Teaching Neural Networks to Computer Science Students in Higher Education: Approaches and Challenges

Meruyert Serik[1], Nurzhanar Karilkhan[1,*], Jaroslav Kultan[2], and Dashzhan Narodkhan[3]

[1]Computer Science Department, Faculty of Information Technology, L.N. Gumilyov Eurasian National University, Astana, Kazakhstan
[2]Department of Applied Informatics, University of Economics in Bratislava, Bratislava, Slovakia
[3]Departments of the Mining Faculty, Mine aerology and a labor safety, Abylkas Saginov Karaganda Technical University, Karaganda, Kazakhstan
Email: serik_meruerts@mail.ru (M.S.); iskulai@gmail.com (N.K.); jaroslav.kultan@euba.sk (J.K.); dos_good@mail.ru (D.N.)
[*]Corresponding author
Manuscript received April 16, 2025; revised May 12, 2025; accepted August 18, 2025; published December 16, 2025

***Abstract*—Modern artificial Intelligence (AI) technologies are increasingly shaping higher education, particularly in their use in training computer science students and integrating neural networks into the learning process. The research *aims* to evaluate modern approaches and challenges to teaching neural networks to computer science students in higher education. This study employs a quasi-experimental method involving 85 third-year students enrolled in the Computer Science programs at L.N. Gumilyov Eurasian National University and Buketov Karaganda University. The participants were divided into two groups: an experimental group and a control group. The experimental group received instruction with an enhanced curriculum that included modern tools such as TensorFlow, Keras, OpenCV, and Google Colab. Data were collected through pre-tests and post-tests, evaluating changes in student motivation, content comprehension, and technical competencies. Pearson's chi-square test was utilized to analyze the data, which revealed statistically significant improvements in the experimental group compared to the control group. These results suggest that integrating updated content and hands-on technologies into teaching practices enhances students' skills and learning outcomes in neural network education. The revised neural network curriculum had a positive impact on student learning outcomes. The research emphasizes the importance of continually updating the curriculum to meet the evolving demands of modern AI.**

***Keywords*—Artificial Intelligence (AI), machine learning, deep learning, neural networks, computer science education, AI tools, Python programming**

I. Introduction

Over the past decade, AI has garnered significant attention from both the scientific community and the general public. In the field of AI, subfields such as Machine Learning (ML) and Deep Learning (DL) have been widely studied and discussed in both technology and non-technology journals [1]. AI, ML, DL, and Neural Networks (NNs) demonstrate a hierarchical relationship, where each subsequent field is a subset of the previous one [2]. However, there is still considerable confusion regarding the distinctions between DL, ML, and AI. Despite their close associations, these terms are not interchangeable [3].

ML is a subfield of AI that focuses on developing algorithms that can learn from data and improve their performance without explicit programming [4]. AI encompasses a broader concept, involving the creation of systems that mimic human cognitive abilities. DL, a subset of ML, utilizes multilayer NNs to analyze data and extract complex patterns [5].

AI is currently being utilized in various fields. In particular, NNs were initially designed to simulate the structure and function of the human nervous system by modeling computational units as artificial neurons. Their main goal is to facilitate the development of intelligent systems by replicating cognitive processes through biologically inspired architectures [6]. In recent years, NNs have garnered significant scholarly interest due to their ability to emulate the brain's pattern recognition capabilities [7]. They have proven effective in various decision-making domains, including Natural Language Processing (NLP), computer vision, speech recognition, recommendation systems, and autonomous vehicles [8]. The study of Deep Learning and Artificial Neural Networks (ANNs) has evolved into a major subfield within AI, with applications extending to healthcare, law, finance, and scientific research [9].

The integration of AI technologies in higher education has progressed from a theoretical idea to a practical application. This shift reflects a global trend where universities are increasingly adopting AI-driven tools, such as Intelligent Tutoring Systems (ITS), adaptive learning algorithms, and Natural Language Processing (NLP) applications. These technologies aim to enhance instructional delivery and improve student learning outcomes [10]. They help increase student engagement and elevate the overall quality of education. ITS, which are now widely used in university settings, identify students' strengths and weaknesses, providing personalized feedback to support their academic progress.

This adaptive approach tailors learning content to each student's proficiency level. However, a significant gap persists in the literature and educational practice regarding effective methods and curriculum models for teaching NNs in higher education, particularly in developing countries such as Kazakhstan. While several studies discuss the theoretical capabilities of neural networks, few have addressed their pedagogical integration into computer science curricula.

It is necessary to identify effective approaches and key challenges in updating university curricula for teaching NNs to computer science students in accordance with modern requirements.

The research aims to evaluate contemporary approaches and challenges associated with teaching neural networks in higher education for computer science students. The following objectives have been outlined to achieve this aim:

- Explore educational programs for teaching neural networks and identify ways to improve them;
- Incorporate neural network-related subjects into

computer science teacher training programs and evaluate their effectiveness.

Based on these objectives, the study addresses the following research questions within the context of Kazakhstan's higher education system:

- RQ1: What is the significance of neural network training for computer science students, and which pedagogical approaches can be used to improve their learning outcomes?
- RQ2: What challenges may arise in neural network training for computer science students?
- RQ3: Which subjects and topics related to neural networks should be included in the computer science teacher education programs?
- RQ4: How effective is neural network instruction in developing the theoretical knowledge and technical competencies of computer science students?

II. Literature Review

A. University Programs

Leading universities around the world have increasingly incorporated NNs and DL into their curricula. One notable example is the Massachusetts Institute of Technology (MIT), which offers an Introduction to Deep Learning [11] course. This course focuses on DL techniques with applications in fields such as biology, computer vision, and natural language processing. Additionally, it introduces students to generative AI, Large Language Models (LLMs), and the foundational principles of constructing NNs.

At Stanford University, several courses provide in-depth training in DL and neural network applications. These include Machine Learning [12], Deep Learning for Computer Vision [13], and simply Deep Learning [14]. Through these courses, students gain proficiency in designing, configuring, and implementing NNs while exploring contemporary ML techniques and real-world applications.

The University of Oxford also offers various courses that explore NNs and DL, such as Deep Neural Networks [15], Machine Learning [16] and Physics-Informed Neural Networks [17]. These courses cover both supervised and unsupervised ML approaches and introduce students to the fundamental principles of contemporary multilayer NNs. Additionally, they employ physics-based neural network methods to tackle complex mathematical problems.

Similarly, the University of California, Berkeley, provides a range of DL-focused courses, including Introduction to Machine Learning [18], and Designing, Visualizing, and Understanding Deep Neural Networks [19]. These courses cover deep neural network structures, parameter optimization, generative and adversarial models, and advanced visualization techniques. They address both theoretical and applied aspects of ML, including time series analysis, Bayesian networks, probabilistic models, supervised and unsupervised learning, and dimensionality reduction. Students also engage in practical programming assignments and real-world applications.

A comparative analysis of neural network teaching practices at leading global institutions and top universities in Kazakhstan was conducted to evaluate the current state of neural network education (see Fig. 1).

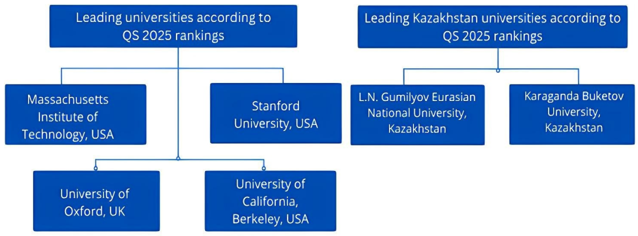

Fig. 1. Leading universities according to the QS 2025 rankings.

B. Implementation Challenges

In Kazakhstan, the Ministry of Science and Higher Education, in collaboration with Google, has introduced a new course titled "Generative AI" [20]. Initially piloted at 14 universities, this course will become a mandatory part of students' academic programs, underscoring the importance of foundational AI education at the university level. While institutions such as L. N. Gumilyov Eurasian National University and Karaganda Buketov University have introduced AI-related courses, specific coverage of NN topics remained relatively underdeveloped.

A curriculum review up to 2024 at L. N. Gumilyov Eurasian National University, Karaganda Buketov University, and several other institutions revealed the absence of dedicated neural network topics within their computer science programs. In response to this gap, and driven by the results of this study, neural network topics have been incorporated into the machine learning courses at these universities starting from 2025.

Given rapid global advancements in neural network technologies, there remains a critical need for comprehensive curricula emphasizing both theoretical foundations and practical applications. To effectively understand and apply neural network concepts, computer science students must possess competencies in algorithmic thinking, programming, systems architecture, linear algebra, calculus, probability, statistics, and foundational machine learning.

Similar to findings in other countries, challenges related to limited infrastructure and inadequate teacher training continue to hinder the effective integration of advanced AI topics in educational institutions [21].

C. Pedagogical Resources

Recent literature highlights both pedagogical and institutional challenges in implementing neural network education within higher education systems. For instance, a large-scale study involving 1,664 teachers revealed a substantial gap in AI-related content and technological knowledge among educators [22]. The findings confirmed that insufficient Technological Pedagogical Content Knowledge (TPACK) and a lack of training directly hinder effective AI instruction in educational settings.

In addition to these insights, a recent study emphasized that successfully introducing AI education—particularly regarding NNs—requires alignment between secondary and tertiary education [23]. This study outlines methodological challenges in structuring AI-related curricula and emphasizes the importance of establishing clear educational goals and scalable frameworks across educational levels.

Furthermore, Project-Based Learning (PBL) has been identified as a powerful strategy for improving classroom engagement and student motivation. Hugerat [24] demonstrated that PBL approaches in science education positively impacted the learning environment and student attitudes. Similarly, Darayseh and Mersin [25] emphasized the importance of teacher preparedness and mindset when integrating emerging technologies such as generative AI into STEM education. These studies underscore the need for well-structured, context-sensitive teaching strategies that align with both technological advancement and learner needs.

The fundamental concepts of NNs are clearly by Gurney [26], which does not rely on complex mathematical formalism. Similarly, Kubat [27] covers the essential concepts of ML and introduces key tools such as decision trees, NNs, Bayesian classifiers, linear and polynomial classifiers, as well as reinforcement learning techniques. Both of these books are crucial for the learning process, as they provide comprehensive and accessible coverage of fundamental concepts in NNs and ML.

D. Technological Tools

Numerous studies have investigated the use of Software tools and digital research equipment to support the application of NNs in educational settings. Haykin [28] covers essential topics such as artificial neurons, network behavior in pattern recognition, gradient descent techniques, associative memory, self-organization, and adaptive resonance theory, along with their scientific and commercial applications. Additionally, Chollet [29] provides an accessible yet rigorous overview of the core principles, historical development, practical applications, and future potential of AI, ML, and DL.

Jovanović *et al.* [30] presents a Java-based platform designed to facilitate the teaching of NNs. This platform features an intuitive user interface that enhances research and learning activities while also allowing for future expansion. Students can design and train NNs of various architectures and visualize fundamental concepts such as definitions, topologies, and training methods.

In addition to dedicated software platforms, project-based learning approaches that incorporate emerging technologies have also shown strong potential in enhancing students' technical competencies. For example, Tsai [31] demonstrated that hands-on Internet of Things (IoT) projects using devices like the micro:bit significantly improved preservice elementary teachers' understanding of complex systems and their confidence in using AI-related tools. This supports the notion that combining theoretical instruction with real-world applications can foster deeper engagement and skill acquisition in neural network education.

Several studies have explored the application of NNs and related software in educational contexts. For example, one study identified key factors influencing the adoption of AI-based chatbots in higher education institutions by using a hybrid PLS-SEM and neural network modeling approach, adapting the UTAUT2 model as the theoretical framework [32].

The research on hardware used for NNs is detailed by Badrani *et al.* [33] which explores how NNs can enhance educational processes through various hardware implementations. Abiodun *et al.* [34] placed this technology within a broader context, showcasing its diverse applications across multiple fields. It emphasizes that successful implementation of these technologies in education relies heavily on the availability of adequate computing resources. In a broader context, Abiodun *et al.* [34] highlights the wide applicability of NNs across various fields and emphasizes that their effective implementation in education depends on computational resources.

Several studies have been conducted in Kazakhstan to study the integration of software and hardware in teaching NNs. In Kazakhstan, recent initiatives have focused on utilizing high-performance parallel computing resources, such as the Param Bilim supercomputer at L. N. Gumilyov Eurasian National University [35]. These efforts aim to enhance neural network training and enable large-scale data processing in higher education institutions. The upgraded ParamBilim-2 supercomputer now features a GPU and has a peak performance of 1 petaflop. However, despite global advancements in this area, previous research indicates that there is a pressing need for systematic competency-based training for computer science teachers in Kazakhstan [36].

Following these recommendations, the findings suggest that incorporating practical assignments and interactive teaching methods into neural network education can be highly effective. These insights could contribute to revising and improving Computer Science students' training programs at Kazakhstani universities. Nonetheless, the integration of neural network training in Kazakhstan still faces both infrastructural and pedagogical challenges. This underscores the necessity for curriculum-aligned and context-sensitive teaching models at local universities.

III. Materials and Methods

The study utilized a mixed-methods approach, integrating theoretical, empirical, and statistical techniques. The theoretical phase involved a review and analysis of relevant scientific and methodological literature pertaining to NNs and ML education. The empirical component included classroom observations, surveys conducted among students, and the implementation of a pedagogical quasi-experiment to evaluate the effectiveness of the updated curriculum. To analyze the collected data both quantitatively and qualitatively, mathematical statistics methods, particularly Pearson's chi-square test, were employed, along with graphical representations of the results.

To facilitate the integration of NNs into the higher education curriculum, a specialized pedagogical model was designed and implemented as part of this study. This model aimed to improve the teaching of NNs to computer science students by incorporating current content, tools, and assessment strategies. It was developed in response to contemporary instructional needs and tested through quasi-experimental research in two universities. Before implementation, participating instructors attended training sessions on Python-based tools and Ad lib to ensure effective course delivery. The model included components such as curriculum content, system development technologies, and organizational practices to support learning outcomes (see Fig. 2).

Compared to the VLEEDP model [37], which emphasizes enhancing creative thinking through the engineering design process in a virtual learning environment, our proposed model focuses specifically on teaching NNs in higher education. It combines theoretical content, practical assignments, and modern AI tools to improve students' motivation, technical readiness, and mastery of the subject.

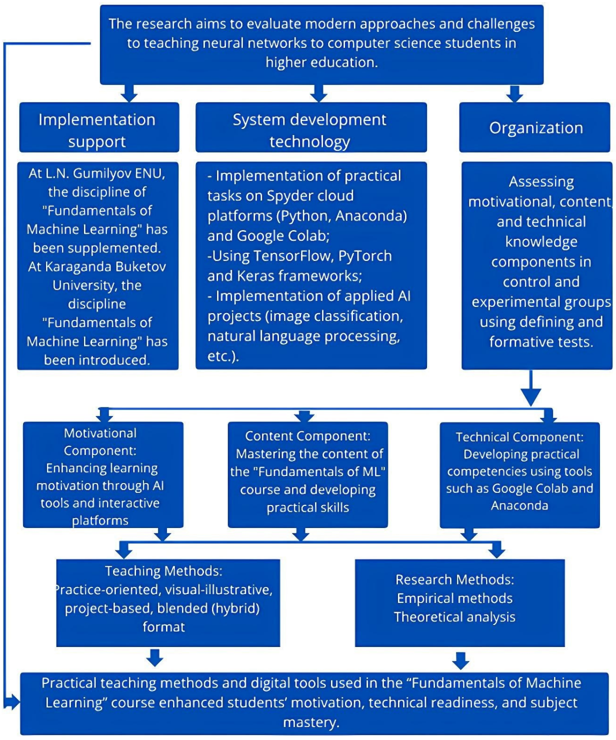

Fig. 2. The pedagogical model for teaching neural networks to computer science students in higher education.

While the model of Guzmán-Ramírez *et al.* [38] focused on teaching neural network design through simulations, our model offers a broader educational framework. It integrates theoretical concepts, hands-on activities, and widely-used AI tools, ultimately enhancing students' motivation and technical skills in higher education.

A. Implementation Support Part

A comparative analysis of syllabi from leading global institutions alongside the existing curriculum at L. N. Gumilyov Eurasian National University and Karaganda Buketov University led to a revision of the course "Fundamentals of Machine Learning" to better align with international best practices.

This update involved a thorough examination of course structures, recommended textbooks, and content from top universities, as well as insights from recent scholarly publications.

1) Curriculum design

As a result, several new topics were introduced, emphasizing DL, neural network architectures, and practical applications using modern tools. The revised course content is detailed in Table 1.

In the "Fundamentals of Machine Learning" course for the Computer Science program, Python was chosen as the most suitable programming language due to its extensive ecosystem of libraries and tools commonly used in ML. Two programming environments were utilized: Google Colab, a cloud-based platform that allows for the execution of computationally intensive tasks without the need for local installation, and Spyder, a desktop-based Integrated Development Environment (IDE) that offers advanced debugging and editing features. These platforms were selected because of their user-friendliness and effectiveness in facilitating both instructional delivery and hands-on experimentation.

Table 1. A list of lectures and practical lessons of the course "Fundamentals of Machine Learning"

No	Lectures and practical topics	Types and methods of teaching
1	ML—artificial intelligence	Explanation—illustrative method
2	Types of ML problems	Explanation—illustrative method
3	Basic ML methods. Bayesian classification. Bayesian formula. Pattern recognition.	Explanation—illustrative method
4	Basic ML methods. Supervised learning. Classification. Regression.	Blended learning—method of explanatory and programmatic (computer) learning.
5	Basic ML methods. Unsupervised learning. Clustering. Dimension reduction.	Explanation—illustrative method
6	ML and tensor processor TPU. Tensor processor TPU. Overview of popular libraries: TensorFlow, PyTorch, Keras.	Explanation—illustrative method
7	Basic methods of ML. NNs. Neuron transfer functions. Neuron classification.	Programmatic (computer) learning
8	ML and DL. NNs. Trends and future of NNs. Fundamentals of building NNs. Inputs and outputs of a neuron.	Explanation—illustrative method
9	ML and DL. NNs. Fundamentals of building multidimensional NNs. Reducing losses in NNs. Methods. Differentiating a complex function in calculations.	Blended learning—method of explanatory and programmatic (computer) learning.
10	Basic methods of ML. Application of NNs in real problems. Examples of successful projects: Alpha Go, DALE, GPT-3. Ethical and social aspects of the use of NNs.	Explanation—illustrative method
11	ML and big data. Methods of using ML algorithms in processing big data. Big data analysis, analytics, Data Mining, and features of ML.	Programmatic (computer) learning
12	ML and proctoring system. Steps of human face recognition and methods of building artificial NNs. OpenCV cascaded Haar classifier. Mediapipe face recognition.	Programmatic (computer) learning
13	Algorithm for predicting human emotions. An algorithm for detecting human emotions using the Face Emotion Recognizer library in Google Colab.	Programmatic (computer) learning
14	ML and preprocessing system. OpenCV library.	Explanation—illustrative method
15	Quantum computers and ML methods.	Explanation—illustrative method

2) Theory

Throughout the semester, students studied five core topics related to NNs. They learned that ML is a crucial subfield of AI and explored its mathematical foundations along with related disciplines. The topics covered included Bayesian classification, supervised and unsupervised learning methods, regression, clustering, and essential ML techniques. Additionally, students engaged with widely-used libraries and frameworks such as Tensor Processing Units (TPUs), TensorFlow, PyTorch, and Keras, which are instrumental in developing modern AI applications. As a result, the students gained a solid understanding of techniques to enhance ML models and visualize the learning process.

In the neural network modules, students explored the fundamental structures and operational principles of ANNs. They developed proficiency in activation functions, neuron classification, and techniques for minimizing loss functions.

The course also covered the theoretical foundations of constructing NNs, training multidimensional networks, and applying stochastic gradient descent.

3) Practice

Students analyzed practical examples, including real-world projects like AlphaGo, DALL·E, and GPT-3, to illustrate advanced implementations. Additionally, the curriculum included topics such as big data processing, ML algorithms used in proctoring systems, emotion recognition, and object detection with OpenCV. Emerging areas like Quantum Machine Learning (QML) were also discussed to familiarize students with cutting-edge applications of ML. This approach helped them understand various AI use cases and encouraged critical thinking about the ethical, technical, and social implications of technology.

This comprehensive curriculum enables students to grasp both the theoretical concepts and practical applications of ML and NNs. An educational portal titled "Courses for the Training of Future Computer Science Teachers: Machine Learning, Network Technologies, and Information Security" [39] (see Fig. 3) features specialized modules, such as "Computer Vision" and "Software Tools for Practical Work in Machine Learning with Python". The portal provides registered users with access to course content, allowing them to review lecture materials, complete assignments, and engage in additional tasks at their own pace. This digital platform encourages independent learning and reinforces the skills acquired during in-person instruction. The course platform can be accessed at https://security.org.kz/. Users must register to access the course materials.

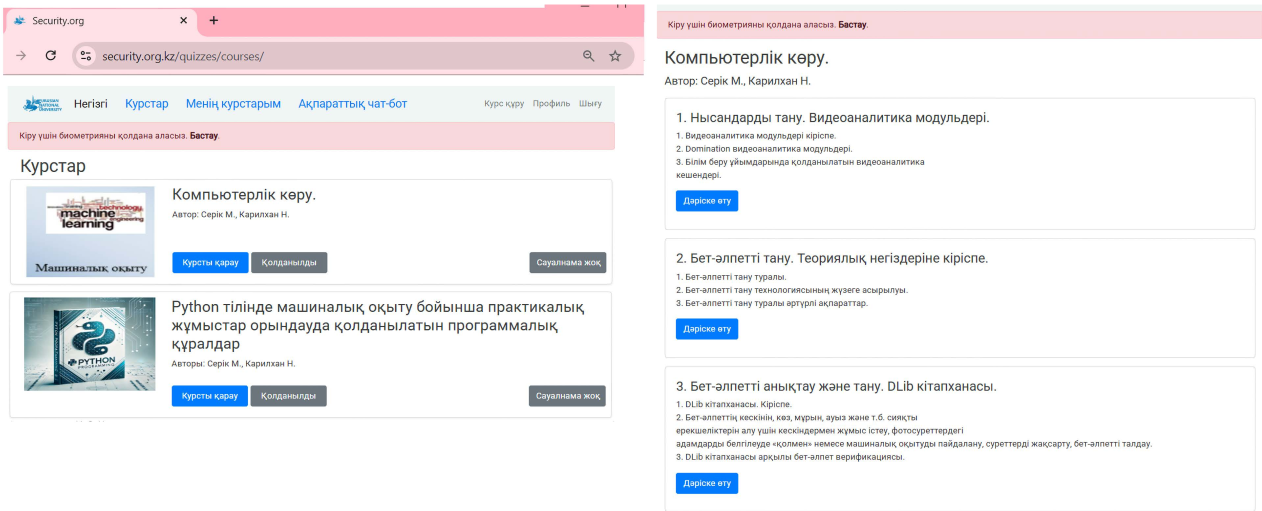

Fig. 3. Information and educational portal "Courses for the Training of Future Computer Science Teachers: Machine Learning, Network Technologies, and Information Security" [39].

B. System Development Technology

Recent advancements in information technology have significantly enhanced the development of face recognition systems. These algorithms are now widely employed across various fields, including security, healthcare diagnostics, and education. This study incorporates practical assignments completed by students using the DLib library for face recognition tasks.

During the practical sessions, students gained hands-on experience using Spyder and the DLib library to implement face detection and recognition algorithms. The tasks involved facial detection, feature extraction, and identity matching between facial images, as illustrated in Fig. 4.

The program utilizes two pre-trained models-shape_predictor_68_face_landmarks.dat and dlib_face_recognition_resnet_model_v1.dat—which are essential for detecting facial landmarks and extracting numerical feature descriptors. When a facial image (e.g., nur.jpeg) is uploaded, the face is detected, and its descriptors are generated. This process is repeated for a second image (e.g., nur_foto.jpeg or doc1.jpg).

After obtaining the descriptors for both images, the Euclidean distance between them is calculated. If the distance is less than 0.6, the images are considered to belong to the same individual; if the distance is greater than 0.6, they are regarded as representing different individuals. The DLib face recognition model employs a ResNet-34-based CNN to extract 128-dimensional face descriptors. A threshold of 0.6 for Euclidean distance is commonly used for identity verification, as supported by previous validation studies [40].

```python
import dlib
from skimage import io
from scipy.spatial import distance
sp = dlib.shape_predictor('shape_predictor_68_face_landmarks.dat')
facerec=dlib.face_recognition_model_v1('dlib_face_recognition_resnet_model_v1.dat'
detector = dlib.get_frontal_face_detector()
img = io.imread('nur.jpeg')
win1 = dlib.image_window()
win1.clear_overlay()
win1.set_image(img)
dets = detector(img, 1)
for k, d in enumerate(dets):
    print("Detection {}: Left: {} Top: {} Right: {} Bottom: {}".format(
        k, d.left(), d.top(), d.right(), d.bottom()))
    shape = sp(img, d)
    win1.clear_overlay()
    win1.add_overlay(d)
    win1.add_overlay(shape)
    face_descriptor1 = facerec.compute_face_descriptor(img, shape)
    print(face_descriptor1)
img = io.imread('nur_foto.jpeg')
win2 = dlib.image_window()
win2.clear_overlay()
win2.set_image(img)
dets_webcam = detector(img, 1)
for k, d in enumerate(dets_webcam):
    print("Detection {}: Left: {} Top: {} Right: {} Bottom: {}".format(
        k, d.left(), d.top(), d.right(), d.bottom()))
    shape = sp(img, d)
    win2.clear_overlay()
    win2.add_overlay(d)
    win2.add_overlay(shape)
    face_descriptor2 = facerec.compute_face_descriptor(img, shape)
a = distance.euclidean(face_descriptor1, face_descriptor2)
print(a)
```

Fig. 4. Code for face recognition using the DLib library.

As part of the experiment, students analyzed the performance of a face recognition algorithm and interpreted its results. For instance, a Euclidean distance of 0.383 between two images of the same individual confirmed that they matched, while a distance of 0.7942 between images of two different individuals demonstrated the algorithm's ability to distinguish between them (see Fig. 5). These outcomes illustrate the practical application of ML techniques in facial recognition.

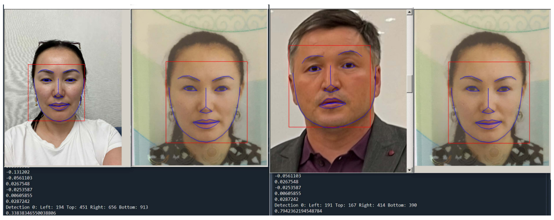

Fig. 5. Face recognition comparison based on image similarity.

Through this process, students explored real-world applications of AI and computer vision, developed practical skills, and gained the ability to evaluate algorithm performance and understand how facial recognition systems operate in practical contexts. Their competencies were assessed through practical assignments involving facial recognition tasks using the DLib library, where they implemented identity matching, measured Euclidean distances, and interpreted outcomes based on established accuracy thresholds (e.g., the 0.6 distance rule). Evaluation rubrics were used to assess their technical proficiency, including successful script execution, result interpretation, and model optimization. Having completed the implementation phase, the next step was to evaluate the model's effectiveness through a structured experimental design.

C. Organization

In teaching the discipline "Fundamentals of Machine Learning", we used Bloom's taxonomy to establish clear learning objectives and ensure cognitive progress. Bloom's taxonomy outlines six main categories—knowledge, understanding, application, analysis, synthesis, and evaluation—arranged from simple to complex. This structure supports the gradual development of students' critical and analytical skills [41].

A quasi-experimental design was employed to evaluate the effectiveness of the updated curriculum. Participants were assigned nonrandomly to either an experimental group or a control group based on their existing enrollment, which ensured ecological validity. Both groups were taught by the same instructor over a 15-week semester using standardized materials.

The study involved 85 third-year Informatics students from L.N. Gumilyov Eurasian National University and Karaganda Buketov University. The experimental group, comprising 42 students, participated in a course on the Fundamentals of Machine Learning, based on an updated curriculum. In contrast, the control group, which included 43 students, followed a traditional program.

Table 2. Data from the participants of the experiment

University Name	Experimental group	Control group	Number of participants
L.N. Gumilyov ENU	22	23	45
Karaganda BU	20	20	40
Total	42	43	85

To maintain appropriate group sizes and ensure ecological validity, the study adhered to typical academic group sizes in Kazakhstan, which range from 10 to 25 students. The experimental group consisted of 22 students from one university and 20 from the other. Meanwhile, the control group comprised 23 students from one university and 20 from the other (see Table 2).

The experimental group of students reported significant improvements in their spatial reasoning skills, as well as a deeper understanding of NNs and ML.

D. Summative Phase of the Experiment

The Pearson chi-square (χ^2) test was selected to examine the research hypotheses because the study included more than 30 participants. This test is particularly suitable for identifying statistically significant associations between the three evaluation components—motivational, content, and technical—and the three performance levels: high, medium, and low.

In the final stage of the experiment, the course was tested with students, and its effectiveness was evaluated through statistical analysis of data from both the experimental and control groups. During this phase, a structured knowledge-based test was administered at the beginning and at the end of the course to assess its effectiveness. The test items were categorized into three main evaluation categories: motivational, content, and technical components, as detailed in Table 3. Each category consisted of four questions and was designed for both diagnostic and formative assessment purposes.

To verify the research hypotheses, Pearson's chi-square (χ^2) test was employed as the primary analysis method. This test was chosen because the study involved more than 30 participants and aimed to examine statistically significant associations across the three evaluation criteria (motivational, content-related, and technical) and three performance levels (high, medium, and low) in line with the research hypothesis. The results indicated a positive impact of the updated course on students' theoretical and technical competencies.

Table 3. Test item categories for evaluating knowledge components (motivational, content, technical) via defining and formative assessment

Motivation component	Content component	Technical component
Determining the interest of future teachers in NNs and AI technologies. Examining neural networks' effectiveness in the educational process	Assessing the level of understanding and mastering the basic concepts of NNs and ML theoretical and practical materials using interactive methods	Assessing the level of preparation for working with hardware and software for training and developing neural networks. Determining the qualification in working with tools such as Google Colab, TensorFlow, Keras, and OpenCV
1. What was the reason for studying neural networks? 2. What motivates you to continue studying neural networks? 3. How are you planning to use knowledge of neural networks in the future? 4. What was the most interesting part of studying neural networks?	1. What kind of neural networks are you familiar with? 2. What kind of problems are best solved by neural networks? 3. What kind of neural network teaching methods are you familiar with? 4. What are the possible challenges in studying neural networks?	1. What platform do you use to train neural networks? 2. What programming language do you use to train neural networks? 3. Which libraries do you think are the most convenient for developing neural networks? 4. How confident are you in your neural network programming skills?

IV. Results

The study assessed the impact of the revised curriculum by analyzing three core components: motivation, content comprehension, and technical proficiency. Each component was measured using four targeted questionnaire items. Students' responses were categorized into three performance levels: low (0–49), medium (50–84), and high (85–100). The analysis included both observed (empirical) and expected (theoretical) frequency distributions, accompanied by a comparative summary table. This method allowed for a comprehensive evaluation of how instructional changes influenced students' engagement with NNs, mastery of ML concepts, and readiness to utilize advanced AI tools such as Google Colab, TensorFlow, Keras, and OpenCV.

To evaluate motivation levels, frequency distributions were calculated separately for the control and experimental groups.

At the low motivation level, the control group had 14 out of 43 students (approximately 32.6%), while the expected theoretical count was only 8.6. In contrast, the experimental group had just 3 students (7.1%) in this category, significantly lower than the expected 8.4. This suggests that the revised instructional approach contributed to a notable decrease in the number of students with low motivation.

For the medium motivation level, both groups closely matched the theoretical projections. The control group recorded 25 students (compared to an expected 26.31), while the experimental group had 24 students (expected = 25.69), indicating that most participants fell into the moderate motivational category.

However, the high motivation level revealed significant differences. Only 4 students (9.3%) in the control group achieved this level, falling short of the expected value of 8.09. In contrast, the experimental group recorded 12 students (28.6%) achieving high motivation, surpassing the expected 7.91. These results highlight a positive shift in motivation for students exposed to the revised curriculum, particularly in fostering higher engagement.

In result, the comparative analysis between actual and theoretical distributions demonstrates a statistically significant improvement in motivation among students in the experimental group. The use of modern instructional strategies and AI-integrated tools played a crucial role in reducing low motivation and encouraging higher levels of engagement.

A detailed comparison of empirical and theoretical frequencies is presented in Table 4.

Table 4. Motivation component summary

Groups	Levels	Empirical	Theory	$(E - T_{cr})^2/T_{cr}$
Students in control group	Low level	14	8.60	3.39
	Medium level	25	26.31	0.06
	High level	4	8.09	2.07
Students in the experimental group	Low level	3	8.40	3.47
	Medium level	27	25.69	0.07
	High level	12	7.91	2.12
Total ($\chi^2_{Motivation\ component}$)		11.18		

The second category, content mastery, was evaluated using a three-level scoring scale: low (0–49), medium (50–84), and high (85–100). Frequency distributions were calculated for both empirical observations and theoretical expectations.

At the low content level, the control group recorded 16 out of 43 students (approximately 37.2%), which is significantly higher than the expected value of 9.11. In contrast, the experimental group had only 2 students (approximately 4.8%) in this category, which is well below the expected frequency of 8.89. These results suggest that the revised curriculum has substantially reduced the number of students demonstrating weak understanding of the course content.

At the medium level, both groups exhibited patterns similar to the theoretical expectations. The control group included 24 students (expected = 26.81), while the experimental group had 29 students (expected = 26.19), indicating a stable performance range for the majority of learners.

At the high content mastery level, clear differences emerged. In the control group, only 3 students (approximately 7.0%) achieved high-level performance, whereas the expected number was 7.08. On the other hand, the experimental group had 11 students (approximately 26.2%) reaching the high level, surpassing the expected frequency of 6.92. This result demonstrates the effectiveness of integrating practical tools and updated content delivery to enhance students' comprehension of ML topics.

Overall, the analysis confirms that the revised instructional model resulted in a significant improvement in content mastery. Fewer students remained at the low level, while more advanced learners achieved high performance. These differences are further detailed in Table 5.

Table 5. Content component summary

Groups	Levels	Empirical	Theory	$(E - T_{cr})^2/T_{cr}$
Students in control group	Low level	16	9.11	5.22
	Medium level	24	26.81	0.29
	High level	3	7.08	2.35
Students in the experimental group	Low level	2	8.89	5.34
	Medium level	29	26.19	0.30
	High level	11	6.92	2.41
Total ($\chi^2_{Content\ component}$)			15.92	

The third component assessed was technical readiness, utilizing a scoring scale of low (0–49), medium (50–84), and high (85–100). The analysis compared observed (empirical) frequencies with expected (theoretical) values for both the control and experimental groups.

At the low level of technical skills, there was a sharp contrast between the groups. In the control group, 15 out of 43 students (approximately 34.9%) demonstrated low technical readiness, significantly exceeding the theoretical expectation of 8.09. In contrast, only 1 student (approximately 2.4%) in the experimental group fell into this category, which was well below the expected frequency of 7.91. This finding suggests that the implemented instructional strategies effectively addressed technical weaknesses.

For the medium level of technical readiness, the distribution of students aligned closely with theoretical estimates. The control group had 26 students in this category (expected = 28.84), while the experimental group had 31 students (expected = 28.16). This consistency shows that the majority of students in both groups possessed moderate technical proficiency.

More notably, at the high technical readiness level, the experimental group outperformed the control group. In the control group, only 2 students (approximately 4.7%) achieved high proficiency, which was below the expected value of 6.07. Meanwhile, 10 students (approximately 23.8%) in the experimental group reached this level, surpassing the expected count of 5.93. These results emphasize the positive impact of using platforms such as Google Colab, along with tools like TensorFlow and OpenCV, in enhancing students' technical competencies.

In result, the analysis of the technical component highlights a clear improvement in students' practical skills and readiness to work with AI tools as a result of the revised curriculum. A detailed breakdown of these findings is provided in Table 6.

Table 6. Technical component summary

Groups	Levels	Empirical	Theory	$(E - T_{cr})^2/T_{cr}$
Students in control group	Low level	15	8.09	5.89
	Medium level	26	28.84	0.28
	High level	2	6.07	2.73
Students in the experimental group	Low level	1	7.91	6.03
	Medium level	31	28.16	0.29
	High level	10	5.93	2.79
Total ($\chi^2_{Technical\ component}$)			18.01	

To compare the outcomes between the control and experimental groups, we used a non-parametric Pearson's chi-squared (χ^2) test [42]. The null hypothesis (H_0) posited that there is no statistically significant association between the independent and dependent variables. In contrast, the alternative hypothesis (H_1) suggested the existence of a significant relationship.

The chi-square statistic was calculated using the appropriate formula:

$$\chi^2 = \frac{\sum|(Fe - Fb)^2|}{Fb} \quad (1)$$

In this analysis, *Fe* represents the relative frequency for the experimental group, while *Fb* denotes the relative frequency for the control group. The degrees of freedom were calculated using the formula: $df = (R - 1) \times (C - 1)$, where R indicates the number of rows and C represents the number of columns. In this instance, with 3 rows and 3 columns, the degrees of freedom equal 4. At a significance level of 0.05, the critical chi-square value (χ^2_{cr}) is 9.49. The calculated chi-square values for each component are as follows.

$$\chi^2_{Motivation\ component} = 11.18 > 9.49$$

$$\chi^2_{Content\ component} = 15.92 > 9.49$$

$$\chi^2_{Technical\ component} = 18.01 > 9.49$$

According to the decision rule, the null hypothesis (H_0) is rejected in favor of the alternative hypothesis (H_1) if the calculated value of χ^2 for any component exceeds the critical value ($\chi^2 > \chi_{cr}$). In this study, all three calculated χ^2 values surpassed the threshold, leading to the rejection of H_0 for each component. In this study, all three calculated χ^2 values surpassed the threshold, leading to the rejection of H_0 for each component [43].

This indicates that the differences between the control and experimental groups regarding student motivation, content comprehension, and technical readiness are statistically significant, thereby validating the effectiveness of the revised instructional model.

The test was administered using Google Forms, and the results are summarized in Table 7 and illustrated in Fig. 6.

Table 7. Results of the experiment

Factors and Indicators Influencing Learners' Motivation to Understand Neural Networks in Computer Science Education	Low level		Medium level		High level	
	control group (%)	experimental group (%)	control group (%)	experimental group (%)	control group (%)	experimental group (%)
Motivation component	32.6	7.1	58.1	64.3	9.3	28.6
Content component	37.2	4.8	55.8	69.0	7.0	26.2
Technical component	34.9	2.4	60.5	73.8	4.7	23.8

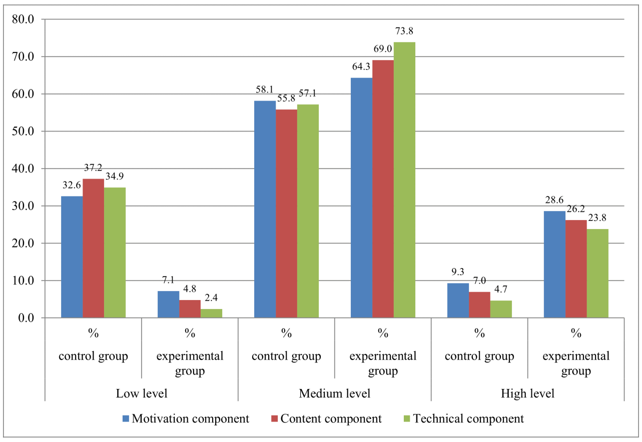

Fig. 6. Results of the experiment.

V. Discussion

These findings suggest that the proposed instructional approach for teaching ML and NNs may be effective. By comparing control and experimental groups, the study assessed three core aspects of student development: motivation, content comprehension, and technical proficiency.

A key outcome was the enhanced motivation observed in the experimental group, where 28.6% of students exhibited high levels of motivation compared to only 9.3% in the control group. This result underscores the positive impact of integrating interactive teaching methods and real-world problem-solving scenarios in neural network education.

In terms of academic content, students exposed to the updated curriculum demonstrated a considerably better understanding of neural network concepts. Specifically, 26.2% of students in the experimental group achieved a high level of content component, compared to only 7.0% in the control group. These findings align with prior research indicating that engaging with advanced AI technologies and real-world examples can improve students' theoretical understanding and their capacity to contextualize complex concepts in AI.

The analysis of technical skills revealed significant improvement among students in the experimental group, with 23.8% performing well in practical tasks. In contrast, only 4.7% of students in the control group achieved similar results. This improvement aligns with the principles of constructivist learning theory, which states that active engagement and immediate feedback enhance deeper conceptual understanding and skill retention. The observed gains also reflect the concepts of self-determination theory, highlighting autonomy, competence, and relatedness as essential factors that drive intrinsic motivation.

While the results are encouraging, they should be interpreted with caution due to the study's limited scope and duration. Nonetheless, these findings highlight the necessity for continuous modernization of academic curricula. Many leading global institutions have already integrated deep learning into their computer science programs, establishing a benchmark for contemporary AI education. In this context, higher education institutions in Kazakhstan are urged to adopt similar innovations, taking into account local educational contexts and infrastructure to ensure both global relevance and contextual effectiveness. Some earlier studies have raised concerns about the scalability and long-term effectiveness of technology-driven approaches in low-resource settings, highlighting issues related to implementation fidelity and instructor preparedness [21]. The observed improvements can be attributed to integrating collaborative, problem-based tasks that enhance active engagement and increase intrinsic motivation among students.

These findings suggest that the use of interactive, technology-enhanced instructional approaches provides significant pedagogical advantages in neural network education. Collaborative projects, AI-based tools, and digital platforms contribute to improved academic outcomes. The successful implementation of Intelligent Tutoring Systems (ITS) in the future will require further investigation into adaptive instructional strategies tailored to individual learning profiles. These findings support earlier studies suggesting that project-based learning enhances student motivation and fosters a deeper understanding of complex topics [24], particularly in programming and Internet of Things (IoT) [31] contexts.

While the results are promising, several limitations should be considered to better understand the findings. First, the quasi-experimental design utilized pre-existing groups without random assignment, which may compromise internal validity and limit the generalizability of the outcomes. Second, the sample consisted only of third-year students from two universities, which reduces its representativeness across different institutional contexts. Third, the intervention occurred over a single academic semester, limiting insights into the long-term effects of the updated curriculum. Finally, although cloud-based platforms like Google Colab helped alleviate hardware constraints, students encountered challenges such as inconsistent internet access and varying levels of digital literacy, which could have impacted their learning outcomes.

These challenges align with broader findings in the literature, which indicate that while STEM educators generally express positive attitudes toward AI in education, they encounter barriers related to usability and implementation. Without proper training and institutional support, the full potential of AI-based instructional tools may remain untapped, particularly in low-resource settings [25].

Ultimately, systematically incorporating neural network instruction into teacher education programs will elevate the quality of computer science instruction. For Kazakhstan, adopting these methods necessitates not only curricular changes but also specific professional development programs for educators. This ensures that global best practices are tailored to address local infrastructure, language, and pedagogical needs effectively.

VI. Conclusion

This study aimed to assess modern approaches and challenges in teaching neural networks to computer science students in higher education through the following objectives.

The study initially focused on analyzing and enhancing educational programs for teaching NNs by examining the practices of leading international universities and current scientific research. Consequently, topics related to the development of NNs were incorporated into the curriculum, and a pedagogical model was created to effectively teach NNs to computer science students.

Moreover, the effectiveness of this pedagogical model was validated through its practical implementation in universities.

The research questions formulated within this framework led to the following findings:

RQ1: Neural network training significantly enhances the motivation and academic performance of computer science students. The use of interactive and practice-oriented teaching strategies encourages active participation and contributes to improved academic outcomes.

RQ2: During the implementation of the neural network training process, several challenges were encountered. Given that this is one of the fastest-growing areas in information technology, the curricula for computer science students are not keeping pace with these developments. Additionally, the content of manuals and online platforms also requires constant updates. In response to this issue, a training platform has been developed that allows for yearly updates. Currently, this platform includes theoretical lectures and practical exercises related to the course "Fundamentals of Machine Learning".

RQ3: The subject "Fundamentals of Machine Learning" was implemented for Computer science students. The curriculum addressed key topics in machine learning and neural networks, emphasizing their ethical implications and applications in emerging technologies.

RQ4: The effectiveness of the updated curriculum was assessed using a test designed to evaluate and enhance students' motivation, content knowledge, and technical skills. The overall effectiveness of the model was validated using Pearson's chi-square test. Students in the experimental group demonstrated significantly higher achievement levels across all evaluated components, confirming the effectiveness of the revised, tool-based instructional approach.

The methods and platform described in this model can be used by any higher education institution to teach computer science students.

This study is limited in both its institutional scope and the duration of implementation. Future research may explore long-term impacts, perform cross-institutional comparisons, and investigate the integration of neural network instruction with other AI-related disciplines.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

K. N. and N. D. were responsible for conceptualizing the study, conducting the data analysis, and drafting the original manuscript. S. M. and K. J. provided supervision and methodological guidance, and they contributed to the review and editing of the manuscript. All authors have read and approved the final version of the manuscript.

Funding

This research is part of the grant project AP23489632, which is funded by the Ministry of Science and Higher Education of the Republic of Kazakhstan. The project is titled "Theoretical and Practical Foundations for the Comprehensive Improvement of Computer Science Teacher Training Based on STEM Education and Machine Learning."

References

[1] J. M. Brown *et al*., "Automated diagnosis of plus disease in retinopathy of prematurity using deep convolutional neural networks," *JAMA Ophthalmology*, vol. 136, no. 7, pp. 803–810, 2018. https://doi.org/10.1001/jamaophthalmol.2018.1934

[2] I. Maglogiannis, L. Iliadis, J. Macintyre, M. Avlonitis, and A. Papaleonidas, *Artificial Intelligence Applications and Innovations*, Springer Nature, 2024. https://doi.org/10.1007/978-3-031-63223-5

[3] R. Y. Choi, A. S. Coyner, J. Kalpathy-Cramer, M. F. Chiang, and J. P. Campbell, "Introduction to machine learning, neural networks, and deep learning," *Transl. Vis. Sci. Technol*., vol. 9, no. 2, 14, 2020. https://doi.org/10.1167/tvst.9.2.14

[4] S. J. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 3rd ed., Pearson, 2016. https://doi.org/10.1167/tvst.9.2.14

[5] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, Cambridge, MA, USA: MIT Press, 2016.

[6] C. Aggarwal, "An introduction to neural networks,"*Neural Networks and Deep Learning*, Cham: Springer, 2023. https://doi.org/10.1007/978-3-319-94463-0_1

[7] L. Chand, S. S. Cheema, and M. Kaur, "Understanding neural networks,"*Factories of the Future: Technological Advancements in the Manufacturing Industry*, pp. 83–102, 2023. https://doi.org/10.1002/9781119865216.ch4

[8] H. Taherdoost, "Deep learning and neural networks: Decision-making implications," *Symmetry*, vol. 15, no. 9, 1723, 2023. https://doi.org/10.3390/sym15091723

[9] D. J. Santry, *Demystifying Deep Learning: An Introduction to the Mathematics of Neural Networks*, John Wiley & Sons, Ltd., United Kingdom, 2023. https://doi.org/10.1002/9781394205639

[10] D. Kazimova *et al*., "Transforming university education with AI: A systematic review of technologies, applications, and implications," *Int. J. Eng. Pedagogy (iJEP)*, vol. 15, no. 1, pp. 4–24, 2025. https://doi.org/10.3991/ijep.v15i1.50773

[11] *Introduction to Deep Learning*. [Online]. Available: https://introtodeeplearning.com

[12] *CS229: Machine Learning*. [Online]. Available: https://cs229.stanford.edu

[13] *CS231n: Convolutional Neural Networks for Visual Recognition*. [Online]. Available: https://cs231n.stanford.edu

[14] *CS230: Deep Learning*. [Online]. Available: https://cs230.stanford.edu/

[15] *Deep Neural Networks Course, University of Oxford*. [Online]. Available: https://www.cs.ox.ac.uk/softeng/subjects/DNN.html

[16] *Machine Learning Course, University of Oxford*. [Online]. Available: https://www.cs.ox.ac.uk/teaching/courses/2024-2025/ml/

[17] *Physics-Informed Neural Networks Course, University of Oxford*. [Online]. Available: https://www.cs.ox.ac.uk/teaching/courses/2024-2025/pinn/

[18] *CS189: Machine Learning, UC Berkeley*. [Online]. Available: https://www2.eecs.berkeley.edu/Courses/CS189/

[19] *CS282A: Advanced Machine Learning, UC Berkeley*. [Online]. Available: https://www2.eecs.berkeley.edu/Courses/CS282A

[20] *Kazakhstan Science and Innovation News*. [Online]. Available: https://www.gov.kz/memleket/entities/sci/press/news/details/840197?lang=kk

[21] O. Atabek, "Challenges in integrating technology into education," arXiv preprint, arXiv:1904.06518, 2019.

[22] M. Yue, M. S. Y. Jong, and D. T. K. Ng, "Understanding K–12 teachers' technological pedagogical content knowledge readiness and attitudes toward artificial intelligence education," *Education and Information Technologies*, vol. 29, pp. 19505–19536, 2024. https://doi.org/10.1007/s10639-024-12621-2

[23] I. V. Levchenko, A. R. Sadykova, L. I. Kartashova, and P. A. Merenkova, "Teaching artificial intelligence in secondary school: From development to practice," *RUDN Journal of Informatization in Education*, vol. 20, no. 3, pp. 265–280, 2023. https://doi.org/10.22363/2312-8631-2023-20-3-265-280

[24] M. Hugerat, "How teaching science using project-based learning strategies affects the classroom learning environment," *Learn. Environ. Res*., vol. 19, no. 3, pp. 383–395, 2016. https://doi.org/10.1007/s10984-016-9212-y.

[25] A. Al-Darayseh and N. Mersin, "Integrating generative AI into STEM education: Insights from science and mathematics teachers," *International Electronic Journal of Mathematics Education*, vol. 20, no. 3, em0832, 2025. https://doi.org/10.29333/iejme/16232

[26] K. Gurney, *An Introduction to Neural Networks*, 1st ed., CRC Press, 1997. https://doi.org/10.1201/9781315273570

[27] M. Kubat, *An Introduction to Machine Learning*, Cham: Springer, 2017. https://doi.org/10.1007/978-3-319-63913-0

[28] S. Haykin, *Neural Networks: A Comprehensive Foundation*, Englewood Cliffs, NJ: Prentice Hall PTR, 1994.

[29] F. Chollet, *Deep Learning with Python*, 2nd ed., Simon and Schuster, 2021.

[30] N. Jovanović, S. Stamenković, and S. Jovanović, "NNeduca: A software environment to teach artificial neural networks," *Education and Information Technologies*, 2024. https://doi.org/10.1002/cae.22655

[31] F.-H. Tsai, "Development and evaluation of an internet of things project for preservice elementary school teachers," *Sustainability*, vol. 16, no. 17, 7632, 2024. https://doi.org/10.3390/su16177632

[32] N. I. M. Rahim *et al*., "AI-based chatbots adoption model for higher-education institutions: A hybrid PLS-SEM-neural network modeling approach," *Sustainability*, vol. 14, no. 19, 12726, 2022. https://doi.org/10.3390/su141912726

[33] M. Badrani *et al*., "Personalized guidance for moroccan students: An approach based on machine learning and big data," *Int. J. Eng. Pedagogy (iJEP)*, vol. 15, no. 1, pp. 125–136, 2025. https://doi.org/10.3991/ijep.v15i1.51985

[34] O. I. Abiodun *et al*., "State-of-the-art in artificial neural network applications: A survey," *Heliyon*, vol. 4, no. 11, 2018. https://doi.org/10.1016/j.heliyon.2018.e00938

[35] N. Karelkhan *et al*., "High-Performance parallel computing on the param bilim supercomputer in higher education," *World Trans. Eng. Technol. Educ*., vol. 22, no. 2, 2024.

[36] D. A. Kazimova, "Conditions for training future computer science teachers based on a systems approach," *Bull. Karaganda Univ., Pedagogy Ser*., vol. 11529, no. 3, pp. 6–13, 2024. https://doi.org/10.31489/2024ped3/6-13

[37] P. Thammaariyasakun, W. Napapongs, J. Tansakul, and C. Inkaew, "Development of a virtual learning environment with the engineering design process to enhance students' creative thinking skills," *International Journal of Information and Education Technology*, vol. 15, no. 1, pp. 137–147, Jan. 2025. https://doi.org/10.18178/ijiet.2025.15.1.2226

[38] E. Guzmán-Ramírez, I. Garcia, and M. García-Juárez, "A 'learning by design' application for modeling, implementing, and evaluating hardware architectures for artificial neural networks at undergraduate level," *Computer Applications in Engineering Education*, vol. 27, no. 5, pp. 1236–1252, 2019. https://doi.org/10.1002/cae.22148

[39] Information Educational Portal. *Courses for the Training of Future Computer Science Teachers*. [Online]. Available: https://security.org.kz

[40] H. R. Chou, J. H. Lee, Y. M. Chan, and C. S. Chen, "Data-specific adaptive threshold for face recognition and authentication," in *Proc. IEEE Conf. Multimedia Inf. Process. Retrieval (MIPR)*, Mar. 2019, pp. 153–156. https://doi.org/10.1109/MIPR.2019.00034

[41] S. R. Sobral, "Bloom's taxonomy to improve teaching-learning in introduction to programming," *International Journal of Information and Education Technology*, vol. 11, no. 3, pp. 148–153, Mar. 2021. https://doi.org/10.18178/ijiet.2021.11.3.1504

[42] M. I. Grabar and K. A. Krasnianskaya, *Application of Mathematical Statistics in Pedagogical Research*, Moscow: Pedagogy, p. 136, 1977.

[1] N. Karelkhan and N. Uderbayeva, "The effectiveness of using virtual and augmented reality technologies for teaching computer science in schools," *Int. J. Inf. Educ. Technol*., vol. 14, no. 11, 2024. https://doi.org/10.18178/ijiet.2024.14.11.2187