COURSE DESCRIPTION CARD - SYLLABUS

Course name
Deep neural networks

Course

Field of study
Computing

Year/semester
1/2

Area of study (specialization)
Intelligent Information Technologies

Profile of study
general academic

Level of study
Second-cycle studies

Course offered in
Polish

Form of study
full-time

Requirements
compulsory

Number of hours

Lecture
30

Laboratory classes
30

Other (e.g. online)

Tutorials

Projects/seminars

Number of credit points

5

Lecturers

Responsible for the course/lecturer:
prof. dr hab. inż. Krzysztof Krawiec
email: krzysztof.krawiec@cs.put.poznan.pl
tel. 61 665-3061
Institute of Computing Science, Faculty and Computing and Telecommunications
ul. Piotrowo 2, 60-965 Poznań

Responsible for the course/lecturer:
mgr inż. Jakub Bednarek
email: jakub.bednarek@put.poznan.pl
tel. 61 665-3063
Institute of Robotics and Machine Intelligence, Chair of Robotics
ul. Piotrowo 3A, 60-965 Poznań

Prerequisites

Learning objectives of the first cycle studies defined in the resolution of the PUT Academic Senate

Course objective

1. To make the students familiar with the fundamentals and selected topics of deep neural networks and related topics in machine learning.

2. To help the students in developing skills of solving problems pertaining to deep neural networks, in particular classification, regression, representation learning and feature engineering tasks.
3. To help students acquire experience needed to make use of the above capabilities when solving real-world problems.
4. To help the students in developing capabilities of leading and participating in small software projects involving deep neural networks, including teamwork skills and good practices.

Course-related learning outcomes

Knowledge

The student:

Has a structured and theoretically founded general knowledge related to key issues in the field of neural networks [K2st_W2].

Has advanced detailed knowledge regarding selected topics in deep neural networks, in particular machine learning, representation learning, feature engineering [K2st_W3].

Has advanced and detailed knowledge of the processes occurring in the life cycle of deep neural networks systems, including data acquisition techniques, and designing, testing and deployment of such systems [K2st_W5].

Knows advanced methods, techniques and tools used to solve complex engineering tasks and conduct research typical for solving classification, regression, and feature engineering tasks approached with deep neural networks techniques [K2st_W6].

Skills

The student:

Is able to plan and carry out experiments, including computer measurements and simulations, interpret the obtained results and draw conclusions and formulate and verify hypotheses related to complex engineering problems and simple research problems in deep neural networks [K2st_U3].

Can use analytical, simulation and experimental methods to formulate and solve engineering problems and simple research problems in deep neural networks [K2st_U4].

Can - when formulating and solving engineering tasks typical for neural networks - integrate knowledge from different areas of computer science (and if necessary also knowledge from other scientific disciplines) and apply a systemic approach, also taking into account non-technical aspects [K2st_U5].

Is able to assess the suitability and the possibility of using new achievements (methods and tools) and new IT products in neural networks and related areas of artificial intelligence [K2st_U6].

Can carry out a critical analysis of existing technical solutions used in deep neural networks systems and propose their improvements (streamlines) based on existing software libraries and environments such as TensorFlow and PyTorch [K2st_U8].

Is able - using among others conceptually new methods - to solve complex tasks involving design and implementation of deep neural networks systems, including atypical tasks and tasks containing a research component, e.g. advanced representation learning and/or feature engineering [K2st_U10].

Social competencies

The student:

Understands that in the field of deep neural networks the knowledge and skills quickly become obsolete [K2st_K1].

Understands the importance of using the latest knowledge in the field of deep neural networks, neural networks and related areas of machine learning and artificial intelligence in solving research and practical problems [K2st_K2].

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

Formative assessment:

a) lectures:

- asking student questions pertaining to the material presented in previous lectures,

b) laboratory classes:

- evaluation of progress in project realization (checkpointing)

Total assessment:

a) verification of assumed learning objectives related to lectures:

- Evaluation of acquired knowledge in the form of a written exam (5-8 open questions pertaining to lecture contents). Roughly half of questions are theoretical (define, describe, characterize, etc.), the other half are practical and require manual calculations (e.g., apply the erosion algorithm to a small binary image). Maximum total score: 25 points, of which 13 are required to obtain a positive grade.

b) verification of assumed learning objectives related to laboratory classes:

- Evaluation of progress along the semester classes, based on the project carried out by students and based on its documentation; students work on the project in part during the classes, and partially individually. The assigned grade reflects also student’s teamwork skills.
- Evaluation of student’s “defense” of project report and project presentation taking place at the last laboratory class, with the other students in the audience.

Additional assessment elements include:

- Student’s capability of applying the acquired knowledge to the problem posed in the project.
- Student’s remarks aimed at improving the quality of teaching material.
- Indications of students’ problems at acquisition and understanding of the knowledge presented at the lectures, aimed at improving the overall quality of the teaching process.

Programme content

Lecture:

The overall goal of the course it to familiarize students with theoretical and practical aspects of artificial neural networks, and in particular:

1. Deep neural networks and artificial neural networks as machine learning and optimization methods,

2. Algorithms for training of deep neural networks models,

3. Practical applications of neural networks for solving problems in classification, detection, regression, computer vision, and time series analysis.

To achieve the above objectives, the programme of this course has been structured as follows: Introduction. Definition of deep neural networks as a specific machine learning, optimization and modeling paradigm. Definition of parameters and hyperparameters of models. Discussion of the modular characteristics of deep models. Description of the most important and most used deep neural networks components, including dense, convolutional, aggregating, folding, reducing, and residual layers. Nonlinear and normalizing components. Connection with selected functional programming concepts. Taxonomy of loss functions and characteristics of the most frequently used loss functions. Learning through hetero- and auto-association. Implementation of deep neural networks algorithms. Deep architectures for the analysis of variable-size combinatorial structures, especially graphs. Deep models for unsupervised learning, in particular for cluster analysis. Generative models (GAN). Illustration of the performance of selected models on various metrics.

Lab:

The lab classes (15 x 2 hours) take place in computer laboratories. They start with a 6-hour preparatory part (three meetings in the beginning of the semester). The exercises and projects are carried out in two-people teams. Introduction (2h): Presentation of the outline of lab classes. Presentation of software tools used in lab classes (software libraries, software development environments). Preparatory sessions (6h): Hands-on training in software implementation of selected deep neural networks architecture and training algorithms in popular programming environments (Python, Keras, TensorFlow, Pytorch). Testing the implemented algorithms on real-world data and synthetic benchmarks. Assessment of algorithm correctness and efficiency (in particular time complexity/efficiency). Good practices in design and implementation of deep neural networks systems. Common pitfalls and how to avoid them. Software project (22h): Students form two-people teams and carry out software projects concerning specific deep neural networks tasks.

Teaching methods

1. Lectures: multimedia presentation, illustrated with examples, with occasional use of black board. Software demonstration.
2. Labs: practical exercises, problem solving, design and implementation of deep neural networks systems, performing computational experiments, discussion, teamwork, presentation of project outcomes (software and computational experiments).

Bibliography

Basic

1. Josh Patterson, Adam Gibson, deep neural networks : praktyczne wprowadzenie. Grupa Wydawnicza Helion. 2018.

2. Ian Goodfellow, Yoshua Bengio, Aaron Courville, deep neural networks: systemy uczące się. Wydawnictwo Naukowe PWN, 2018.

Additional

1. Valentino Zocca, Gianmario Spacagna, deep neural networks: uczenie głębokie z językiem Python: sztuczna inteligencja i sieci neuronowe, Grupa Wydawnicza Helion, 2018.

2. Krzysztof Krawiec Jerzy Stefanowski. Uczenie maszynowe i sieci neuronowe. Politechnika Poznańska. Wydawnictwo, 2004.

Breakdown of average student's workload

	Hours	ECTS
Total workload	125	5
Classes requiring direct contact with the teacher	60	2.5
Student's own work (literature studies, preparation for laboratory classes/tutorials, preparation for tests/exams, project preparation)[1]	65	2.5

[1] niepotrzebne skreślić lub dopisać inne czynności