on a creative writing task simultaneously).
4. A limit on the maximum number of hours assigned per day or week to prevent cognitive overload.

2.7. The scheduling model

In the personalized learning system, the scheduling model is based on three key entities: Subject (learning module), Learner, and Study Time Slots. Each of these variables is described below:

Variables:

1. Learning Module Object: The number of learning module objects corresponds to the number of subjects or learning units the learner will engage with during a given period. The system stores data on assigned study sessions, completion status, and the associated study time slots.

Table 4. Learning Module Object.

Integer completionLevel=0;	Used to check if the learning module is completed. If completionLevel < workload, the module is incomplete.
Integer workload;	The total number of study hours allocated to the learning module.
String timeSlotCode[workload];	An array of strings used to store time slots assigned for studying the module.
String learnerCode[workload];	Stores the learner ID assigned to the module in the corresponding time slots.
Constructor and get/set functions	Functions to manage the module data.

2. Learner Object: Each learner object stores data on available study hours, the number of assigned learning tasks, and preferences such as learning pace and time slots for study.

Table 5. Learner Object.

Integer studyLoad=0;	Used to check if the learner is assigned the correct amount of study hours. If studyLoad < planned load, assignments are incomplete.
Integer workload;	Stores the total number of study hours required by the learner for the session.
String timeSlotCode[workload];	An array of strings used to store study session time slots for the learner.
String moduleCode[workload];	An array of strings used to store learning module codes corresponding to the assigned time slots.
Constructor and get/set functions	Functions to manage the learner's study plan.

3. Time Slot Object: The number of time slot objects is determined by the learner's available study hours per week, multiplied by the number of study sessions scheduled. Each time slot object stores data on which learning module and learner it is assigned to.

Table 6. Time Slot Object.

Integer assigned=0;	Used to check if the time slot is assigned to a learner for a specific study session.
String moduleCode;	Stores the learning module assigned to the time slot.
set, get functions	Functions to manage time slot assignments.
Constructor of the class	Initialization functions for creating new time slots.

2.8. Domains for the variables

In the Personalized Learning Planning System (PLPS), the domain of each variable is defined based on the type of object involved in the learning process.

The domain of Learning Module objects consists of the set of study sessions (time slot codes) and learner IDs. The domain of Learner objects includes pairs of time slot codes and learning module codes, representing when and what the learner is studying. The domain for Time Slot objects is the set of time slot codes that represent the available study periods during a learner's schedule.

2.9. Constraints for the Master Learning Path Scheduler (MLP)

The number of variables in the system equals the number of Learning Module objects (subjects) that need to be scheduled.

The system creates Learning Module objects identified by their module codes. For example, module codes could include PY-ML-01 for Python Basics or DS-ML-02 for Data Structures in Machine Learning. These objects are partitioned into categories based on their complexity or subject type, such as core modules and elective modules.

Two sets of time slots are maintained.

Assigned time slots: These are the slots that have already been allocated to a learner. Unassigned time slots: These are the available slots that can be used for further assignments. Assignments are made from the unassigned time slots based on the following constraints:

Constraint 1: Core learning modules that are prerequisites for more advanced subjects must be scheduled first. For instance, foundational topics such as Python Basics need to be completed before moving on to Machine Learning.

Constraint 2: The number of hours assigned to a learning module must match its workload, ensuring that each module is given enough time for mastery.

Constraint 3: Using learner preferences for time of day (morning, afternoon, evening), assignments are made from the set of unassigned time slots. Additionally:

Intensive or project-based learning sessions should be assigned consecutive slots. Regular theory-based sessions should be spread out to ensure a balanced cognitive load.

Constraint 4: No two modules of the same cognitive domain (e.g., two highly technical subjects) should be assigned back-to-back without a break to avoid cognitive fatigue. For example, a learner should not have Data Structures followed by Advanced Algorithms without a break or a creative task in between.

The system performs constraint checking only for assigned time slots. Once all core learning modules are scheduled, elective modules are assigned using similar constraints.

2.10. Constraints for the Personalized Learning Planner (PLP)

Once the Master Learning Path (MLP) assigns learning modules to time slots, the Personalized Learning Planner (PLP) assigns specific study sessions or collaborative tasks to learners. The following constraints are applied.

Constraint 1: The system checks each learner's learning preferences to ensure that study sessions align with their preferred schedule (e.g., morning study sessions for a learner who performs better earlier in the day).

Constraint 2: A learner cannot be assigned two different modules at the same time. The system ensures that no overlapping study sessions are assigned, avoiding schedule conflicts between modules.

3. Conclusion

The implementation of a personalized education planning system using neural networks demonstrates a promising advancement in the field of education technology. The system provides a dynamic, adaptive learning experience that can adjust to individual learners' needs, preferences, and learning styles. This approach not only enhances the efficiency of learning processes but also helps learners achieve better outcomes.

Through the integration of real-time data, neural networks continuously assess learner progress, optimize learning paths, and provide tailored recommendations that evolve with the learner's development. This allows for a more engaging and supportive educational experience. Moreover, the ability to adapt to various learning management systems (LMS) makes the proposed system highly scalable and versatile across different educational settings.

[1] Elfatouaki H., Adnane L., Charafeddine A. Z., Mohamed A. AI-Enhanced ECG diagnosis system for acute myocardial infarction with LBBB: Constant-Q transform and ResNet-50 integration. Mathematical Modeling and Computing. **11** (3), 654–662 (2024).

[2] Roudani M., Elkari B., El Moutaouakil K., Ourabah L., Hicham B., Chellak S. Twitter-sentiment analysis of Moroccan diabetic using Fuzzy C-means SMOTE and deep neural network. Mathematical Modeling and Computing. **11** (3), 835–847 (2024).

[3] Zoubaidi Z., Herrou B., Sekkat S., Khadiri H. Conception of a new quality control method based on neural networks. Mathematical Modeling and Computing. **11** (3), 692–701 (2024).

[4] Sghir A., Ziani M., El Handri K. A data-driven fusion of deep learning and transfer learning for orange disease classification. Mathematical Modeling and Computing. **11** (3), 870–882 (2024).

[5] Belhabib F., El Moutaouakil K., Rbihou S., Elafaar A. Big data clustering through fusion of FCM, optimized encoder-decoder CNN, and BiLSTM. Mathematical Modeling and Computing. **11** (3), 798–813 (2024).

[6] Solomka I. R., Liubinskyi B. B., Torshyn V. V. Application of machine learning algorithms to enhance blockchain network security. Mathematical Modeling and Computing. **11** (3), 893–903 (2024).

[7] Thakur R. K., Agrawal N. K., Kumar P. A practical approach to college timetable scheduling. Mathematical Modeling and Computing. **11** (3), 710–719 (2024).

[8] Jedidi Y., Ibriz A., Benslimane M., Hachmoud A., Tmimi M., Hajjioui Y., Rahhali M. Predicting students' academic performance and modeling using data mining techniques. Mathematical Modeling and Computing. **11** (3), 814–825 (2024).

[9] Roll I., Wylie R. Evolution and Revolution in Artificial Intelligence in Education. International Journal of Artificial Intelligence in Education. **26**, 582–599 (2016).

[10] VanLehn K. The relative effectiveness of human tutoring, intelligent tutoring systems, and other tutoring systems. Educational Psychologist. **46** (4), 197–221 (2011).

[11] Hwang G.-J., Sung H.-Y., Chang S.-C., Huang X.-C. A fuzzy expert system-based adaptive learning approach to improving students' learning performances by considering affective and cognitive factors. Computers and Education: Artificial Intelligence. **1**, 100003 (2020).

[12] Zawacki-Richter O., Latchem C. Exploring four decades of research in Computers & Education. Computers & Education. **122**, 136–152 (2018).

[13] Dzikovska M., Steinhauser N., Farrow E., Moore J., Campbell G. BEETLE II: deep natural language understanding and automatic feedback generation for Intelligent tutoring in Basic Electricity and Electronics. International Journal of Artificial Intelligence in Education. **24** (3), 284–332 (2014).

[14] Sajja R., Sermet Y., Cikmaz M., Cwiertny D., Demir I. Artificial Intelligence-Enabled Intelligent Assistant for Personalized and Adaptive Learning in Higher Education. Information. **15** (10), 596 (2024).

[15] Amri S., Bani R., Zenkouar L., Guennoun Z. Improving Amazigh POS tagging using machine learning. Mathematical Modeling and Computing. **11** (3), 741–751 (2024).

[16] Lafkiar S., En Nahnahi N. An Arabic question generation system based on a shared BERT-base encoder-decoder architecture. Mathematical Modeling and Computing. **11** (3), 763–772 (2024).

Персоналізоване планування навчання на основі нейронних мереж

Копильчак О., Казимира І., Мукан О., Бондар Б.

Національний університет «Львівська політехніка»,
вул. С. Бандери, 12, 79013, Львів, Україна

У статті представлено персоналізовану систему планування навчання, яка використовує нейронні мережі та штучний інтелект для динамічної адаптації навчальних шляхів для окремих користувачів. Система використовує нейронні мережі для аналізу профілів користувачів, уподобань і даних про ефективність у реальному часі, що дозволяє створювати індивідуальні навчальні плани. Нейронні мережі є невід'ємною частиною прогнозування потреб учнів шляхом аналізу минулої успішності, стилю навчання та моделей залучення, що дозволяє системі рекомендувати відповідні навчальні модулі та оптимальні розклади персоналізованого навчання. Крім того, система коригує навчальні плани в реальному часі, балансуючи когнітивне навантаження та забезпечуючи персоналізований темп, щоб запобігти втомі. Використовуючи ці вдосконалені механізми, система надає рекомендації щодо контенту та графіку навчання, які постійно змінюються в міру прогресу користувача. Адаптивний характер системи додатково посилюється завдяки здатності нейронних мереж оптимізувати довгострокові стратегії навчання, забезпечуючи баланс між поставленим завданням і підтримкою. Запропоновану систему можна легко інтегрувати з системами управління навчанням (LMS), пропонуючи масштабоване рішення для персоналізованого навчання. У статті підкреслюється ефективність нейронних мереж у створенні ефективних, орієнтованих на учня планів навчання та покращенні результатів навчання за допомогою адаптації на основі даних.

Ключові слова: *персоналізоване планування навчання; нейронні мережі в освіті; навчальні системи на основі штучного інтелекту; адаптивне навчання; персоналізована рекомендація навчання; управління когнітивним навантаженням; динамічний розклад навчання; штучний інтелект в освіті; освітні технології; навчання, орієнтоване на учня.*