E-learning in the Transformation of Education in Digital Society
Scientific Editor Eugenia Smyrnova-Trybulska
"E-learning", 14, Katowice–Cieszyn 2022, pp. 15–28
https://doi.org/10.34916/el.2022.14.02

PROPOSAL OF ARTIFICIAL INTELLIGENCE EDUCATIONAL MODEL USING ACTIVE LEARNING IN A VIRTUAL LEARNING ENVIRONMENT

Jan Skalka[1] **& Martin Drlik**[2]
[1,2] Constantine the Philosopher University in Nitra
[1] jskalka@ukf.sk, ORCID 0000-0003-2211-2794
[2] mdrlik@ukf.sk, ORCID 0000-0002-5958-7147

Abstract: *Artificial Intelligence (AI) is currently one of the fastest-growing areas, but pedagogical research on the design of curriculum and teaching methods for AI education is relatively rare. The article aims to identify techniques and content used in university AI teaching and its pros and cons. Subsequently, it identifies suitable forms of AI teaching for MOOCs, based on the experience and the possibilities of integration into the most modern educational environments. As a result, the article proposes an educational model based on a combination of active learning and a virtual learning environment that supports the distribution of educational content to the broadest possible groups of people interested in AI. At the same time, it describes the requirements and process of integrating the Jupyter notebook microenvironment into an independent web application, which is an essential prerequisite for connecting MOOC and computing server infrastructure.*

Keywords: artificial intelligence learning, virtual learning environment, active learning, e-learning, Jupyter notebook technology.

INTRODUCTION

Artificial Intelligence (AI) is currently one of the fastest-growing areas, which shows a growing demand for experts with advanced knowledge and the ability to learn and discover new approaches. According to (Acemoglu & Restrepo, 2020), AI, as a technology platform, can automate tasks previously performed by humans and generate new tasks and activities in which humans can be productively employed. However, recent technological changes are moving towards automation with insufficient focus on defining new jobs. This choice may result in stagnant demand for the labour market, a declining share of labour in national income, rising inequality and reduced productivity growth. Therefore, each country needs to respond and adapt to the changes that the implementation of AI brings in various areas of human activities (European Commission, 2021), (Zhang & Lu, 2021).

Moreover, this trend must be reflected first and foremost by the educational institutions that train future leaders and employees in this field. Due to the difficulty of individual areas of AI, this task falls to the greatest extent to universities. The advantage is that universities can fully benefit from the digital skills their students acquired at lower levels of study; the disadvantage is that only a tiny part of students choose to study AI as a part of IT. Furthermore, universities must deal with the consequences of AI mystification in the media and the complexity of the issues. Mastering practically every field of AI is complicated and requires prerequisites in the knowledge of mathematics, programming, statistics, etc. Therefore, universities must reconsider the form and structure of the educational content to increase the number of stakeholders interested in AI careers (Gao, Li, & Liu, 2021) (Chen, Chen, & Lin, 2020).

The article presents a proposal for an educational model based on a combination of active learning and a virtual learning environment that supports the distribution of educational content to the broadest possible groups of people interested in AI. At the same time, it presents the technological requirements for a tool supporting AI teaching and a case study within the FITPED-AI project (https://www.fitped.eu). The article consists of three main parts. The first one focuses on identifying best practices and content used in university teaching of AI and its pros and cons. The second part identifies suitable forms of AI teaching for MOOCs while building on the authors' experience and the possibilities of integration into the most modern educational environments. The last part presents a case study of the design and implementation of an educational environment supporting AI teaching.

1. ARTIFICIAL INTELLIGENCE EDUCATION

In this part, the article focuses on the presentation of best practices and content taught at universities in the field of AI.

In research connecting AI and education, the authors aim most often on AI contribution and application in education (Chassignol, Khoroshavin, Klimova, & Bilyatdinova, 2018), adoption of AI in the university environment (Rico-Bautista et al., 2021) or to specific scientific or educational areas in which AI tools are used (Xu & Babaian, 2021), (Lindqwister, Hassanpour, Lewis, & Sin, 2021).

A study (Xu & Babaian, 2021) shows widespread agreement that introductory AI courses are generally challenging to teach in engineering programmes despite growing enthusiasm for AI education. (Eaton et al., 2018) and (Langley, 2019) state that the primary reason is the required broad scope of students' entry knowledge and the complexity of AI caused by many advanced topics and techniques. A secondary reason is the constant updating of content due to the research and creation of new types of AI applications, which causes rapid obsolescence of knowledge, often in a short time. Integrating AI into the engineering study brings many advanced topics such as pattern identification, decision-making, and combining them into higher levels of reasoning abilities, sequential control, plan generation and integrated intelligent agents. Langley (Langley, 2019) defines the following requirements supporting integration:

- Present a system perspective that shows how mechanisms interact to produce intelligence (to combat views that AI is a collection of disconnected algorithms).
- Give students experience with encoding representational content that mechanisms interpret to produce behaviour (to clarify the centrality of structured representations in intelligent agents).
- Present topics in a cumulative manner, with later material layered on the earlier content, much as calculus builds on algebra, which draws on arithmetic (to emphasise the hierarchical character of intelligence).
- Teach students not only how to use AI methods but how to construct them from simpler components (to give them the ability to develop their own mechanisms when existing ones do not suffice).
- Cover important abilities exhibited in human intelligence even when challenging to formalise (to show the link between AI and psychology that address many of the same core phenomena).

The reason for these requirements is a lack of understanding of the basic principles of AI, the solution of isolated (partial, abstract) problems, as well as the fact that AI teaching is currently mainly oriented to the use of existing libraries without the necessity of an internal understanding of their principles. According to (Xu & Babaian, 2021), pedagogical research on the design of curriculum and teaching methods for AI training is relatively rare.

Considering the scope of the AI introduction curriculum, a typical representative of which is e.g. a book (Russell & Norvig, 2021), used in teaching in more than 1500 universities, integrating the above requirements is almost impossible. However, preparing educational content that can capture and retain the interest of students less skilled in abstract thinking is a constant challenge for authors. This statement is evidenced by a number of publications aimed at providing basic knowledge in the field of AI (Finlay, 2020), (Ertel, 2018), (Flasiński, 2016), (Jackson, 2019).

A popular output aimed at popularising artificial intelligence to the public is the course Elements of AI (https://www.elementsofai.com/) developed by the University of Helsinki and first launched in Finland in 2018. This course presents elements, problems, and selected solutions from the field of AI at the level of secondary school knowledge in an exciting way. According to (Heintz & Roos, 2021), the overall experience of setting up and running the course was very interesting and rewarding. Moreover, its impact was considerable, with many companies requesting the opportunity for their employees to participate in the course.

The requirements for effectively providing basic knowledge of AI to as wide a community as possible come from several basic views. From the point of view of user comfort, it is desirable to apply modern principles of digital content creation (Smyrnova-Trybulska, Noskova, Pavlova, Yakovleva, & Morze, 2016), (Latwal, Sharma, Mahajan, & Kommers, 2020). The focus of the task should guarantee the acquisition of skills and experience in accordance with the principles presented in (Capay, Skalka, & Drlik, 2017). Combined with the results of the analysis of the Elements of AI course platform, the general content requirements can be established as follows:

- active learning – the emphasis must be on dynamic content; the content creator must prefer explanations using examples and solving tasks,
- allow students to make mistakes and look for better solutions – prioritise content in the form of activities allowing them to make mistakes, optimise the solution, improve, and compete with each other,
- prioritise practicality at the expense of abstractness, even if the practical solution does not quite correspond to the theoretical basis – especially in the introductory chapters, where it is necessary to "build the user's relationship with AI",
- put less emphasis on the amount of content versus more focus on understanding it and building practical skills,
- divide the content into smaller units and "close them", thanks to which the student will have the feeling that he has already mastered some areas, even if they are only a prerequisite for understanding other topics,
- to support the mutual evaluation of students' solutions, the benefit of which is the understanding of different ways of thinking and approaches to solutions.

Even though we can find many courses focused on AI and specific areas of AI on educational portals (Table 1), they mostly do not meet the requirements mentioned above.

Table 1. Some types of AI-focused courses/educational materials on selected educational platforms (in August 2022). The content of AI and Data Science often overlaps in the courses, so this area was also included in the survey (Other popular portals, e.g. Khanacademy.org, and Udacity.com, contained a significantly smaller number of courses covering the given areas)

Educational portal	Artificial Intelligence	Data Science	Machine Learning	Deep Learning	Natural Language Processing
Coursera.org	731	1.506	676	269	79
Edx.org	242	345	208	224	12
Udemy.com	353	2.593	624	238	97
number of users in Udemy courses	*2.435 mil. learners*	*6.5 mil. learners*	*7.0 mil. learners*	*1.75 mil. learners*	*0.565 mil. learners*
Total	**1.316**	**4.444**	**1.138**	**731**	**188**

Source: Own work.

The reason is that the creation of such content is demanding and laborious, and there is a risk that during the preparation of the learning materials, the content will become outdated before they are completed. As a result, linear courses in the form of video lectures or video tutorials of varying quality are created. Moreover, despite the success and indisputable quality of the content, they often include the shortcomings mentioned in (Langley, 2019).

The data in Table 1 shows a strong interest in AI and selected areas that overlap or are part of it.

2. LEARNING FORMS SUITABLE FOR TEACHING AI

In this part, the article focuses on identifying suitable forms for teaching AI while looking for often used features in teaching programming, which has been the subject of intensive research in recent years.

The primary target group of university students focused on IT expects an effective acquisition of knowledge and practical skills, emphasising simplification. In other words, they wish to learn highly specialised knowledge and skills in AI following their habits to be ready for a career in AI (regardless of whether they finally choose it). Therefore, increasing the level of highly specialised knowledge and skills of students who consider or have already decided on a career in AI will be realised using a work-based learning strategy with elements of active-based, collaborate-based and problem-based learning.

Active learning, which transfers responsibility for progress in the educational process to the student, is one of the most effective and probably suitable forms of education for building knowledge and skills in AI. According to (Hartikainen, Rintala, Pylväs, & Nokelainen, 2019), active learning as an instructional approach includes different forms of activation, such as increased physical activity, interaction, social collaboration, deeper processing, elaboration, exploration of the material, etc. Active learning from this point of view is defined and viewed mostly through student activation. Other authors (Markant, Ruggeri, Gureckis, & Xu, 2016) proved that active learning leads to better outcomes than passive forms of instruction.

If it is considered that studying AI represents the same leap in thinking as learning to program, then it has to be also taken into account the results of flipped classroom experiments (D'Souza & Rodrigues, 2015), (Özyurt & Özyurt, 2018), (Peethambaran, Renumol, & Murthy, 2018). This method is one of the few that undoubtedly improves student results. However, its success is strongly conditioned by strict adherence to defined rules and measures that ensure students do their homework honestly (Skalka & Drlik, 2020).

Another provably functional and currently functioning approach is microlearning supporting and enabling study within short intervals (Carter & Youssef-Morgan, 2022). Moreover, if it is supplemented with appropriate gamification elements, it will demonstrably increase the satisfaction and motivation of students (Gasca-Hurtado & Gomez-Alvarez, 2021).

Based on (Skalka et al., 2021), it can be stated that the combination of micro-learning, gamification, immediate feedback, and the automatically evaluated programme assessment increased the quality of the training of experts in the field of programming. These elements represent the basis, integrated into the educational environment, enabling self-study with the automatic evaluation of results within the framework of microlearning, as well as automatic evaluation of programmes and provision of feedback. The virtual learning environment Priscilla (Skalka & Drlik, 2018a) can serve as an example and starting solution.

What does teaching AI require in addition to teaching programming? Suppose the teaching of AI follows the teaching of programming supported by an educational system with the features mentioned above. In that case, it is appropriate to integrate AI content into the same environment.

As a result, students meet a familiar environment and are not distracted by unknown functionalities and rules. They can thus fully concentrate on studying the content. However, from the view of the system creators, it is essential to identify the modules necessary to explain the initial problems of AI and thus ensure the closest possible connection between the perception of the real world and its transformation into tasks. For this purpose, the careful development of interactive modules enabling various kinds of experimentation (decision making, deductive reasoning, genetic algorithms, heuristic algorithms, etc.) is necessary.

The work of a data scientist is very often intertwined with the use of an environment using Jupyter notebooks where students write code that processes data and generates outputs prepared for interpretation. Sometimes the work ends at this point, and sometimes, the result is a model that will be deployed to solve the problems of the given class. If the student should follow these steps, he needs a tool that allows him to experiment with data, obtain the created model, verify its functionality, success, overfitting, speed, etc.

In the context of the existing educational system supporting the teaching of programming and the requirements mentioned above for AI education, the learning objects for AI courses can be defined as follows:

- microlearning – introduction to the issue, familiarisation with terms, presentation of superficial relationships and practice of simple tasks,
- automatic source code evaluation – will be available to prepare assignments, especially in the case of initial familiarisation with libraries,
- domain- and problem-specific independent modules enabling the solution of specific tasks defined as snapshots of reality; this part represents the most time-consuming activity of creating tasks focused primarily on motivating and building the educator's relationship with AI,
- Jupyter notebook ecosystem – represents an environment in which students, who master the essential topics, can experiment and transform real-world problems into it; mastering this environment is also a prerequisite for applying in the field of Data Science and/or AI, where Jupyter notebooks are widely used,
- collaboration, competition, and gamification – the training of AI professionals should be implemented through a learning strategy integrating the parts of active, collaborative and problem-based learning, using gamification and competition, which can make learning more interesting, more fun, more friendly, and more practical.

3. CASE STUDY: TECHNOLOGICAL BACKGROUND FOR AI LEARNING ENVIRONMENT

The last part presents the design of an educational model suitable for teaching AI and its support through a software solution.

As the output of the FITPED and FITPED-AI project consortium consisted of universities and SME organisations, the educational model and virtual learning environment focused on teaching programming languages were designed and implemented (Skalka & Drlík, 2018a).

The system combines microlearning and automatic evaluation of source codes, but it was designed to support the integration of other elements and activities as efficiently as possible. The system includes a web development environment that allows writing, running, and debugging programs without installing any supporting applications on the computer (Figure 1). Instead, the code is saved, executed and run on the server.

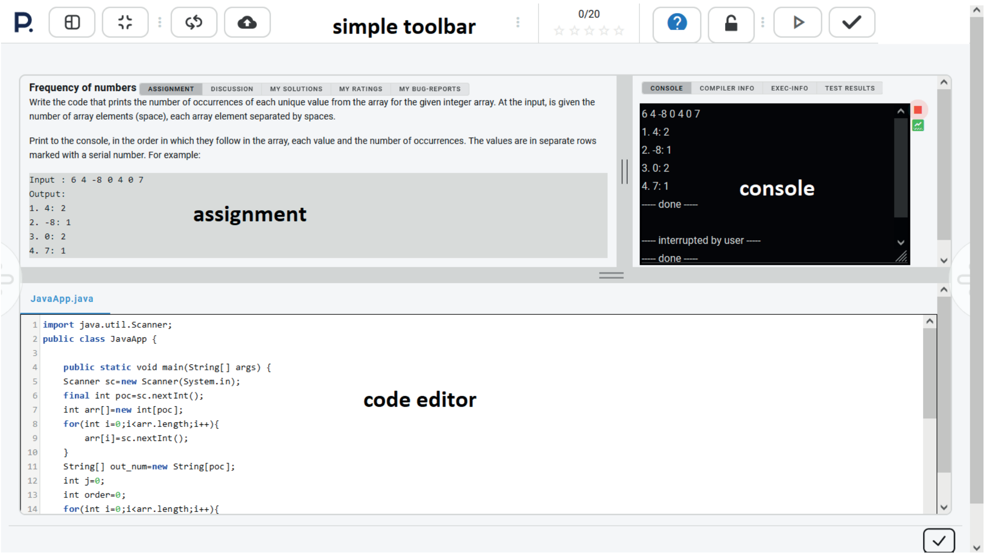


Figure 1. Web interface dedicated to solving and code writing in PRISCILLA system

Source: Own work.

To leave task solving to powerful server processors is a standard approach even in environments oriented to solving AI tasks. The primary reason is that to obtain a result, the programs, in many cases, need high performance and a long time, which cannot generally be provided on local devices.

The requirement for solving tasks and experimenting with data is currently most often implemented by Jupyter notebook technology (Pérez & Granger, 2015). Thanks to its openness, simplicity, and constant development, it has become a popular tool in teams focused on Data Science and AI. Currently, it is used not only as a format used in the processing of Data Science but also in education (Johnson, 2020). Its strength lies in combining text, source code and editing and running this code any time with a single click. Furthermore, the results can be displayed as part of the document content.

Jupyter server/notebook technology has a significant disadvantage, which was recently identified by the authors during its maturation – to use the computing and processing components, it is necessary to run the content from the given server – because notebooks could work via localhost by default (Project Jupyter, 2022). This approach made cooperation with other systems and front-end applications difficult or impossible.

The Jupyter Kernel Gateway (JKG) technology is currently used as one of the alternatives enabling the communication between an independent front-end and a Jupyter server running on the backend. According to (Project Jupyter Team, 2022), JKG is a web server that provides headless access to Jupyter kernels. As a result, the independent applications communicate with the kernels remotely through REST calls and WebSockets rather than ZeroMQ (Hintjens, 2013) messages.
Thanks to JKG, it was possible to implement modules that ensured communication with the Python language kernels, usually used to solve Data Science and AI tasks. A single kernel can be simultaneously connected to one or more front-ends.
In order to integrate the Jupyter infrastructure into the used Priscilla system and enable communication with Python kernels, it was necessary to create a clone of the design of a standard Jupyter notebook and enrich it with possible additional features (the ability to stop the program, friendly insertion of input data into the running program, the ability to combine with rich text, etc.) An example of the prepared content (from the FITPED-AI project) is presented in Figure 2.

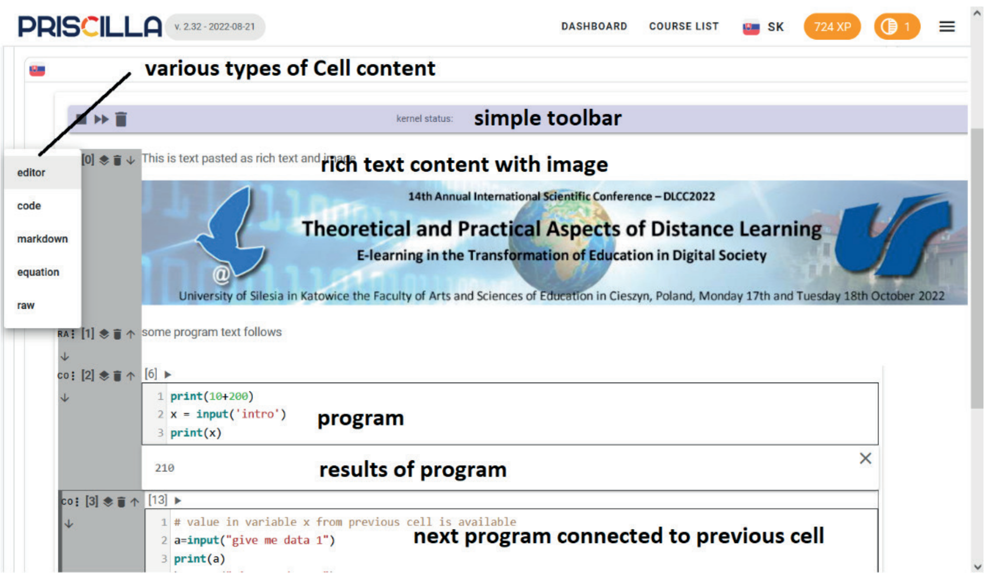
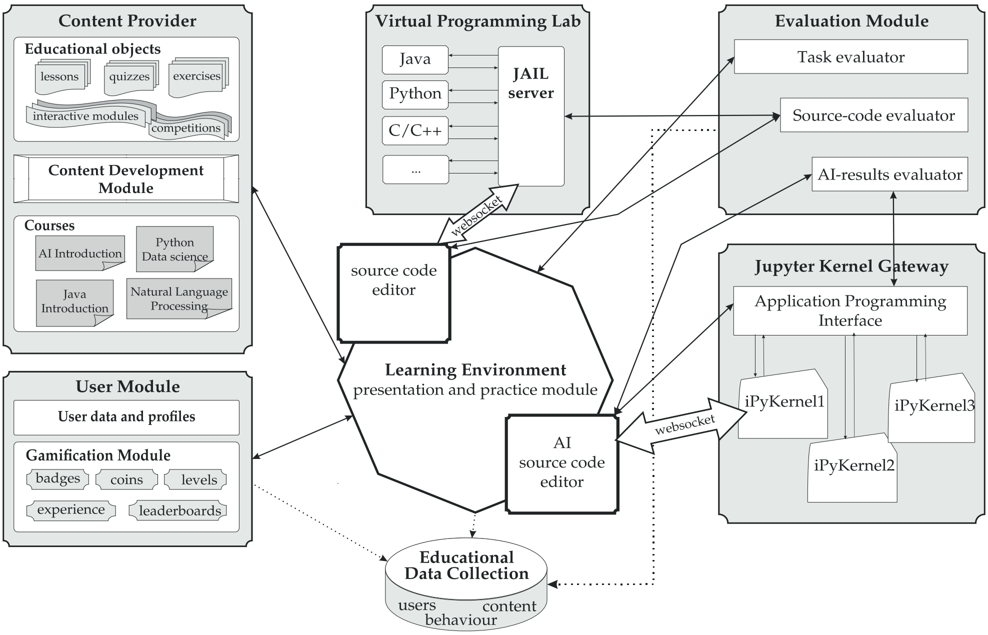


Figure 2. Integration of the Jupyter notebook design into the Priscilla system (a case study of the FITPED-AI project)

Source: Own work.

Integration of the Jupyter notebook clone environment with the backend technology of the Jupyter server and proven features of the Priscilla system provides the new system with all the original benefits (microlearning, gamification, automatic evaluation of source code, communication between users, etc.). A new logical and communication structure is presented in Figure 3.
The key part of the model is the **Learning environment**, which provides the content for the user/student and communicates with other modules with the aim of, e.g.

checking the answer correctness, logging the student's activity and attempts, providing assistance or help etc.
The Learning environment also includes separate modules dedicated to code writing or AI task solving in the Jupyter microenvironment (presented in Figures 1 and 2). These modules require communication with modules executing programs.



F i g u r e 3. Logical and communication structure of the technological solution ensuring coverage of the requirements of the educational model over the Priscilla system (a case study of the FITPED-AI project)

S o u r c e: Own work.

The first module is the web interface of the **source code editor**, with the backend covered by the Virtual Programming Lab server (Rodríguez-del-Pino, Rubio-Royo, & Hernández-Figueroa, 2012), which supports the execution of programs in various codes. Communication starts initially via REST API and then continues via WebSocket. The results of the program can be verified against the expected inputs, or the environment allows only program execution and communication through the console (entering inputs and reading outputs).
AI source code editor is defined by a structure consisting of cells that can contain various forms of text (images or equations) and source code currently in Python. The code in each cell can be executed independently, or the cells can contain pieces of code that follow each other. Each cell can be run separately and any number of times. When the code is started for the first time, a kernel is created on the server (via the REST API) in **Jupyter Kernel Gateway**. This kernel then communicates with the user via WebSocket. Listing of results and loading of inputs takes place in its front-end interface. The results are always listed under the cell whose code was run.

The evaluation module checks the correctness of the answers on three levels at the moment when the user decides to submit the task. Currently, three types of verification are available – validation of the solution from microlearning (compares against the database of correct answers), verification of the correctness of the program and verification of the results of the AI program (compares against the defined correct outputs for the prepared inputs).

All user attempts and responses are stored by the **Educational Data Collection module** tools, scored concerning gamification rules in the **Gamification module**, and logged as problematic in case of non-standard behaviour within the System module.

The **Content provider** is an essential part of the system. It ensures the creation of content based on individual types of educational objects and enables their organisation into lessons, chapters, courses, competitions, etc. In addition, questionnaires and discussions about the content are part of the module.

New modules of the system are currently in pilot operation, and content creation for courses in AI has been started.

CONCLUSION

The article aimed to propose an educational model supporting effective education in the field of AI.

In the first part, the forms and techniques used in teaching AI were presented, primarily within the subject defined as an introduction to AI. Most authors encountered the problems of the lack of research in AI education and the high demands on students' entry knowledge. Therefore, it is evident that the teaching of AI needs to be precisely planned within the study programme and define the study subject prerequisites that must be passed before starting the AI study.

In the second part, the forms of education that are accepted by nowadays students and, at the same time, can be used as part of AI teaching were presented. Tools providing immediate feedback, either based on the evaluation of the text response or the evaluation of the source code, were confirmed as suitable forms of content provision.

Finally, in the third part, the educational model was presented with the software technology enabling creating, solving and verifying the results of student solutions. This model copies the technologies used in AI in the labour market. At the same time, it enables the creation of educational content in a form accessible to a broad audience, thanks to the fact that it does not require any configuration or installation of software.

Newly developed modules needed to teach artificial intelligence courses will provide immediate feedback and support students' projects in artificial intelligence. The created educational content will consist of lessons for learning prerequisites of AI, classes for teaching basics of AI (Data Preparation, Knowledge Discovery, Artificial Intelligence, Machine Learning) and courses for teaching application domains of AI (Natural Language Processing, Educational Data Mining, Cybersecurity). In addition, educational data will be collected within several rounds of courses, which will be used for identifying students' behaviour and problem areas in the educational content and teaching process.

The steps leading to the creation of a mature graduate of a study covering the field of artificial intelligence with an IT orientation can be defined in two layers:

- Artificial intelligence demystification – on the one hand, artificial intelligence is not expected to solve all the world's problems. But on the other hand, many tabloid authors present it as the greatest danger for future generations. The content and activities should answer questions about what AI really is, its potential, and its risks for society.
- Knowledge and skills development to create solutions based on artificial intelligence mastery of AI technologies – the training courses should provide all the knowledge needed to understand the principles and design their solutions based on AI. They should also present specific solutions in knowledge discovery, cyber security, recommender systems, natural language processing and learning analytics.

ACKNOWLEDGEMENTS

The following grants funded this research: Science Grant Agency of the Ministry of Education of the Slovak Republic, grant number: 1/0490/22 "Design and Evaluation of Selected Learning Analytics Methods for Understanding and Improving the Learning Process and Virtual Learning Environment", European Commission under the ERASMUS+ Programme 2021, KA2, grant number: 2021-1-SK01-KA220-HED-000032095 "Future IT Professionals EDucation in Artificial Intelligence", the Ministry of Education of Slovakia, grant number 004UKF-2-1/2021 "Preparation and development of teaching courses in English with a focus on artificial intelligence in the form of blended-learning", and Ministry of Education of Slovakia, grant number: 2020/8148:34-A1101 "Support for the development of practical skills of UKF students in Nitra".

REFERENCES

Acemoglu, D. & Restrepo, P. (2020). The wrong kind of AI? Artificial intelligence and the future of labour demand. *Cambridge Journal of Regions, Economy and Society*, *13*(1). Retrieved from https://doi.org/10.1093/cjres/rsz022.

Capay, M., Skalka, J., & Drlik, M. (2017). Computer science learning activities based on experience. In *IEEE Global Engineering Education Conference, EDUCON*. Retrieved from https://doi.org/10.1109/EDUCON.2017.7943025.

Carter, J.W. & Youssef-Morgan, C. (2022). Psychological capital development effectiveness of face-to-face, online, and Micro-learning interventions. *Education and Information Technologies*. Retrieved from https://doi.org/10.1007/s10639-021-10824-5.

Chassignol, M., Khoroshavin, A., Klimova, A., & Bilyatdinova, A. (2018). Artificial Intelligence trends in education: A narrative overview. In *Procedia Computer Science* (Vol. 136). Retrieved from https://doi.org/10.1016/j.procs.2018.08.233.

Chen, L., Chen, P., & Lin, Z. (2020). Artificial Intelligence in Education: A Review. *IEEE Access*, *8*. Retrieved from https://doi.org/10.1109/ACCESS.2020.2988510.

D'Souza, M.J. & Rodrigues, P. (2015). Investigating the effectiveness of the flipped classroom in an introductory programming course. *New Educational Review*, *40*(2). Retrieved from https://doi.org/10.15804/tner.2015.40.2.11.

Eaton, E., Koenig, S., Schulz, C., Maurelli, F., Lee, J., Eckroth, J., … Williams, T. (2018). Blue sky ideas in artificial intelligence education from the EAAI 2017 new and future AI educator program. *AI Matters*, *3*(4). Retrieved from https://doi.org/10.1145/3175502.3175509.

Ertel, W. (2018). *Introduction to artificial intelligence*. Springer. ISBN 978-3-319-58486-7.

European Commission (2021). A European approach to Artificial intelligence. *A European Approach to Artificial Intelligence*.

Finlay, J. (2020). *An introduction to artificial intelligence*. CRC Press.

Flasiński, M. (2016). *Introduction to artificial intelligence*. Springer.

Gao, P., Li, J., & Liu, S. (2021). An Introduction to Key Technology in Artificial Intelligence and big Data Driven e-Learning and e-Education. *Mobile Networks and Applications*. Retrieved from https://doi.org/10.1007/s11036-021-01777-7.

Gasca-Hurtado, G.P. & Gomez-Alvarez, M.C. (2021). Mobile application based on gamification to promote microlearning in Software Engineering. In *Iberian Conference on Information Systems and Technologies, CISTI*. Retrieved from https://doi.org/10.23919/CISTI52073.2021.9476569.

Hartikainen, S., Rintala, H., Pylväs, L., & Nokelainen, P. (2019). The concept of active learning and the measurement of learning outcomes: A review of research in engineering higher education. *Education Sciences*, *9*(4). Retrieved from https://doi.org/10.3390/educsci9040276.

Heintz, F. & Roos, T. (2021). Elements of AI – Teaching the Basics of AI to Everyone in Sweden. In *EDULEARN21 Proceedings* (Vol. 1). Retrieved from https://doi.org/10.21125/edulearn.2021.0559.

Hintjens, P. (2013). *ZeroMQ: messaging for many applications*. O'Reilly Media, Inc.

Jackson, P.C. (2019). *Introduction to artificial intelligence*. Courier Dover Publications.

Johnson, J.W. (2020). Benefits and Pitfalls of Jupyter Notebooks in the Classroom. In *SIGITE 2020 – Proceedings of the 21st Annual Conference on Information Technology Education*. Retrieved from https://doi.org/10.1145/3368308.3415397.

Langley, P. (2019). An integrative framework for artificial intelligence education. In *33rd AAAI Conference on Artificial Intelligence, AAAI 2019, 31st Innovative Applications of Artificial Intelligence Conference, IAAI 2019 and the 9th AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019*. Retrieved from https://doi.org/10.1609/aaai.v33i01.33019670.

Latwal, G.S., Sharma, S.K., Mahajan, P., & Kommers, P. (2020). *Role of ICT in Higher Education: Trends, Problems, and Prospects*. CRC Press.

Lindqwister, A.L., Hassanpour, S., Lewis, P.J., & Sin, J.M. (2021). AI-RADS: An Artificial Intelligence Curriculum for Residents. *Academic Radiology*, *28*(12). Retrieved from https://doi.org/10.1016/j.acra.2020.09.017.

Markant, D.B., Ruggeri, A., Gureckis, T.M., & Xu, F. (2016). Enhanced Memory as a Common Effect of Active Learning. *Mind, Brain, and Education*. Retrieved from https://doi.org/10.1111/mbe.12117.

Özyurt, H. & Özyurt, Ö. (2018). Analysing the effects of adapted flipped classroom approach on computer programming success, attitude toward programming, and programming self-efficacy. *Computer Applications in Engineering Education*, *26*(6). Retrieved from https://doi.org/10.1002/cae.21973.

Peethambaran, M.K.P., Renumol, V.G., & Murthy, S. (2018). Flipped Classroom Strategy to Help Underachievers in Java Programming. In *Proceedings – 2018 6th International Conference on Learning and Teaching in Computing and Engineering, LaTiCE 2018*. Retrieved from https://doi.org/10.1109/LaTICE.2018.000-7.

Pérez, F. & Granger, B.E. (2015). Project Jupyter : Computational Narratives as the Engine of Collaborative Data Science. *UC Berkeley* (April).

Project Jupyter (2022). Standalone Jupyter server enhancement. Retrieved 30 August 2022, from https://jupyter.org/enhancement-proposals/28-jupyter-server/jupyter-server.html.

Project Jupyter Team (2022). Jupyter Kernel Gateway. Retrieved 30 August 2020, from https://jupyter-kernel-gateway.readthedocs.io/en/latest/.

Rico-Bautista, D., Medina-Cardenas, Y., Coronel-Rojas, L.A., Cuesta-Quintero, F., Maestre-Gongora, G., & Guerrero, C.D. (2021). Smart University: Key Factors for an Artificial Intelligence Adoption Model. In *Advances in Intelligent Systems and Computing* (Vol. 1307 AISC). Retrieved from https://doi.org/10.1007/978-981-33-4565-2_10.

Rodríguez-del-Pino, J.C., Rubio-Royo, E., & Hernández-Figueroa, Z. (2012). A Virtual Programming Lab for Moodle with automatic assessment and anti-plagiarism features. *Conference on E-Learning, e-Business, Entreprise Information Systems, & e-Government*.

Russell, S. & Norvig, P. (2021). Artificial Intelligence A Modern Approach (4th Edition). *Pearson Series*.

Skalka, J. & Drlik, M. (2018a). Priscilla – Proposal of System Architecture for Programming Learning and Teaching Environment. *IEEE International Conference on Application of Information and Communication Technologies*. Retrieved from https://publons.com/publon/27387754/.

Skalka, J. & Drlik, M. (2020). Automated assessment and microlearning units as predictors of at-risk students and students' outcomes in the introductory programming courses. *Applied Sciences (Switzerland)*, *10*(13). Retrieved from https://doi.org/10.3390/app10134566.

Skalka, J. & Drlík, M. (2018b). *Conceptual framework of microlearning-based training mobile application for improving programming skills. Advances in Intelligent Systems and Computing* (Vol. 725). Retrieved from https://doi.org/10.1007/978-3-319-75175-7_22.

Skalka, J., Drlik, M., Benko, L., Kapusta, J., Del Pino, J.C.R., Smyrnova-Trybulska, E., … Turcinek, P. (2021). Conceptual framework for programming skills development based on microlearning and automated source code evaluation in virtual learning environment. *Sustainability (Switzerland)*, *13*(6). Retrieved from https://doi.org/10.3390/su13063293.

Smyrnova-Trybulska, E., Noskova, T., Pavlova, T., Yakovleva, O., & Morze, N. (2016). New educational strategies in contemporary digital environment. *International Journal of Continuing Engineering Education and Life-Long Learning*, *26*(1). Retrieved from https://doi.org/10.1504/IJCEELL.2016.075036.

Xu, J.J. & Babaian, T. (2021). Artificial intelligence in business curriculum: The pedagogy and learning outcomes. *International Journal of Management Education*, *19*(3). Retrieved from https://doi.org/10.1016/j.ijme.2021.100550.

Zhang, C. & Lu, Y. (2021). Study on artificial intelligence: The state of the art and future prospects. *Journal of Industrial Information Integration*, *23*. Retrieved from https://doi.org/10.1016/j.jii.2021.100224.