**SYLLABUS**

**regarding the qualification cycle from 2023 to 2024**

1. Basic Course/Module Information

|  |  |
| --- | --- |
| Course/Module title | Artificial Intelligence |
| Course/Module code \* |  |
| Faculty (name of the unit offering the field of study) | College of Natural Sciences |
| Name of the unit running the course | Artificial Intelligence |
| Field of study | Computer Science & Computer Science and Econometrics |
| Qualification level | second degree |
| Profile | general academic profile |
| Study mode | full-time studies |
| Year and semester of studies | Year I, semester I |
| Course type | course subject |
| Language of instruction | English |
| Coordinator | Michał Kępski, PhD, Eng. |
| Course instructor | Michał Kępski, PhD, Eng. |

\* - as agreed at the faculty

1.1. Learning format – number of hours and ECTS credits

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Semester  (n0.) | Lectures | Classes | Colloquia | Lab classes | Seminars | Practical classes | Internships | others | **ECTS credits** |
|  |  |  |  | 30 |  |  |  |  | 4 |

1.2. Course delivery methods

- conducted in a traditional way

- involving distance education methods and techniques

1.3. Course/Module assessment (exam, pass with a grade, pass without a grade)

pass with a grade

2. Prerequisites

|  |
| --- |
| Mathematics (I.E. matrix algebra, derivatives and integrals of functions), programming in Phyton. |

3. Objectives, Learning Outcomes, Course Content, and Instructional Methods

3.1. Course/Module objectives

|  |  |
| --- | --- |
| O1 | *emphasise the main principles of computer-based image recognition* |
| O2 | *programming exemplary modules from image recognition system* |

3.2. Course/Module Learning Outcomes (to be completed by the coordinator)

|  |  |  |
| --- | --- | --- |
| Learning Outcome | The description of the learning outcome  defined for the course/module | Relation to the degree programme outcomes |
| LO\_01 | Familiarizing oneself with the fundamental principles and applications of various artificial intelligence methods and algorithms. | K\_W03, K\_W04 |
| LO\_02 | developing problem-solving skills with the use of selected methods of artificial intelligence. | K\_U04, K\_U05 |

**3.3. Course content**

1. Lectures

|  |
| --- |
| 1. Classification problem. k-Nearest Neighbor classifier. |
| 2. Linear classification, loss functions, regularization. |
| 3. Optimization methods for model parameters search (SGD, SGD-Momentum, etc). |
| 4. Decision tree models and its variations. Boosting. Random forest, XGBoost. |
| 5. Artificial neuron model. Neural networks and backpropagation. |
| 6. Convolutional Neural Networks: Convolution / Pooling Layers. |
| 7. Neural networks: Learning and evaluation. State-of-the-art architectures for various tasks. |
| 8. Data Preparation and Augmentation Techniques |
| 9. Understanding and Visualizing Convolutional Neural Networks. |
| 10. Transfer Learning and Fine-tuning Convolutional Neural Networks. |

1. Laboratories

|  |
| --- |
| Content outline |
| * + - 1. Intro to scientific computation with python and numpy. Jupyter notebook. |
| 2. Classification problem. k-Nearest Neighbor classifier. Train/val/test splits, cross validation. Performance measures. |
| 3. Linear classification: Support Vector Machine, Softmax. |
| 4. Optimization: Stochastic Gradient Descent. |
| 5. Artificial neuron model. |
| 6. Neural networks and backpropagation. |
| 7. Intro to pyTorch library. |
| 8. Neural networks architectures. Learning and evaluation. |
| 9. Convolutional Neural Networks: Architectures, Convolution / Pooling Layers |
| 10. Understanding and Visualizing Convolutional Neural Networks |
| 11.Transfer Learning and Fine-tuning Convolutional Neural Networks. |

3.4. Methods of Instruction

*Laboratory classes: writing code, conducting experiments.*

4. Assessment techniques and criteria

4.1 Methods of evaluating learning outcomes

|  |  |  |
| --- | --- | --- |
| Learning outcome | Methods of assessment of learning outcomes (e.g. test, oral exam, written exam, project, report, observation during classes) | Learning format (lectures, classes,…) |
| LO-01 | observation during classes, project | laboratories |
| LO-02 | observation during classes, project | laboratories |

4.2 Course assessment criteria

|  |
| --- |
| The assessment method of this course is as follows:  Every student must submit all notebooks with assignments, which will be graded on a scale of 0 to 65 points.  A project, which is evaluated up to 35 points. |

5. Total student workload needed to achieve the intended learning outcomes

– number of hours and ECTS credits

|  |  |
| --- | --- |
| Activity | Number of hours |
| Scheduled course contact hours | 30 |
| Other contact hours involving the teacher (consultation hours, examinations) | 15 |
| Non-contact hours - student's own work (preparation for classes or examinations, projects, etc.) | 65 |
| Total number of hours | 110 |
| Total number of ECTS credits | 4 |

\* One ECTS point corresponds to 25-30 hours of total student workload

6. Internships related to the course/module

|  |  |
| --- | --- |
| Number of hours |  |
| Internship regulations and procedures |  |

7. Instructional materials

|  |
| --- |
| Compulsory literature:   1. Ian J. Goodfellow, Yoshua Bengio, & Aaron Courville (2016). Deep Learning. MIT Press. 2. https://cs231n.github.io/ |
| Complementary literature:  Yoshua Bengio. (2012). Practical recommendations for gradient-based training of deep architectures; <https://arxiv.org/abs/1206.5533>  <https://numpy.org/doc/>  <https://pytorch.org/docs/> |

Approved by the Head of the Department or an authorised person