Module name: Advanced Computational Intelligence

Academic year: 2019/2020  Code: EINF-2-209-MS-s  ECTS credits: 3

Faculty of: Faculty of Electrical Engineering, Automatics, Computer Science and Biomedical Engineering

Field of study: Computer Science  Specialty: Systems Modelling and Intelligent Data Analysis

Study level: Second-cycle studies  Form and type of study: Full-time studies

Lecture language: English  Profile of education: Academic (A)  Semester: 2

Course homepage: http://home.agh.edu.pl/~horzyk/lectures/ahdydaci.php

Responsible teacher: dr hab. Horzyk Adrian (horzyk@agh.edu.pl)

**Module summary**
The course expands the knowledge and practical experiences in design, implementation, and use of methods, algorithms, optimization, normalization, regularization and dropout techniques of Computational Intelligence for structures, methods, and efficiency improvements. We aim to develop complex well-optimized models using various types of neural networks (in Jupyter, Tensorflow, Keras etc) well-adapted to various kinds of training data of different groups: static, sequential, or structured.

**Description of learning outcomes for module**

<table>
<thead>
<tr>
<th>MLO code</th>
<th>Student after module completion</th>
<th>Connections with FLO</th>
<th>Method of learning outcomes verification (form of completion)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>has the knowledge/ knows how to/is able to</td>
<td>INF2A_K02, INF2A_K01</td>
<td>Presentation, Project, Execution of a project, Execution of laboratory classes, Involvement in teamwork</td>
</tr>
</tbody>
</table>

**Social competence: is able to**

- He can share knowledge, findings, discoveries and achievements. He can work in a team and communicate with other team members. He is able to think creatively and use the tools of creative thinking, e.g. brainstorming technique. He developed the ability to think entrepreneurial about implementation of the computational intelligence solutions in the industry.
M_U001

He can use libraries of computational intelligence methods to solve problems using various learning and adaptation techniques as well as is able to implement these methods, choose parameters and structures in order to optimize them for any given task.

INF2A_U08, INF2A_U02, INF2A_U01, INF2A_U03, INF2A_U05, INF2A_U07, INF2A_U06

Presentation, Project, Execution of a project, Execution of laboratory classes

Knowledge: he knows and understands

INF2A_U08, INF2A_U02, INF2A_U01, INF2A_U03, INF2A_U05, INF2A_U07

Presentation, Project, Execution of a project, Execution of laboratory classes

M_W001

He has knowledge in the area of designing, implementation and use of methods and techniques in the field of computational intelligence and can apply them to various problems and tasks.

INF2A_W04, INF2A_W03, INF2A_W02, INF2A_W01, INF2A_W05

Presentation, Project, Execution of a project, Execution of laboratory classes

M_W002

He knows how to select appropriate methods, architectures and parameters to optimize operations and the quality of generalization of applied methods of computational intelligence.

INF2A_W04, INF2A_W03, INF2A_W02, INF2A_W01, INF2A_W05, INF2A_W07

Execution of a project, Execution of laboratory classes, Presentation, Project

### Number of hours for each form of classes

<table>
<thead>
<tr>
<th>Suma</th>
<th>Lectures</th>
<th>Auditorium classes</th>
<th>Laboratory classes</th>
<th>Project classes</th>
<th>Conversation seminar</th>
<th>Seminar classes</th>
<th>Practical classes</th>
<th>Fieldwork classes</th>
<th>Workshops</th>
<th>Prace kontrolne i przejściowe</th>
<th>Lektorat</th>
</tr>
</thead>
<tbody>
<tr>
<td>42</td>
<td>14</td>
<td>0</td>
<td>14</td>
<td>14</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

### FLO matrix in relation to forms of classes

<table>
<thead>
<tr>
<th>MLO code</th>
<th>Student after module completion has the knowledge/ knows how to/is able to</th>
<th>Form of classes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Lectures</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Auditorium classes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Laboratory classes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Project classes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Conversation seminar</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Seminar classes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Practical classes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fieldwork classes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Workshops</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Prace kontrolne i przejściowe</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Lektorat</td>
</tr>
</tbody>
</table>

Social competence: is able to
M_K001: He can share knowledge, findings, discoveries and achievements. He can work in a team and communicate with other team members. He is able to think creatively and use the tools of creative thinking, e.g. brainstorming technique. He developed the ability to think entrepreneurial about implementation of the computational intelligence solutions in the industry.

Skills: he can

M_U001: He can use libraries of computational intelligence methods to solve problems using various learning and adaptation techniques as well as is able to implement these methods, choose parameters and structures in order to optimize them for any given task.

Knowledge: he knows and understands

M_W001: He has knowledge in the area of designing, implementation and use of methods and techniques in the field of computational intelligence and can apply them to various problems and tasks.

M_W002: He knows how to select appropriate methods, architectures and parameters to optimize operations and the quality of generalization of applied methods of computational intelligence.

---

**Student workload (ECTS credits balance)**

<table>
<thead>
<tr>
<th>Student activity form</th>
<th>Student workload</th>
</tr>
</thead>
<tbody>
<tr>
<td>Udział w zajęciach dydaktycznych/praktyka</td>
<td>42 h</td>
</tr>
<tr>
<td>Preparation for classes</td>
<td>6 h</td>
</tr>
<tr>
<td>przygotowanie projektu, prezentacji, pracy pisemnej, sprawozdania</td>
<td>34 h</td>
</tr>
<tr>
<td>Realization of independently performed tasks</td>
<td>6 h</td>
</tr>
<tr>
<td>Contact hours</td>
<td>2 h</td>
</tr>
<tr>
<td>Summary student workload</td>
<td>90 h</td>
</tr>
<tr>
<td>Module ECTS credits</td>
<td>3 ECTS</td>
</tr>
</tbody>
</table>

---

**Additional information**
Module content

Lectures

Deep Learning and Deep Neural Networks
Deepen the knowledge about modern deep learning networks as a way of high-level abstractions for optimizing neural structures and results will be presented. Various ways of creation of complex multiple processing layers for hierarchical feature extraction and concept creation will be explained. A few kinds of deep architectures and deep neural networks will be introduced and methods of their creation and training will be presented. Novel computational tools like TensorFlow, Keras, Jupyter Notebook will be presented and used for experiments.

Optimization, Regularization, Normalization, Vectorization and Speeding up Computations
Optimization of parameters and hyperparameters, Xavier initialization, various activation functions, ReLU, normalizing inputs, regularization, dropout, momentum, RMSprop, Adam optimization, learning rate decay and solving problem of local minima, vectorization, stochastic, mini-batch, and batch learning to speed up training, bias and variance, overfitting, dealing with vanishing and exploding gradients, and gradient checking. Introduction to criteria that allow identifying performance problems. Presentation of advanced techniques to hyperparameter tuning and optimization to raise the performance of the developed models.

Recurrent Neural Networks
Dynamic convergence to attraction points in recurrent neural networks together with adaptation methods of these kind of networks will be presented and supplemented with introduction of areas of their use. A few kinds of recurrent neural networks, their properties and abilities will be compared. The limitations of these kinds of networks due to their ability to remember training samples will be discussed.

Reinforcement, Motivated and Associative Learning Strategies
Reinforcement learning – interacts with the environment and maximizes a cumulative reward that controls a training process where data are sequential in time. It is an area of machine learning concerned with how agents ought to take actions in the environment so as to maximize a cumulative reward.
Motivated learning – defines fundamental needs and automatically develops secondary needs which affect the fundamental ones and control the interactions with the environment. During the learning process, fundamental needs should be satisfied what minimize pain (a penalty) and maximize pleasure (a reward) – they work as motivating factors.
Associative learning (cognitive learning) – aggregates the representation of similar features and objects, links them due to their real relations and actions of various kinds, connects them with different strengths and allows to trigger created associations recalling back related objects for a given context in time.

Fuzzy Logic Learning Systems
The main idea of fuzzy logic and fuzzy systems will be introduced. How to handle the concept of partial truth, linguistic variables and fuzzy attributes will be shown. Various kinds of fuzzy logic functions and fuzzy operators will be presented and compared. Fuzzy algebra will be introduced together with computational techniques that enable us to use fuzzy systems to solve various tasks on fuzzy data. Fuzzification and defuzzification processes and methods will be explained. Neuro-fuzzy systems will be also presented and the fuzzy systems will be used to adapt neural systems.

Associative Semantic Memories and Neural Networks
Since associative processes have a great impact on information processes in a human brain some of these processes will be modelled and presented on computational models. The way of working of various kinds of associative memories will be introduced and the substantial differences will be explained. An expanded model of association in neural structures will be introduced to model a kind of semantic and episodic memories. On this background, a few kinds of associative neural networks, their advanced associative features and concluding abilities will be presented. It will be shown how various data relations can be implemented and represented in these associative neural graph structures. This will allow us to substitute the timeconsuming search operations on classic data structures with more efficient operations on these associative neural structures.

Cognitive knowledge-based networks and intelligent linguistic chatbot systems
Cognitive knowledge-based networks and systems based on linguistic approaches, artificial needs, associative networks, and memories will be discussed. Intelligent linguistic knowledge-based chatbot systems will be described based on reinforcement and motivated learning approaches.

Support Vector Machines
Support Vector Machine idea for optimal discrimination and separation of classes will be explained, proven and analyzed. Nonlinear (polynomial, radial and sigmoidal) SVM and the way of their creation and training will be shown. Various types of SVM for classification and regression (approximation) will be introduced. Techniques of SVM adaptation to the larger number of classes will be presented. Limitations and computational problems of quadratic programming with linear constraints will be discussed.

Laboratory classes
We will use deep learning strategy to develop and adapt deep neural networks with the use of the advanced regularization, normalization, standardization, vectorization and other advanced optimization techniques to sample data in order to compare results to other previously obtained ones by other methods to conclude about their efficiency, adaptability and generalization properties. We will try to achieve better training and generalization results than using various data structure, optimization of hyperparameters of the developed models.

Deep convolutional, recurrent and associative neural network implementation, construction and learning
Use of Jupyter Notebook for construction of deep and recurrent neural network together with optimization, regularization, normalization, dropout, and other methods and techniques. We will develop complex architectures for various computational tasks working with different kinds of training data.

Project classes
Intelligent linguistic knowledge-based chatbot systems
We will develop intelligent linguistic knowledge-based chatbot systems based on associative networks and memories as well as on reinforcement and motivated learning approaches.

Associative structures, neural graph and memories implementation and adaptation
We will use associative graph data structures and associative systems to represent sample data in an associative form where horizontal and vertical data relations are
represented and can be easy used for concluding about them, e.g. their similarities, differences, correlations, classes and other attributes. We will automatically draw conclusions about these data, find classes and mine some interesting information. Parallel implementation using GPU will be an advantage.

Teaching methods and techniques:
Lectures: The content presented at the lecture is provided in the form of a multimedia presentation in combination with a classical lecture panel enriched with demonstrations relating to the issues presented.
Laboratory classes: During the laboratory classes, students independently solve the practical problem, choosing the right tools. The leader stimulates the group to reflect on the problem so that the obtained results have a high substantive value.
Project classes: Students carry out the project on their own without major intervention. This is to create a sense of responsibility for group work and responsibility for making decisions.

Warunki i sposób zaliczenia poszczególnych form zajęć, w tym zasady zaliczeń poprawkowych, a także warunki dopuszczenia do egzaminu:
On the basis of the quality, scope, difficulties of the implemented methods, models, structures, and functionalities in the project and the achieved results.

Zasady udziału w poszczególnych zajęciach, ze wskazaniem, czy obecność studenta na zajęciach jest obowiązkowa:
Lectures:
- Attendance is mandatory: No
- Participation rules in classes: Students participate in the classes learning the next teaching content according to the syllabus of the subject. Students should constantly ask questions and explain doubts.
An audiovisual recording of the lecture requires the teacher’s consent.
Laboratory classes:
- Attendance is mandatory: Yes
- Participation rules in classes: Students carry out laboratory exercises in accordance with materials provided by the teacher. The student is obliged to prepare for the subject of the exercise, which can be verified in an oral or written test. Completion of classes takes place on the basis of presenting a solution to the problem. Completion of the module is possible after completing all laboratory classes.
Project classes:
- Attendance is mandatory: Yes
- Participation rules in classes: Students carry out practical work aimed at obtaining competences assumed by the syllabus. The project implementation method and the final result are subject to evaluation.

Method of calculating the final grade
Individually, on the basis of the quality and scale of implemented methods, models, structures, and functionalities of the chosen and implemented project.

Sposób i tryb wyrównywania zaległości powstałych wskutek nieobecności studenta na zajęciach:
Students are obliged to make up the arrears resulting from the absence according to individual arrangements with the teacher depending on the type and subject of the vacated activities.

Prerequisites and additional requirements
Each student who will take a part in this course should have a good background in computational techniques, object programming languages. Every student should be able to use these languages to create computer applications and use complex data structures, e.g. inhomogeneous graphs or trees of classes representing neurons. It is also necessary to have sufficient skills and knowledge of the English language to take part in lectures and classes.
Recommended literature and teaching resources
1. Cruse, Holk; Neural Networks as Cybernetic Systems, 2nd and revised edition, file:///C:/Users/Adrian/Downloads/bmm615.pdf

Scientific publications of module course instructors related to the topic of the module
11. A. Horzyk and J.A. Starzyk, Multi-Class and Multi-Label Classification Using Associative Pulsing

**Additional information**

The course language is English, so the students should know this language enough to understand lectures, learn from English written materials and use it to communicate with a teacher and other students to solve tasks and take an active part in laboratory and project classes.