Master in Artificial Intelligence (2021)

91254 - Image Processing and Computer Vision

PDF, ADOC.

Learning outcomes

At the end of the course, the student masters the basic algorithms, tools and systems for the management, processing and analysis of digital images. The student is able to design and develop simple systems oriented to real-world computer vision applications such as those requiring segmentation and classification of objects in digital images.

Course contents

Introduction – Basic definitions related to

image processing and computer vision. An overview across major

application domains.

Image Formation and Acquisition – Geometry of

image formation. Pinhole camera and perspective projection.

Geometry of stereopsis. Using lenses. Field of view and depth of

field. Projective coordinates and perspective projection matrix.

Camera calibration: intrinsic and extrinsic parameters, lens

distortion. Camera calibration based on planar targets and

homography estimation (Zhang’s algorithm). Image rectification and

stereo calibration. Basic notions on image sensing, sampling and

quantization.

Intensity Transformations - Image

Histogram, Linear and non-linear contrast stretching. Histogram

equalization.

Spatial Filtering – Linear Shift-invariant

operators. Convolution and correlation. Mean and Gaussian filtering. Median

Filtering. Bilateral filtering. Non-local means.

Image Segmentation – Binarization by global

thresholding. Automatic threshold estimation. Spatially adaptive

binarization. Colour-based segmentation.

Binary Morphology – Dilation and erosion.

Opening and closing- Hit-and-Miss.

Blob Analysis – Distances on the image plane

and connectivity. Labeling of connected components. Basic

descriptors: area, perimeter, compactness, circularity. Orientation

and bounding-box, form factor and related descriptors. Euler

number. Image moments, invariant moments.

Edge Detection - Image gradient. Smooth

derivatives: Prewitt, Sobel, Frei-Chen. Non-maxima suppression.

Laplacian of Gaussion. Canny edge detector.

Local Invariant Features – Detectors and

descriptors. Harris Corners. Scale invariant features. SIFT features. Efficient feature matching by kd-trees.

Object Detection – Pattern matching by SSD,

SAD, NCC and ZNCC. Fast pattern matching. Shape-based mathing.

Hough Transform for analystic shapes, Generalized Hough

Transform. Object detection by local invariant features:

Hough-based voting, least-squares similarity

estimation.

Theory is complemented by assisted hands-on lab

sessions based on Python and the OpenCV library. Lab sessions cover selected

topics such as intensity transformations, spatial filtering, camera calibration, motion estimation and local invariant features.

Readings/Bibliography

91247 - Cognition and Neuroscience

PDF, ADOC.

Learning outcomes

At the end of the course, the student knows state-of-art human and animal research that uses neuroscience techniques to understand the cognitive and emotional aspects of the human mind and behavior. The student is able to critically read experimental and theoretical studies of cognitive and affective neuroscience, to evaluate their methods and results, explain their significance, and apply such notions in the study and development of artificial intelligence systems.

Course contents

How do neurons in the brain give rise to mind – to our abilities to sense and perceive the world, to act and think about it, to learn and remember it ? This course will provide an accessible but highly challenging survey of the empirical evidence, theories and methods in cognitive neuroscience exploring how cognition is instantiated in neural activity. Drawing on a wide variety of investigative tools available to cognitive neuroscience, the course explores the neural mechanisms underlying complex cognitive processes, including visual perception, selective attention, learning and emotion, cognitive control and decision-making. Importantly, the course examines the logic and assumptions that permit us to interpret brain activity in psychological terms.

At the end of the course the student is be able to:

get in-depth understanding of the neural substrates and functional mechanisms of mental processes;

get knowledge of the-state-of-the-art methodologies and novel approaches of current research in Cognitive Neuroscience;

critically review and discuss the theoretical and empirical contributions of the current literature, understand and analyse the methods employed, interpret their results and critically assess their conclusions;

exercise the ability to engage in creative thinking leading to formulations of new hypotheses and planning of their empirical testing.

The course Cognition & Neuroscience is divided into two teaching modules.

Module 1 (4 CFU):

Associative learning

Reinforcement learning

Decision-making

Module 2 (2 CFU):

Vision

Visual object recognition

Visual selective attention

Previous knowledge required:

Prerequisite involves high-school knowledge of the biology of the Central Nervous System

In addition, it is highly recommended to attend to online video lectures on Neuroscience Core concepts, freely available at the Society for Neuroscience website:

https://www.brainfacts.org/core-concepts

Rea

91762 - Combinatorial Decision Making and Optimization

PDF, ADOC.

Learning outcomes

At the end of the course, the student has an understanding of the most popular methods in operations research and artificial intelligence for modeling and solving complex combinatorial optimization problems such as constraint programming, integer linear programming, SAT and SMT.

Course contents

Prerequisites

Problem solving in artificial intelligence (covered by Module 1 of 91248 - Fundamentals of Artificial Intelligence and Knowledge Representation), logic for computer science (covered by Module 1 of 91251 - Languages and Algorithms for Artificial Intelligence), basic notions of algorithms and programming preferably in Python (covered by 91256 - Introduction to Algorithms and Programming), basic algebra and analysis.

Course topics

The first half of the course (Module 1) will introduce the fundamental concepts in Constraint Programming (CP) and SAT, which are two general-purpose approaches in Artificial Intelligence (AI) to combinatorial decision making and optimization. The topics include:

Introduction to CP

Modelling in CP

Constraint propagation and global constraints

Constructive tree search and propagation, branching heuristics, randomization and search restarts, Large Neighbourhood Search (LNS)

The SAT problem and encoding in SAT

SAT encodings

Basic solving techniques (resolution, unit propagation, DPLL)

Conflict-driven clause learning algorithms

The second half of the course (Module 2) will introduce the fundamentals of Satisfiability Modulo Theory, which is a generalization of SAT. The last part of the module will be about the essential Operations Research (OR) results from linear programming and in particular (mixed) integer programming, by prioritizing the width of the covered area over the depth of the introduction. The topics include:

Introduction to Satisfiability Modulo Theories (SMT)

Eager vs lazy SMT encoding, the DPLL(T) algorithm

SMT theory solvers, combinations of theories and extensions

Introduction to Operations Research (OR) and Linear Programming (LP)

Simplex algorithm and duality

Integer Linear Programming (ILP)

Extensions and hybrid methods

The course will also host an invited lecture by a researcher in the field in order to provide a broader perspective on the AI-based approaches to combinatorial decision making and optimization.

91248 - Fundamentals of Artificial Intelligence and Knowledge Representation

PDF, ADOC.

Learning outcomes

At the end of the course, the student knows the main knowledge representation techniques and reasoning methods that underlie artificial intelligence problem solving. The student is able to develop simple solvers for artificial intelligence systems.

Course contents

The course introduces the fundamental principles and methods used in Artificial Intelligence to solve problems, with a special focus on the search in the state space, planning, knowledge representation and reasoning, and on the methods for dealing with uncertain knowledge. The course will include hands-on labs and seminars on selected topics.

Prerequisites: user-level knowledge of a high-level programming language, in order to successfully understand case studies and applications presented during the lessons.

Part I (Module 1)

Module 1

Introduction to Artificial Intelligence: historical perspective, main application fields, introduction to knowledge-based systems and architectural organization.

Problem-solving in AI: representation through the notion of state, forward e backward reasoning, solving as a search and search strategies. Games. Constraint satisfaction problems.

Local Search methods, meta heuristics, solving through decomposition, constraint relaxation, branch-and-bound techniques

Introduction to Planning, Linear planning, partial order planning, graph-based methods (GraphPlan), Scheduling.

Part II (Modules 2, 3, and 4):

Module 2

Introduction to knowledge representation and reasoning

Representing Terminological Knowledge: semantic networks, description logics, foundation of ontologies

Representing actions, situations, and events.

Reasoning with Beliefs.

Nonmonotonic Reasoning and reasoning with default Information, Truth Maintenance Systems.

Module 3

Uncertainty

Probabilistic Reasoning

Module 4

Rule-based systems: Prolog and extensions, meta-interpreters, DCG, planning in prolog, Prolog for temporal reasoning with the Event Calculus, LPAD.

Forward chaining and RETE, Drools

Limits of KR based on Logic: Abductive, Inductive, and case-based reasoning. Fuzzy reasoning.

91251 - Languages and Algorithms for Artificial Intelligence

PDF, ADOC.

Learning outcomes

At the end of the course, the student has an understanding of the main linguistic and algorithmic techniques used in the context of AI, including the main aspects related to functional, logic and constraint programming, with their theoretical foundations and practical applications. The student has an understanding of the basic principles of distributed models and algorithms, consensus problems, and large-scale algorithms for the analysis of big data.

Course contents

Module 1

Introduction to mathematical logic. Unification. Resolution.

Introduction to logic programming. Prolog languages.

Constraint logic programming and concurrent constraint programming. Constraint programming. The language MiniZinc.

Module 2

Programming and learning paradigm, and the languages for inference. Introduction to the language Scheme/Racket and its probabilistic extensions Church/Gamble. Generative models as probabilistic programs. Bayesian programming in Church/Gamble. Learning as conditional inference. Inference algorithms (rejection sampling and Metropolis Hastings). Operational semantics and inference by evaluation. Time permitting, an overview of differentiable programming.

Module 3

Rudiments of Computability and Complexity Theory. Undecidability, polynomial time computability, and NP-completeness. Between the feasible and the unfeasible. A Formal Learning Model: PAC Learning. Learning via Uniform Convergence. The Bias-Complexity Tradeoff.

The VC-Dimension.

Re

91256 - Introduction to Algorithms and Programming

PDF, ADOC.

Learning outcomes

At the end of the course, the student is able to analyze a problem, formulate an algorithm for solving it, and code such an algorithm into an executable program. The student understands some basic algorithms and a general-purpose programming language.

Course contents

Computer science and problem solving. Abstract machines. Python machine. Programming in Python: names and visibility, functions, immutable and mutable objects, basic data types (numbers, strings, tuples, lists, dictionaries) and their use in the solution of problems. Classes and objects. Exceptions. Basics of computational complexity. Methods and inheritance. Introduction to the library NumPy and its N-dimensional array objects.

Some lab work will propose experiments and homework.

Prerequisites: some programming experience. In its 40 hours, the course shall only introduce one programming language, and cannot treat programming techniques, which are largely independent from the specific language.

91255 - Statistical and Mathematical Methods for Artificial Intelligence

PDF, ADOC.

Learning outcomes

At the end of the course, the student masters the basic mathematical and statistical methods needed to acquire skills in artificial intelligence foundations, theories and applications.

Course contents

Unless new restrictions, the course will be held in blended modality. See the timetable for more details.

1.Elements of linear algebra.

vectorial calculus, linear mappings, normed spaces, orthogonal projections.

matrix calculus, matrix norms, special matrices.

Singular Values Decomposition, Principal Component Analysis.

Eigenvalues and eigenvectors.

Laboratory exercises in Matlab or Phyton.

1. Elements of multivariate analysis

Gradient, Jacobian, Hessian. Taylor theorem.

Convex functions and sets.

1. Multivariate optimization

Linear least squares.

Extrema of multivariate functions. Optimality conditions.

Descent methods. Gradient type methods and Newton type methods.

Regularization.

Basis concepts of stochastic optimization.

Laboratory exercises in Matlab or Phyton.

1. Elements of probability and statistics.

Probability and Bayes theorem.

Random variables. Continuous and discrete distributions of random variables. Normal and Poisson distributions.

Independent and dependent variables. Covariance and correlation.

Estimates: Maximum Likelihood and Maximum a Posteriori estimates.

Cross entropy and Kullback-Leibler divergence.

Laboratory exercises in Matlab or Phyton.

Reading

91258 - Natural Language Processing

PDF, ADOC.

Learning outcomes

At the end of the course, the student masters the basic algorithms, tools and systems for the management, processing and analysis of natural language. The student is able to design and develop simple systems oriented to real-world natural language processing applications such as those requiring segmentation and classification of elements of text.

Course contents

Topics

Introduction to NLP

Elements of computational linguistics

Basic text processing

Language modeling with n-grams

Part-of-speech tagging

Constituency and dependency parsing

Text classification using linear models

Vector semantics and embeddings

Deep learning methods for NLP

Typical NLP applications, e.g., sentiment analysis, question answering, machine translation, and dialog systems

Advanced topics and applications, e.g., argument mining, legal analytics, neural-symbolic architectures

Prerequisites

The course is intended for 2nd-year students of the Masters' Degree in Artificial Intelligence, who have a solid background in foundational subjects and technological tools for tackling modern Natural Language Processing. There are no formal prerequisites in terms of exams taken, but students are expected to be familiar with machine learning and deep learning concepts, methodologies and tools, AI & KR frameworks and methods, statistical and mathematical methods for AI, and the Python programming language.

91257 - Ethics in Artificial Intelligence

PDF, ADOC.

Learning outcomes

At the end of the course, the student understands the main ethical aspects related to the creation and use of artificial intelligence systems. The student is able to reason critically about the main machine ethics issues connected with the moral behavior of AI systems endowed with autonomous decision capabilities and is able to evaluate the impact of the ethical and legal aspects connected with AI systems.

Course contents

The course aims to provide a thorough overview of the many ethical and social issues raised by information and communication technologies (ICTs), with particular attention to Artificial Intelligence and its impact on society and individuals.

Students will learn about the most compelling social and ethical challenges posed by ICTs and how to approach them in a critical way. Conceptual analysis will be supported by discussion of practical case studies.

Erasmus students follow the same programme of regular students.

Course Contents:

Introduction to Ethics

Information technologies, purposeful behavior and intelligence;

Singularity and Superintelligence;

Artificial Agency, Free Will, Consciousness;

Artificial Agents and Responsibility;

Machine Ethics;

AI Ethics and Roboethics;

Machine Learning, Big Data and the issues of Bias and Discrimination;

Anthropomorphism, Human-Computer/Robot Interaction (HCI, HRI), and Human Dignity; AI and Trust;

Human in the loop, Security, and Accountability;

AI Explainability and Transparency;

Assessing AI use cases. Socio-Technical Scenarios

Assessing AI use cases. Ethical tensions, Trade offs.

Possible use cases:

Military robots and Autonomous Weapon Systems;

Self-driving cars;

Expert systems: COMPAS, Watson, …;

Aviation and Air Traffic Management;

Machine artistic creativity (TheNextRembrandt, Obvious Art, Shimon, …);

AI Ethics in Healthcare

Legal relevance of AI Ethics

Ethical Business

AI and privacy

Modelling of legal and ethical norms and reasoning (with exercises)

Rea

91259 - Architecture and Platforms for Artificial Intelligence

PDF, ADOC.

Learning outcomes

At the end of the course, the student has a deep understanding of the requirements of machine-learning workloads for computing systems, has an understanding of the main architectures for accelerating machine learning workloads and heterogeneous architectures for embedded machine learning, and of the most popular platforms made available by cloud providers to specifically support machine/deep learning applications.

Course contents

Module 1 (for students of 93398 and 91259, by Prof. L. Benini)

From ML to DNNs - a computational perspective

Introduction to key computational kernels (dot-product, matrix multiply…)

Inference vs training - workload analysis characterization

The NN computational zoo: DNNs, CNNs, RNNs, GNNs, Attention-based Networks

Running ML workloads on programmable processors

recap of processor instruction set architecture (ISA) with focus on data processing

improving processor ISAs for ML: RISC-V and ARM use cases

fundamentals of parallel processor architecture and parallelization of ML workloads

Algorithmic optimizations for ML

Key bottlenecks taxonomy of optimization techniques

Algorithmic techniques: Strassen, Winograd, FFT

Topology optimization: efficient NN models - depthwise convolutions, inverse bottleneck, introduction to Neural Architectural Search

Module 2 (for students of 93398, by Prof. F. Conti)

Representing data in Deep Neural Networks

Recap of canonical DNN loops – a tensor-centric view

Data quantization in Deep Neural Networks

Brief notes on data pruning

From training to software-based deployment

High-performance embedded systems (NVIDIA Xavier, Huawei Ascend)

Microcontroller-based systems (STM32)

From software to hardware acceleration

Principles of DNN acceleration: spatial and temporal data reuse; dataflow loop nests and taxonomy; data tiling

The Neural Engine zoo: convolvers, matrix product accelerators, systolic arrays – examples from the state-of-the-art

Module 2 (for students of 91259, by Prof. G. Zavattaro)

Introduction to parallel programming.

Parallel programming patterns: embarassingly parallel, decomposition, master/worker, scan, reduce, …

Shared-Memory programming with OpenMP.

OpenMP programming model: the “omp parallel” costruct, scoping costructs, other work-sharing costructs.

Some examples of applications.

Readings

91261 - Artificial Intelligence in Industry

PDF, ADOC.

Learning outcomes

At the end of the course, the student has a deep knowledge of industrial applications that benefit from the use of machine learning, optimization and simulation. The student has a domain-specific knowledge of practical use cases discussed in collaboration with industrial experts in a variety of domains such as manufacturing, automotive and multi-media.

Course contents

The course is primarily delivered as a series of simplified industrial use cases. The goal is provide examples of challenges that typically arise when solving industrial problems.

Use cases may cover topics such as:

Anomaly detection

Remaining Useful Life (RUL) estimation

RUL based maintenance policies

Resource management planning

Recommendation systems with fairness constraints

Power network management problems

Epidemic control

Production planning

The course will emphasize the ability to view problems in their entirety and adapt to their peculiarities. This will frequently require to combine heterogeneous solution techniques, using integration schemes both simple and advanced.

The employed method will include:

Mathematical modeling of industrial problems

Predictive and diagnostic models for time series

Combinatorial Optimization

Integration methods for Probabilistic Models and Machine Learning

Integration methods for constraints and Machine Learning

Integration methods for combinatorial optimization and Machine Learning

The course will include seminars on real-world use cases, from industry experts.

The course contents may (and typically will) be subject to changes, so as to adapt to some degree to the interests and characteristics of the attending students.

91262 - Data Mining, Text Mining and Big Data Analytics

PDF, ADOC.

Learning outcomes

At the end of the course, the student understands how a possibly very large set of data can be analyzed to derive strategic information and to address "data-driven" decisions. The student has a knowledge of the main data-mining tasks such as data selection, data transformation, analysis and interpretation, with specific reference to unstructured text data, and with the issues related to analysis in "big data" environments.

Course contents

Presentation slides of the course

Module 1 - Data Mining (Claudio Sartori)

See 75194 - DATA MINING M [https://www.unibo.it/en/teaching/course-unit-catalogue/course-unit/2021/391683] Module 2 only

Module 2 - Text Mining (Gianluca Moro)

The text mining module focuses on knowledge discovering from large corpora of unstructured text which is fundamental to deal with several natural language processing tasks, such as text representation models, indexing and classification, analysis of topics, semantic similarity search, explaining behaviours and phenomenon of interest (a.k.a. descriptive text mining), sentiment analysis and opinion mining, text summarisation, chatbot or digital assistant creation, cross-modal information retrieval of texts and images etc.

The learning outcomes of the module are the capabilities of defining and implementing text mining processes, from text pre-processing and representation with traditional approaches and then with novel neural language models, up to the knowledge discovery with data science methods and machine & deep learning algorithms from several sources, such tweets, facebook posts, reviews, web pages, emails, loan requests, legal cases, news and documents in general.

The module introduces non-contextual language models based on word embeddings, such as GloVe and word2vec, and memory based neural networks particularly effective for textual data, such as recurrent neural networks like LSTM, GRU and BiLSTM, up to the attention mechanism, the transformer and the state-of-the-art of contextual word embeddings based on BERT and new linear transformer models for text summarization and deep metric learning for cross-modal information retrieval of texts and images.

Last but not least, the unit illustrates the transfer learning paradigm to exploit and fine tune existing models in target domains which are semantically different from their training source domains; this is particularly useful in order to overcome the lack of labeled data in the target domain.

Module 3 - Big Data Analytics (Stefano Lodi)

Maps and reductions in parallel programming. The MapReduce programming model.

The Hadoop implementation of MapReduce.

The Python API to the Spark system and examples of parallel programs.

The Machine Learning

Library (MLlib) of Spark.

Read

91265 - Knowledge Engineering

PDF, ADOC.

Learning outcomes

At the end of the course, the student knows some (semi-)automated methods for joint interpretation of data and content as sources of knowledge. The student masters the basics of knowledge extraction, engineering, and linking, making data suitable to machine querying and automated reasoning, typically on decentralized platforms such as the Web.

Course contents

Knowledge graphs and ontologies

OWL, RDF, SPARQL

Ontology Design methodologies, focus on Extreme Design

Intensional and Extensional modelling

Ontology design patterns

Applied reasoning

Knowledge extraction from text

Ontology evaluation and quality

91266 - Machine Learning for Computer Vision

PDF, ADOC.

Learning outcomes

At the end of the course, the student masters the most popular modern machine-learning approaches to computer-vision tasks, with particular reference to specialized deep-learning architectures. The student has both a theoretical understanding and the necessary practical skills required to develop state-of-the-art image and video analysis systems for real-world applications.

Course contents

The course is held in the first semester, from September to December.

Topics:

Image classification/recognition: limits of hand-crafted methods; brief review of machine learning and data-driven methods; bag-of-words models applied to dense descriptors; introduction to Neural Networks (NNs) and Convolutional NNs; AlexNet and the deep learning revolution. Advanced and state-of-the-art architectures for image classification. Hands-on sessions on image classification: training from scratch and transfer learning.

Intro to PyTorch and practical aspects of model training. Discussion of notebooks provided by the instructor to get to know PyTorch and learn how to train CNNs for image classification; discussion on the impact of several design and optimization choices.

Object detection. Introduction to ensemble learning via boosting. The Viola-Jones detector and its applications. Specialized NN architectures for object detection. Two-stages, one-stage, and anchor-free detectors. RoI Pooling operator, Feature Pyramid Networks. Imbalanced learning and the focal loss. Hands-on session on training a state-of-the-art network on a small dataset.

Dense prediction problems: semantic/instance segmentation and depth from mono/stereo. Ensemble learning via bagging and random forests. The algorithm behind the Kinect body part segmentation. Fully Convolutional Networks. Transposed and dilated convolutions. RoI Align operator. Specialized NN architectures for semantic, instance, and panoptic segmentation. Deep networks for depth estimation: DispNet, GCNet, Monodepth.

Metric learning. Deep metric learning and its applications to face recognition/identification and beyond. Locally connected layers. Contrastive and triplet loss. Hands-on session on face recognition.

Attention and transformers. Image classification and object detection architectures based on Transformers.

3D Computer Vision. Intro to 3D data: point clouds, meshes, voxel grids, normals. Specialized architectures to classify, and segment point clouds.

Prerequisites:

Basic knowledge of the Python programming language

Basic knowledge of computer vision and image processing: image formation, image digitization, camera modelling and calibration, basic image manipulation and processing, local image features, basics of stereo vision.

If you attended Computer Vision and Image Processing, you already fulfill the first and second prerequisites. If you didn’t, you can ask Prof. Di Stefano access to his slides and lab sessions.

Nice-to-have knowledge, but not prerequisites

Basic knowledge of machine learning: supervised versus unsupervised learning, classification versus regression, underfitting and overfitting, regularization; data split in training, validation and test sets; hyper-parameters and cross-validation.

Basic knowledge of PyTorch: a good intro is available at https://pytorch.org/tutorials/beginner/basics/intro.html

We will briefly revise the basic ML knowledge at the beginning of the course, and the instructor will provide notebooks to introduce PyTorch and discuss them in class.

R

91264 - User Experience Design

PDF, ADOC.

Learning outcomes

At the end of the course, the student is able to design, implement and evaluate the quality of the interaction with software systems with respect to the dimensions of practicality, experience, affection, meaning and value that they may have on the target audience. The student is able to focus the functional analysis of the software on the characteristics and needs of the target audience, to drive the development process so as to guarantee a constant connection between the technical and implementation features and the expectation of the audience, and to evaluate whether and according to which metrics a software system satisfies these expectations.

Course contents

Teaching will be mostly in English (some explanations will be repeated in Italian if needed and when asked). The exam will be allowed in both Italian and English. The course content is divided in distinct parts:

Background The evolution of the discipline from Human Computer Interaction to User Experience Design. A description of its scope: the human, the computer, and their interaction.

Usability analysis and design A systematic discussion of the techniques and standards for the management of the process of user experience design, with particular attention to the phases of usability analysis (with and without the participation of users) and the user- and goal-oriented usability design methodologies.

Guidelines, patterns and methods for usability design A discussion, with historical aspects, of the framework on which the concrete aspects of usability design is based. we will also give strong attention to the problem of usability for web applications and mobile apps.

81683 - Internet of Things

PDF, ADOC.

Learning outcomes

At the end of the course, the student understands the architectures and methodologies at the basis of the Internet of Things (IoT) following a data centric approach. The student has a deep understanding of network protocols, integration of mobile and pervasive end-devices, middleware platforms for M2M-based IoT systems, edge/distributed/gateway computing principles, new services, service platforms and innovative application scenarios.

Course contents

The classes will be in English (previous years' editions were in Italian). The class will introduce the enabling technologies, protocols, software architectures and applications for the development of the emerging Internet of Things (IoT) paradigm. After a short introduction to pervasive computing and the emerging fields of applications (e.g. Industry 4.0, domotics, intelligent transportation systems, wearable devices, etc.) the class will provide a exhaustive illustration of the enabling components of IoT systems. These will include wireless communications among end-devices and towards infrastructures, data processing, languages for the development of applications and prototypes (Arduino, STM32 Nucleo, etc.). Specifically, the components of typical IoT system will be illustrated by following a data-oriented approach: from data generation on behalf of sensors to data transmission in WSAN, WPAN, WLAN, up to cloud streaming and storages, and including the data processing and integration in complex software systems.

A short summary of the course contents includes:

Introduction to IoT systems: definitions, applications, enabling technologies

IoT components: from sensors to gateway.

Sensors and actuators, sketchy illustration of design principles of electronic circuit boards, sensing and data acquisition policies.

Wireless communication and coding technologies for IoT

Machine-to-machine (M2M) communication among devices: wireless technologies for WPAN (BLE, IEEE 802.15.4, Z-Wave, etc.), WLAN and WSAN (LoRA, Dash7, Spirit, etc.)

Network architectures and routing (6LoWPAN, RPL)

Microcontrollers and System on Chip (SoC) prototypes’ programming (e.g. Arduino, STM32 Nucleo, ESP8266, etc.)

IoT components: from gateway to cloud.

Data acquisition protocols: XMPP, CoAP, MQTT, AMQP, WebSocket, etc.

Architectures for data storage and processing: cloud/fog/edge computing

IoT platforms: AWS IoT, Watson IoT, ThingSpeak, etc.

IoT & Big-data

IoT components: from cloud to applications.

Web of Things and Semantic Web 3.0 Standard for M2M and IoT

Machine learning principles (classification, clustering, regression, anomaly detection, etc.) and applications for sensor data analysis.

Frameworks for integrated development of IoT applications: AllJoyn, Google Thing, Apple HomeKit, etc.

IoT Open Issues and bottlenecks.

Privacy and Security management;

Energy efficiency for end-devices and power saving techniques;

Scalability.

R

91267 - Multi-Agent Systems

PDF, ADOC.

Learning outcomes

At the end of the course, the student understands the conceptual and technical foundation of multi-agent systems, and is able to model complex computational and socio-technical systems in terms of agent-oriented abstractions; build complex computational and socio-technical systems using agent-oriented technologies and methodologies; design and engineer intelligent systems as multi-agent systems, by integrating techniques and methods from artificial intelligence in an effective and methodologically-sound way.

Course contents

- Autonomy in natural and artificial systems
- Agent-oriented middleware for (intellligent) distributed systems

ACL & Protocols; FIPA; JADE

- Models for intelligent agents and MAS

Game Theory; Intentional agents, BDI; A&A

- Technologies for intelligent agents and MAS

Jason, ASTRA, CArtAgO

- Agreement technologies

Semantics; Norms; Organisations & Institutions; Argumentation & Negotiation; Trust & Reputation

Coordination Models & Technologies; Linda & TuCSoN

- Agent-oriented Software Engineering

GAIA & SODA

- Other contents

Simulation and MAS; NetLogo, Alchemist

Self-organisation, Swarm intelligence and MAS

Logics for MAS

91284 - Project Work in Data Mining, Text Mining and Big Data Analytics

PDF, ADOC.

Learning outcomes

At the end of the course, the student is able to apply the knowledge acquired in Data Mining, Text Mining And Big Data Analytics in order to carry out autonomously a project focusing on a topic agreed upon with the teacher.

Course contents

See 91262 - Data Mining, Text Mining and Big Data Analytics [http://www.unibo.it/en/teaching/course-unit-catalogue/course-unit/2020/446610]

91277 - Project Work in Image Processing and Computer Vision

PDF, ADOC.

Learning outcomes

At the end of the course, the student is able to apply the knowledge acquired in Image Processing And Computer Vision in order to carry out autonomously a project focusing on a topic agreed upon with the teacher.

Course contents

See "73302- Computer Vision and Image Processing M"

91287 - Project Work in Knowledge Engineering

PDF, ADOC.

Learning outcomes

At the end of the course, the student is able to apply the knowledge acquired in Knowledge Engineering in order to carry out autonomously a project focusing on a topic agreed upon with the teacher.

Course contents

Students will develop a project, under the supervision of the teacher. The project can be both individual or developed in group (of max three students). The topic and the goals of the project will be identified and agreed case by case but they will be relevant for the knowledge engineering area. A non-complete list of application areas is the following:

knowledge graph creation from text

ontology engineering

question answering based on knowledge graphs

knowledge graph visualisation

knowledge graph embeddings

ontology / knowledge graph matching

development of Protégé plugins for ontology design

development of ontology and knowledge graph visualisation tools

A non-exhaustive list of available projects is the following:

Learning from and integrating common sense resources to enable common sense reasoning

Implementation of a method for question answering over knowledge graphs

Developing a tool for supporting semi-automatic ontology testing

Developing a framework for automatic ontology evaluation

Implementation of a visualisation tool for ontology and knowledge graph summarisation

Knowledge extraction from text for populating knowledge graphs

Knowledge extraction from audio and/or music notations for populating knowledge graphs

Music recommendation based on knowledge graphs

Populating and refining knowledge graphs with games

91288 - Project Work in Machine Learning for Computer Vision

PDF, ADOC.

Learning outcomes

At the end of the course, the student is able to apply the knowledge acquired in Machine Learning For Computer Vision in order to carry out autonomously a project focusing on a topic agreed upon with the teacher.

Course contents

See "91266 - Machine Learning for Computer Vision" contents here.

93103 - Project Work in Multimedia Data Management M

PDF, ADOC.

Learning outcomes

At the end of the course, the student is able to apply the knowledge acquired in Multimedia data management M in order to carry out autonomously a project focusing on a topic agreed upon with the teacher.

Course contents

A scientific paper is usually considered as the base for the development of a new multimedia data management algorithm and/or infrastructure and/or service.

91278 - Project Work in Natural Language Processing

PDF, ADOC.

Learning outcomes

At the end of the course, the student is able to apply the knowledge acquired in Natural Language Processing in order to carry out autonomously a project focusing on a topic agreed upon with the teacher.

Course contents

This teaching activity is about developing a project in natural language processing or natural language understanding, following the course 91258 - Natural Language Processing. Upon request, the teaching staff can offer suggestions and help in defining the topic and objective of the project work. The student is, however, encouraged to take the initiative by reviewing the relevant literature and proposing a project work, for instance, about:

exploring a foundational aspect;

exploring NLP methods for low-resource languages or languages other than English;

working on a challenging application;

working on a research topic;

preparing a literature survey on a research topic;

significantly expanding a project developed within the NLP course.

The proposal has to be approved by the teaching staff before the student starts working on the project. Working as a group is encouraged.

Particularly innovative projects may receive extra support aimed to a possible submission of the work to a relevant scientific event.

91285 - Project Work in User Experience Design

PDF, ADOC.

Learning outcomes

At the end of the course, the student is able to apply the knowledge acquired in User Experience Design in order to carry out autonomously a project focusing on a topic agreed upon with the teacher.

Course contents

The module is a follow-up of the main course in Usability and User Experience Design specifically meant for students in Artificial Intelligence and discusses the following topics:

Understand that (artificial) intelligence is not just about learning and reasoning, but also about communicating (e.g. explaining to humans).

Understand what is the state-of-the-art of eXplainable AI (XAI) and how to design user-centred explanatorY AI (YAI).

We will define a common glossary on the meaning of explanation, explainability and interpretability in this context, discuss the importance of explainable and explanatory software in modern society, mentioning the recent EU’s “Right to Explanation”and works toward a user-centred explanatory software compliant with the GDPR

Read

81799 - Project Management and Soft Skills M

PDF, ADOC.

Learning outcomes

At the end of the course, the student is able to acquire the necessary skills to be a successful engineer, with a particular focus on non-technical (also known as soft) skills such as inductive/deductive/analytical/design reasoning, communication skills and employability. The student is able to use the basic tools of project management, including Gantt and Pert diagrams, work package breakdown, project cost estimation, and relevant aspects.

Course contents

The syllabus is reported below. More information on the course approach can be found on www.robertoverdone.org.

I.Introduction (approx 3 hours).

II. A Pragmatic Approach to Project Management (approx 9 hours).

- Projects, Roles, WBS
- Gantt and PERT Diagrams
- Cost Assessment
- Submitting a Project Proposal

III. Soft Skills and Aptitudes (approx 12 hours).

- Introduction to Cognitive Processes
- Soft Skills at Time of Industry 4.0
- Success and Talent
- Aptitudes
- Team Work
- Leadership
- Learning Styles

IV. Communication Skills (approx 9 hours).

- Fundamentals of Human Communication
- Persuasion
- Inter-Personal Communication
- Presentation Skills

Additionally, the course will include a Madness session where all students will introduce themselves in one minute, a CV contest, and, possibly, one or two seminars from professionals.

91276 - Project Work in Combinatorial Optimization

PDF, ADOC.

Learning outcomes

At the end of the course, the student is able to apply the knowledge acquired in Combinatorial Optimization in order to carry out autonomously a project focusing on a topic agreed upon with the teacher.

Course contents

This teaching activity is about developing a project in combinatorial optimization, following the course 91762 - Combinatorial Decision Making and Optimization. Upon request, the teacher can offer suggestions and help in defining the topic and objective of the project work. The student is, however, encouraged to take the initiative by reviewing the relevant literature and proposing a project work, for instance, about:

modelling and solving a challenging problem;

working on a research topic;

preparing a literature survey on a research topic.

The proposal has to be approved by the teacher before the student starts working on the project. Working as a group is a possibility depending on the project complexity.

91286 - Project Work in Internet of Things

PDF, ADOC.

Learning outcomes

At the end of the course, the student is able to apply the knowledge acquired in Internet Of Things in order to carry out autonomously a project focusing on a topic agreed upon with the teacher.

Course contents

The course is based on a set of project proposals focusing the Internet of things class topics, which can be selected by candidates (also proposed by candidates). The project proposal must be agreed with the teacher and formally assigned to each single candidate.

The project realization will lead to a platform, system or software, possibly mutually integrated, for the realization of IoT applications, research and development objectives.

91270 - Project Work in Fundamentals of Artificial Intelligence and Knowledge Representation

PDF, ADOC.

Learning outcomes

At the end of the course, the student is able to apply the knowledge acquired in Fundamentals of Artificial Intelligence and Knowledge Representation in order to carry out autonomously a project focusing on a topic agreed upon with the teacher.

Course contents

To be discussed with the professor

91275 - Project Work in Languages and Algorithms for Artificial Intelligence

PDF, ADOC.

Learning outcomes

At the end of the course, the student is able to apply the knowledge acquired in Languages and algorithms for Artificial Intelligence in order to carry out autonomously a project focusing on a topic agreed upon with the teacher.

Course contents

The contents of the project will be agreed with the teacher in charge of the course.

91259 - Architecture and Platforms for Artificial Intelligence

PDF, ADOC.

Learning outcomes

At the end of the course, the student has a deep understanding of the requirements of machine-learning workloads for computing systems, has an understanding of the main architectures for accelerating machine learning workloads and heterogeneous architectures for embedded machine learning, and of the most popular platforms made available by cloud providers to specifically support machine/deep learning applications.

Course contents

Module 1 (for students of 93398 and 91259, by Prof. L. Benini)

From ML to DNNs - a computational perspective

Introduction to key computational kernels (dot-product, matrix multiply…)

Inference vs training - workload analysis characterization

The NN computational zoo: DNNs, CNNs, RNNs, GNNs, Attention-based Networks

Running ML workloads on programmable processors

recap of processor instruction set architecture (ISA) with focus on data processing

improving processor ISAs for ML: RISC-V and ARM use cases

fundamentals of parallel processor architecture and parallelization of ML workloads

Algorithmic optimizations for ML

Key bottlenecks taxonomy of optimization techniques

Algorithmic techniques: Strassen, Winograd, FFT

Topology optimization: efficient NN models - depthwise convolutions, inverse bottleneck, introduction to Neural Architectural Search

Module 2 (for students of 93398, by Prof. F. Conti)

Representing data in Deep Neural Networks

Recap of canonical DNN loops – a tensor-centric view

Data quantization in Deep Neural Networks

Brief notes on data pruning

From training to software-based deployment

High-performance embedded systems (NVIDIA Xavier, Huawei Ascend)

Microcontroller-based systems (STM32)

From software to hardware acceleration

Principles of DNN acceleration: spatial and temporal data reuse; dataflow loop nests and taxonomy; data tiling

The Neural Engine zoo: convolvers, matrix product accelerators, systolic arrays – examples from the state-of-the-art

Module 2 (for students of 91259, by Prof. G. Zavattaro)

Introduction to parallel programming.

Parallel programming patterns: embarassingly parallel, decomposition, master/worker, scan, reduce, …

Shared-Memory programming with OpenMP.

OpenMP programming model: the “omp parallel” costruct, scoping costructs, other work-sharing costructs.

Some examples of applications.

Readings

91261 - Artificial Intelligence in Industry

PDF, ADOC.

Learning outcomes

At the end of the course, the student has a deep knowledge of industrial applications that benefit from the use of machine learning, optimization and simulation. The student has a domain-specific knowledge of practical use cases discussed in collaboration with industrial experts in a variety of domains such as manufacturing, automotive and multi-media.

Course contents

The course is primarily delivered as a series of simplified industrial use cases. The goal is provide examples of challenges that typically arise when solving industrial problems.

Use cases may cover topics such as:

Anomaly detection

Remaining Useful Life (RUL) estimation

RUL based maintenance policies

Resource management planning

Recommendation systems with fairness constraints

Power network management problems

Epidemic control

Production planning

The course will emphasize the ability to view problems in their entirety and adapt to their peculiarities. This will frequently require to combine heterogeneous solution techniques, using integration schemes both simple and advanced.

The employed method will include:

Mathematical modeling of industrial problems

Predictive and diagnostic models for time series

Combinatorial Optimization

Integration methods for Probabilistic Models and Machine Learning

Integration methods for constraints and Machine Learning

Integration methods for combinatorial optimization and Machine Learning

The course will include seminars on real-world use cases, from industry experts.

The course contents may (and typically will) be subject to changes, so as to adapt to some degree to the interests and characteristics of the attending students.

90748 - Blockchain and Cryptocurrencies

PDF, ADOC.

Learning outcomes

At the end of the course, the student knows the relevant themes related to blockchain technologies, cryptocurrencies, smart contracts and novel applications that can be built over the blockchain. The student is able to develop simple smart contracts that can be deployed on a blockchain.

Course contents

Bitcoin and novel cryptocurrencies gathered momentum in the last months. More and more investors look with interest at these technologies, while others label them as a dangerous speculative bubble. The truth is that the blockchain, and the alternative implementations of a distributed ledger, represent very interesting technologies, that can be exploited to build novel distributed applications. The underlying building blocks are related to many concepts and research areas of computer science in general. This course will illustrate the main principles and conceptual foundations of the blockchain and the Bitcoin network.

Program

Introduction to peer-to-peer sysystems

Overlay topologies and decentralization

Introduction to Crypto and Cryptocurrencies

The blockchain: how to achieve decentralization

Transactions and transaction scripting languages

Mining

Attacks to the blockchain

Anonymity

Smart contracts

81943 - Complex Systems & Network Science

PDF, ADOC.

Learning outcomes

At the end of the course, the student has basic notions of complexity and network sciences and is able to identify, formulate, model and analyze new problems that arise in modern computing systems that can be studied using them.

Course contents

Description: Modern information systems and services often rely on large numbers of independent interacting components to provide their functions. Under certain conditions, the behavior that results from these interactions can be unexpected and surprising. Complexity Science is an interdisciplinary field for studying global behaviors resulting from many simple local interactions in an effort to characterize and control them. Networks allow us to formalize the structure of interactions. They play a central role in the transmission of information, transportation of goods, spread of diseases, diffusion of innovation, formation of opinions and adoption of new technologies. Network Science is an interdisciplinary field for studying the interconnectedness of modern life by exploring fundamental properties that govern the structure and dynamic evolution of networks.

Contents: Complex systems: definitions, methodologies; Dynamical systems, Nonlinear dynamics; Chaos, Bifurcations and Feigenbaum constant, Predictability, Randomness and Chaos; Models of complex systems, Cellular automata, Wolfram's classification, Game of life; Autonomous agents, Flocking, Schooling, Synchronization, Formation creation; Cooperation and Competition, Game theory basics, Nash equilibrium; Game theory: Prisoner's Dilemma, Coordination games, Mixed strategy games; Adaptation, Evolution, Genetic algorithms, Evolutionary games; Network Science: Definitions and examples; Graph theory, Basic concepts and definitions; Diameter, Path length, Clustering, Centrality metrics; Structure of real networks, Degree distribution, Power-laws, Popularity; Models of network formation; The Erdos-Renyi random model; Clustered models; Models of network growth, Preferential attachment; Small-world networks, Network navigation; Peer-to-peer systems and overlay networks; Structured overlays, DHTs, Key-based routing, Chord; Distributed network formation: Newscast, Cyclon, T-Man; Processes on networks: Aggregation; Rational dynamics: Cooperation in selfish environments, Homophily, Segregation; Diffusion, Percolation, Tipping points, Peer-effects, Cascades.

Prerequisites: Basic notions of computer system architecture, computer networks, operating systems, and probability theory.

93470 - CYBERSECURITY

PDF, ADOC.

Learning outcomes

The student will learn: historical perspective and motivations for cyber-security, principles of data security, principles of software security, component and connection security, network security.

Course contents

Scenarios and strategies

A top-down approach to cybersecurity

Information security

Cyber threats to industry

Cyber risk analysis

Qualitative and quantitative evaluation

Management of cyber risks

Technologies and methodologies for cybersecurity

Assessment and validation

Machine Learning for cybersecurity

Disruptive solutions

Economic, social and political issues of the digital society

91262 - Data Mining, Text Mining and Big Data Analytics

PDF, ADOC.

Learning outcomes

At the end of the course, the student understands how a possibly very large set of data can be analyzed to derive strategic information and to address "data-driven" decisions. The student has a knowledge of the main data-mining tasks such as data selection, data transformation, analysis and interpretation, with specific reference to unstructured text data, and with the issues related to analysis in "big data" environments.

Course contents

Presentation slides of the course

Module 1 - Data Mining (Claudio Sartori)

See 75194 - DATA MINING M [https://www.unibo.it/en/teaching/course-unit-catalogue/course-unit/2021/391683] Module 2 only

Module 2 - Text Mining (Gianluca Moro)

The text mining module focuses on knowledge discovering from large corpora of unstructured text which is fundamental to deal with several natural language processing tasks, such as text representation models, indexing and classification, analysis of topics, semantic similarity search, explaining behaviours and phenomenon of interest (a.k.a. descriptive text mining), sentiment analysis and opinion mining, text summarisation, chatbot or digital assistant creation, cross-modal information retrieval of texts and images etc.

The learning outcomes of the module are the capabilities of defining and implementing text mining processes, from text pre-processing and representation with traditional approaches and then with novel neural language models, up to the knowledge discovery with data science methods and machine & deep learning algorithms from several sources, such tweets, facebook posts, reviews, web pages, emails, loan requests, legal cases, news and documents in general.

The module introduces non-contextual language models based on word embeddings, such as GloVe and word2vec, and memory based neural networks particularly effective for textual data, such as recurrent neural networks like LSTM, GRU and BiLSTM, up to the attention mechanism, the transformer and the state-of-the-art of contextual word embeddings based on BERT and new linear transformer models for text summarization and deep metric learning for cross-modal information retrieval of texts and images.

Last but not least, the unit illustrates the transfer learning paradigm to exploit and fine tune existing models in target domains which are semantically different from their training source domains; this is particularly useful in order to overcome the lack of labeled data in the target domain.

Module 3 - Big Data Analytics (Stefano Lodi)

Maps and reductions in parallel programming. The MapReduce programming model.

The Hadoop implementation of MapReduce.

The Python API to the Spark system and examples of parallel programs.

The Machine Learning

Library (MLlib) of Spark.

Read

91265 - Knowledge Engineering

PDF, ADOC.

Learning outcomes

At the end of the course, the student knows some (semi-)automated methods for joint interpretation of data and content as sources of knowledge. The student masters the basics of knowledge extraction, engineering, and linking, making data suitable to machine querying and automated reasoning, typically on decentralized platforms such as the Web.

Course contents

Knowledge graphs and ontologies

OWL, RDF, SPARQL

Ontology Design methodologies, focus on Extreme Design

Intensional and Extensional modelling

Ontology design patterns

Applied reasoning

Knowledge extraction from text

Ontology evaluation and quality

91266 - Machine Learning for Computer Vision

PDF, ADOC.

Learning outcomes

At the end of the course, the student masters the most popular modern machine-learning approaches to computer-vision tasks, with particular reference to specialized deep-learning architectures. The student has both a theoretical understanding and the necessary practical skills required to develop state-of-the-art image and video analysis systems for real-world applications.

Course contents

The course is held in the first semester, from September to December.

Topics:

Image classification/recognition: limits of hand-crafted methods; brief review of machine learning and data-driven methods; bag-of-words models applied to dense descriptors; introduction to Neural Networks (NNs) and Convolutional NNs; AlexNet and the deep learning revolution. Advanced and state-of-the-art architectures for image classification. Hands-on sessions on image classification: training from scratch and transfer learning.

Intro to PyTorch and practical aspects of model training. Discussion of notebooks provided by the instructor to get to know PyTorch and learn how to train CNNs for image classification; discussion on the impact of several design and optimization choices.

Object detection. Introduction to ensemble learning via boosting. The Viola-Jones detector and its applications. Specialized NN architectures for object detection. Two-stages, one-stage, and anchor-free detectors. RoI Pooling operator, Feature Pyramid Networks. Imbalanced learning and the focal loss. Hands-on session on training a state-of-the-art network on a small dataset.

Dense prediction problems: semantic/instance segmentation and depth from mono/stereo. Ensemble learning via bagging and random forests. The algorithm behind the Kinect body part segmentation. Fully Convolutional Networks. Transposed and dilated convolutions. RoI Align operator. Specialized NN architectures for semantic, instance, and panoptic segmentation. Deep networks for depth estimation: DispNet, GCNet, Monodepth.

Metric learning. Deep metric learning and its applications to face recognition/identification and beyond. Locally connected layers. Contrastive and triplet loss. Hands-on session on face recognition.

Attention and transformers. Image classification and object detection architectures based on

Transformers.

3D Computer Vision. Intro to 3D data: point clouds, meshes, voxel grids, normals. Specialized architectures to classify, and segment point clouds.

Prerequisites:

Basic knowledge of the Python programming language

Basic knowledge of computer vision and image processing: image formation, image digitization, camera modelling and calibration, basic image manipulation and processing, local image features, basics of stereo vision.

If you attended Computer Vision and Image Processing, you already fulfill the first and second prerequisites. If you didn’t, you can ask Prof. Di Stefano access to his slides and lab sessions.

Nice-to-have knowledge, but not prerequisites

Basic knowledge of machine learning: supervised versus unsupervised learning, classification versus regression, underfitting and overfitting, regularization; data split in training, validation and test sets; hyper-parameters and cross-validation.

Basic knowledge of PyTorch: a good intro is available at https://pytorch.org/tutorials/beginner/basics/intro.html

We will briefly revise the basic ML knowledge at the beginning of the course, and the instructor will provide notebooks to introduce PyTorch and discuss them in class.

R

90730 - SOCIAL NETWORK ANALYSIS

PDF, ADOC.

Learning outcomes

At the end of the course, students gain knowledge on the Web as a socio-technical system involving specific processes, entities, and behaviours, using interdisciplinary methods that blend computer science, sociology, ethnography, economics, linguistics, etc. The students are able to analyse the Web phenomena similarly to typical objects from natural sciences, distinguishing between data and applications, agents from computationally generated profiles, and addressing the characteristics of networks of entities emerging from the informationl, physical, social, and conceptual spaces constituting the Web.

Course contents

The programme is the same for both attending and non-attending students:

Introduction to Network Analysis, gentle introduction to the field of network analysis and its usages in other fields of research (e.g., computer science, forensics, archeology, literature, history, science of religion, etc.).

Research Design and How to Read a (Network Analysis) Research Paper: introduction to the scientific publication process, elements of research papers (on network analysis), research design, analysis of research papers.

Mathematics of Networks: networks and their representation, types of networks, graph representations, paths and components, adjacency matrices and matrix representations, ways and modes, operations on Matrices.

Data Collection and Data Management: network questions, data collection and reliability, data formats and transformation, algorithms and software for network analysis and visualisation.

Measures and Metrics, Nodes: kinds of measures, multi-category nominal scales, ordinal and scalar measures, centrality, degree and other kinds of centrality (e.g. Google's PageRank), hubs and authorities, closeness and betweenness centrality, groups of nodes (cliques, cores, components and k-components), clustering and clustering coefficients, reciprocity and similarity, structural and other types of equivalence, homophily and types of assortative mixing.

Testing Hypotheses: the role of hypotheses in the scientific method, testing hypotheses in network analysis, permutation tests, dyadic hypotheses.

Ego Networks: ego networks, obtaining and analysing ego-network data, tie analysis, structural-shape measures.

Measures and Metrics, Networks: small-world effects, degree distribution, power laws and scale-free networks, visualisation and properties of power-law distributions, local-clustering coefficient, cohesion, reciprocity, transitivity and the clustering coefficient, triad census, centralisation and core-periphery indices, centrality, random graphs, means on edges and degree, degree distribution, giant and small component(s), locally tree-like networks.

Dividing Networks into Groups: dividing networks into groups, modularity maximisation, methods based on information theory, methods based on statistical inference, betweenness-based methods (dendrograms), hierarchical clustering, overlapping communities, hierarchical communities, latent spaces, stratified networks, and rank structure, percolation and network resilience, uniform and non-uniform (random) node removal.

Network Visualisation: the importance of network visualisation, graph-layout algorithms, embedding node attributes, node filtering, visualising ego networks, embedding tie characteristics, tie strengths, visualising network change.

Handling Large Networks: reducing the size of the problem, eliminating edges, pruning nodes, divide and conquer, aggregation, sampling, small-world and scale-free networks.

91264 - User Experience Design

PDF, ADOC.

Learning outcomes

At the end of the course, the student is able to design, implement and evaluate the quality of the interaction with software systems with respect to the dimensions of practicality, experience, affection, meaning and value that they may have on the target audience. The student is able to focus the functional analysis of the software on the characteristics and needs of the target audience, to drive the development process so as to guarantee a constant connection between the technical and implementation features and the expectation of the audience, and to evaluate whether and according to which metrics a software system satisfies these expectations.

Course contents

Teaching will be mostly in English (some explanations will be repeated in Italian if needed and when asked). The exam will be allowed in both Italian and English. The course content is divided in distinct parts:

Background The evolution of the discipline from Human Computer Interaction to User Experience Design. A description of its scope: the human, the computer, and their interaction.

Usability analysis and design A systematic discussion of the techniques and standards for the management of the process of user experience design, with particular attention to the phases of usability analysis (with and without the participation of users) and the user- and goal-oriented usability design methodologies.

Guidelines, patterns and methods for usability design A discussion, with historical aspects, of the framework on which the concrete aspects of usability design is based. we will also give strong attention to the problem of usability for web applications and mobile apps.

92858 - Autonomous and Adaptive Systems M

PDF, ADOC.

Learning outcomes

The goal of this module is to provide a solid introduction to the design of autonomous and adaptive computing systems from a theoretical and practical point of view. Topics will include principles of autonomous system design, reinforcement learning, game-theoretic approaches to cooperation and coordination, bio-inspired systems, complex adaptive systems, and computational social systems. The module will also cover several practical applications from a variety of fields including but not limited to distributed and networked systems, mobile

1) and ubiquitous systems, robotic systems, and vehicular and transportation systems.

2) Si propone la modifica degli obiettivi formativi delle seguenti attività:

Course contents

Introduction to the design of adaptive and autonomous systems, intelligent agents and intelligent machines.

Review of Machine Learning/Deep Learning fundamentals.

Introduction to Reinforcement Learning (RL): multi-armed bandits, Montecarlo methods, tabular methods, approximation function methods, and deep RL.

Applications of RL to games.

Applications of RL to classic control theory problems and robotics.

RL and cognitive sciences&neuroscience.

Intelligent machines and creativity: Generative Deep Learning, AI and the Arts.

Models of human, artificial and hybrid societies.

Algorithmic game theory and AI-based decision-making: cooperation and coordination, social dilemmas, and Multi-Agent Reinforcement Learning.

Bio-inspired adaptive systems.

Autonomous and mobile robots.

Driverless cars and autonomous transportation systems.

Ethical implications of AI and autonomous systems.

Machine intelligence, super-intelligence, self-awareness, controllability and the future.

The course will include labs in which we will discuss implementation oriented aspects of the techniques and methodologies presented during the course.

Re

81683 - Internet of Things

PDF, ADOC.

Learning outcomes

At the end of the course, the student understands the architectures and methodologies at the basis of the Internet of Things (IoT) following a data centric approach. The student has a deep understanding of network protocols, integration of mobile and pervasive end-devices, middleware platforms for M2M-based IoT systems, edge/distributed/gateway computing principles, new services, service platforms and innovative application scenarios.

Course contents

The classes will be in English (previous years' editions were in Italian). The class will introduce the enabling technologies, protocols, software architectures and applications for the development of the emerging Internet of Things (IoT) paradigm. After a short introduction to pervasive computing and the emerging fields of applications (e.g. Industry 4.0, domotics, intelligent transportation systems, wearable devices, etc.) the class will provide a exhaustive illustration of the enabling components of IoT systems. These will include wireless communications among end-devices and towards infrastructures, data processing, languages for the development of applications and prototypes (Arduino, STM32 Nucleo, etc.). Specifically, the components of typical IoT system will be illustrated by following a data-oriented approach: from data generation on behalf of sensors to data transmission in WSAN, WPAN, WLAN, up to cloud streaming and storages, and including the data processing and integration in complex software systems.

A short summary of the course contents includes:

Introduction to IoT systems: definitions, applications, enabling technologies

IoT components: from sensors to gateway.

Sensors and actuators, sketchy illustration of design principles of electronic circuit boards, sensing and data acquisition policies.

Wireless communication and coding technologies for IoT

Machine-to-machine (M2M) communication among devices: wireless technologies for WPAN (BLE, IEEE 802.15.4, Z-Wave, etc.), WLAN and WSAN (LoRA, Dash7, Spirit, etc.)

Network architectures and routing (6LoWPAN, RPL)

Microcontrollers and System on Chip (SoC) prototypes' programming (e.g. Arduino, STM32 Nucleo, ESP8266, etc.)

IoT components: from gateway to cloud.

Data acquisition protocols: XMPP, CoAP, MQTT, AMQP, WebSocket, etc.

Architectures for data storage and processing: cloud/fog/edge computing

IoT platforms: AWS IoT, Watson IoT, ThingSpeak, etc.

IoT & Big-data

IoT components: from cloud to applications.

Web of Things and Semantic Web 3.0 Standard for M2M and IoT

Machine learning principles (classification, clustering, regression, anomaly detection, etc.) and applications for sensor data analysis.

Frameworks for integrated development of IoT applications: AllJoyn, Google Thing, Apple HomeKit, etc.

IoT Open Issues and bottlenecks.

Privacy and Security management;

Energy efficiency for end-devices and power saving techniques;

Scalability.

R

91267 - Multi-Agent Systems

PDF, ADOC.

Learning outcomes

At the end of the course, the student understands the conceptual and technical foundation of multi-agent systems, and is able to model complex computational and socio-technical systems in terms of agent-oriented abstractions; build complex computational and socio-technical systems using agent-oriented technologies and methodologies; design and engineer intelligent systems as multi-agent systems, by integrating techniques and methods from artificial intelligence in an effective and methodologically-sound way.

Course contents

- Autonomy in natural and artificial systems
- Agent-oriented middleware for (intellligent) distributed systems

ACL & Protocols; FIPA; JADE

- Models for intelligent agents and MAS

Game Theory; Intentional agents, BDI; A&A

- Technologies for intelligent agents and MAS

Jason, ASTRA, CArtAgO

- Agreement technologies

Semantics; Norms; Organisations & Institutions; Argumentation & Negotiation; Trust & Reputation

Coordination Models & Technologies; Linda & TuCSoN

- Agent-oriented Software Engineering

GAIA & SODA

- Other contents

Simulation and MAS; NetLogo, Alchemist

Self-organisation, Swarm intelligence and MAS

Logics for MAS

77933 - Multimedia Data Management M

PDF, ADOC.

Learning outcomes

At the end of the course, the student has the knowledge and skills required for an effective and efficient management of multimedia (MM) data, with particular attention to the problems of MM data representation, MM data retrieval models, and interaction paradigms between the user and the MM system both for purposes of data presentation and exploration. The student understands the architecture of traditional and advanced MM systems and services, search engines, social networks and recommendation systems.

Course contents

Basics on Multimedia Data Management

Multimedia data and content representations

MM data and applications

MM data coding

MM data content representation

How to find MM data of interest

Description models for complex MM objects

Similarity measures for MM data content

MM Data Base Management Systems

Efficient algorithms for MM data retrieval

MM query formulation paradigms

Sequential retrieval of MM data

Index-based retrieval of MM data

Automatic techniques for MM data semantic annotations

Browsing MM data collections

MM data presentation

User interfaces

Visualization paradigms

Dimensionality reduction techniques

Result accuracy, use cases and real applications

Quality of the results and relevance feedback techniques

Use cases and demos of some applications

Multimedia Data on the Web

Web search engines

Graph-based data: semantic Web and social networks

Web recommender systems

N.B. For students of the second cycle degree programmes (LM) in Artificial intelligence and Computer Science, the "Efficient algorithms for MM data retrieval" part of the program is not required.

81799 - Project Management and Soft Skills M

PDF, ADOC.

Learning outcomes

At the end of the course, the student is able to acquire the necessary skills to be a successful engineer, with a particular focus on non-technical (also known as soft) skills such as inductive/deductive/analytical/design reasoning, communication skills and employability. The student is able to use the basic tools of project management, including Gantt and Pert diagrams, work package breakdown, project cost estimation, and relevant aspects.

Course contents

The syllabus is reported below. More information on the course approach can be found on www.robertoverdone.org.

I.Introduction (approx 3 hours).

II. A Pragmatic Approach to Project Management (approx 9 hours).

- Projects, Roles, WBS
- Gantt and PERT Diagrams
- Cost Assessment
- Submitting a Project Proposal

III. Soft Skills and Aptitudes (approx 12 hours).

- Introduction to Cognitive Processes
- Soft Skills at Time of Industry 4.0
- Success and Talent

- Aptitudes
- Team Work
- Leadership
- Learning Styles

IV. Communication Skills (approx 9 hours).

- Fundamentals of Human Communication
- Persuasion
- Inter-Personal Communication
- Presentation Skills

Additionally, the course will include a Madness session where all students will introduce themselves in one minute, a CV contest, and, possibly, one or two seminars from professionals.

97431 - SCALABLE AND RELIABLE SERVICES M

PDF, ADOC.

Learning outcomes

Digital services have to meet various requirements in terms of performance, scalability and reliability. We analyze the whole lifecycle of high quality services, from design to deployment, testing and operation. In each phase, we aim to provide the students with the ability to gather and analyze data, to identify sources of outages and critical dependencies, and to avoid bottlenecks and single points of failure. Finally, we expect students to be prepared to evaluate and discuss alternatives and justify investment in support of business services.

Course contents

Introduction

Cloud services

Cloud provider architectures

High-quality services: performance, scalability, reliability (and security)

Lifecycle of high-quality services

Part 1: Architectures for scalable and reliable services

Sources of outages and critical dependencies

Bottlenecks and single points of failure

Design: maturity assessment, architectural principles for reliability and scalability, AKF scale cube model

Migration to cloud

Critical data management: Strategy, Governance, Management

Implementation alternatives on cloud platforms and multi-clouds

Part 2: Service deployment

Secure and robust software production

Secure and robust DevOps

Labs on cloud platforms

Part 3: Tests

Benchmarking

Performance testing

Stress testing

Part 4: Scalable and reliable organizations

Roles and responsibilities

Leaders and managers

Putting all together

Capacity planning

Understanding and managing complexity

Managing changes

Part 5: Issues, incident and crises

Performance monitoring

Warning signs in operation

Incidents management

Crisis management