## English-taught courses for exchange students
### Winter Semester (WS) & Summer Semester (SS)

## Faculty of Computer Science (INF)

### Courses taught in English

#### Bachelor

<table>
<thead>
<tr>
<th>Course No.</th>
<th>Abbreviation and Title</th>
<th>WS / SS</th>
<th>Semester</th>
<th>ECTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>FWPM A2l2</td>
<td>Applications of &amp; Introduction to AI</td>
<td>SS</td>
<td>INF B3-7</td>
<td>5 ECTS</td>
</tr>
</tbody>
</table>

#### Master

<table>
<thead>
<tr>
<th>Course No.</th>
<th>Title and Timetable Abbreviation</th>
<th>WS / SS</th>
<th>Semester</th>
<th>ECTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>FWPM InPM</td>
<td>Innovation and Product Management/ Innovation und Produktmanagement (InPM)</td>
<td>WS</td>
<td>INF M1-3</td>
<td>5 ECTS</td>
</tr>
<tr>
<td>FWPM Data</td>
<td>Data Science (Data)</td>
<td>WS</td>
<td>INF M1-3</td>
<td>5 ECTS</td>
</tr>
<tr>
<td>FWPM ML</td>
<td>Machine Learning/ Maschinelles Lernen (ML)</td>
<td>SS</td>
<td>INF M1-3</td>
<td>5 ECTS</td>
</tr>
<tr>
<td>FWPM SPP</td>
<td>Smartphone Programming/ Smartphone Programmierung (SPP)</td>
<td>SS</td>
<td>INF M1-3</td>
<td>5 ECTS</td>
</tr>
<tr>
<td>FWPM SL</td>
<td>Sequence Learning (SL)</td>
<td>SS</td>
<td>INF M1-3</td>
<td>5 ECTS</td>
</tr>
</tbody>
</table>

Applications of & Introduction to AI ................................................................. 2
InPM: Innovation and Product Management ......................................................... 4
Data: Data Science ............................................................................................... 5
ML: Machine Learning ........................................................................................... 7
SPP: Smartphone Programming ............................................................................... 9
SL: Sequence Learning .......................................................................................... 11
BACHELOR

Applications of & Introduction to AI

Responsible for module: Professor Dr. Markus Breunig

Course: Bachelor

Frequency of module: Once a year - summer semester

Compulsory/FWPM: FWPM (subject specific elective)

Prerequisites: An open, inquisitive and curious mind and basic object-oriented coding skills.

Language: English.

Form of teaching: 4 hours per week per semester, seminar lecture

Workload: 60 hours attendance in person, 90 hours self-study/project work

ECTS credits: 5

Media, teaching material: Presentations, practical exercises, project work, hands-on coding

Examination: See course assessment notification

Learning Outcomes & Content

Knowledge / Skills / Abilities / Competencies:

Artificial Intelligence (AI) is a field that has a long history but is still constantly and actively growing and changing. In this course participants will learn what Artificial Intelligence (AI) is, explore use cases and applications of AI, understand AI concepts and terms like machine learning and deep learning.

After this course students will be able to explain concepts of AI and machine learning algorithms to others and compare results of various machine learning algorithms on given data sets. They will have the skills to utilize AI frameworks for machine learning uses cases ad to use services providing AI capabilities and to apply them in applications.

Short module description:

In this course, participants learn the basics of modern AI as well as some of the representative applications of AI. The broad-ranging discussion covers the key subdisciplines within the field, describing practical algorithms and concrete applications via hands-on coding in Python within a (Jupyter-)Notebook-based environment.

Students will leave excited about the numerous applications and huge possibilities in the field of AI, which continues to expand human capability beyond our imagination.
Agenda:

1. Introduction to Machine Learning and Artificial Intelligence
2. Notebook-based Python Coding Environments
3. Classification and Classifier Evaluation
4. Recommender Systems
5. Predictive Maintenance
6. Image Processing with SVM and Deep Learning
7. AI services for composing applications

Reading List & Media

Recommended:


Additionally recommended:

**MAster**

**InPM: Innovation and Product Management**

**Responsible for module:** Professor Dr. Markus Breunig

**Course:** Master (M6 – Project Management & Leadership)

**Frequency of module:** Once a year - **winter semester** (depending on demand, min. 6 participants)

**Compulsory/FWPM:** FWPM (subject specific elective)

**Prerequisites:** Good English skills, elementary knowledge of marketing, basic skills in engineering

**Language:** Written materials are mostly in English. Class languages is English unless only German speaking students attend.

**Form of teaching:** 4 hours per week per semester, seminar lecture with exercises

**Workload:** 60 hours attendance in person, 90 hours self-study/project work

**ECTS credits:** 5

**Media forms:** Lecture, project work, exercises, presentations with beamer and blackboard

**Examination:** See course assessment notification

**Literature:**

**Print:**


**Medien/online**

1. Gestalten statt Verwalten by Jan Erik Baars: de.slideshare.net/designfokus/gestalten-statt-verwalten
2. Design Methodenfinder by V. Fischer, W. Nagel, M. Ottmann & T. Söffing : www.designmethodenfinder.de
3. ABC Nightline - IDEO Shopping Cart: www.youtube.com/watch?v=M66ZU2PClCM
Data: Data Science

Responsible for module: Professor Dr. Markus Breunig

Course: Master Computer Science

Frequency of module: Once a year - winter semester

Compulsory/FWPM: FWPM (subject specific elective)

Prerequisites:
- Good English and German skills
- Bachelor-level knowledge about (relational) Database Systems.
- Bachelor-level knowledge in Statistics, probability theory, and object-oriented programming.
- Basic knowledge about Data Warehouse Systems and Big Data technologies.

Language: Written materials are in English, the class is taught in German.

Form of teaching: 4 hours per week per semester, seminar lecture with exercises

Workload: 150 hours
- 60 hours attendance in person
- 90 hours self-study/project work

ECTS credits: 5

Objectives and Content:

Course objectives:
- Knowledge about the fundamentals of data science.
- Understanding and mastery of the most important algorithms and applications of typical data science problems.
- Ability to solve given data science problems with popular (esp. open-source) tools.
- Team-working skills and presentation skills.

Short course description: Based on the basics of databases, data warehouses, and big data systems, an overview of objectives and application areas provides a practical explanation of the conceptual and statistical fundamentals of data sciences.

With the help of the "notebook" approach the concrete programming language Python is introduced and the most important libraries available for data analysis are applied to specific problems.

This includes both methods of data analysis as well as relevant algorithms from the fields of classification, clustering, outlier detection, frequent pattern mining, sentiment analysis, graph analysis, predictive and prescriptive modeling, and text-mining.
For large amounts of data, the possibility of a scale-up using distributed computation is essential and is considered based on Apache Spark, using the Spark-implementation language Scala.

All written materials are in English, the class itself is taught in German.

Content:

1. Python and the iPython Notebook
2. Basics of Data Science
3. Data Preprocessing – Pandas und NumPy
4. Basics Data Mining algorithm and scikit.learn
5. Introduction to Scala
6. Apache Spark and Zeppelin Notebooks
7. RDD, Persistence, Spark SQL
8. Sentiment Analysis, Machine Learning, Graph Analysis
9. Prescriptive (Optimization) Modelling

Media forms: Lecture, project work, exercises, presentations with beamer and blackboard

Examination: See course assessment notification

Literature:

Provost, Fawcett: Data Science for Business (2013)
Han, Kamber: Data Mining. Concepts and Techniques. (2006)

Methods of teaching and: Inverted classroom using Notebooks.
ML: Machine Learning

Responsible for module: Professor Dr. Breunig/Riedhammer/Schmidt

Course: Master Computer Science

Frequency of module: Once a year - summer semester

Compulsory/FWPM: FWPM (subject specific elective)

Prerequisites: - Advanced English and German.
               - Math and Algorithm-Know-How.

Language: Written materials are in English, the class is taught in German.

Form of teaching: Blended learning and project work

Workload: 150 hours
          - 60 hours attendance in person
          - 90 hours self-study/project work

ECTS credits: 5

Media forms: Self-study based on a video lecture in English; material on the videos worked out (in English); in-person meetings and virtual group work (German or English depending on participants); one-on-one coaching during the project.

Examination: See course assessment notification

Objectives and Content:

Course objectives: The students know and understand the theoretical and algorithmic principles of machine learning. They can analyze difficult subject-specific problems scientifically and understand complex contexts and implement them in software through targeted methods.

Content overview: The students develop the theoretical and algorithmic foundations of machine learning: (statistical) preprocessing, clustering, classification, regression, and the craft of scientific experimentation. This class complements the data science (Data), Sequence Learning (SL) and digital image processing (DBV) classes, which show practical applications of the methods to background knowledge in the field of machine learning. ML and at least one other of these courses are recommended.

The written materials for the course are in English, the lectures are taught in German.
Contents:

1. Introduction and Overview
2. Classification: k-Nearest-Neighbor and Logistic Regression
3. Linear and Polynomial Regression
4. Clustering: k-means/k-medoid, LGB Vector-Quantization, Gaussian Mixtures and Expectation Maximization (EM)
5. Optimal Classifier, Naive Bayes, and Gaussian Classifier
6. Decision Trees: CART/C4.5 and Random Forests
7. Artificial Neural Networks
8. Support Vector Machines
9. Classifier Evaluation: Test/Training, ROC Curves, Cross-Validation, Bootstrapping
10. Feature Normalization
11. Principal Component Analysis (PCA)
12. Feature Rating and Feature Selection

Literature


Methods of teaching: Presentation (Laptop, Projector); Material in the community and exercises.
SPP: Smartphone Programming

Responsible for module: Professor Dr. Markus Breunig
Course: Master
Frequency of module: Once a year - summer semester
Compulsory/FWPM: FWPM (subject specific elective)
Prerequisites: A Bachelor's qualification, good English language skills
Language: The class follows the blended learning concept. Online (Video) lectures are in English, all written material is in English. Class lectures are in English unless only German speaking students attend. Individual coaching sessions are either in English or German. Student papers (specification, etc.) are to be submitted in English, student presentations can be in English or German.

Form of teaching: Blended learning and project work
Workload: 30 hours self-study, 30 hours group work, 90 hours project work
ECTS credits: 5
Media forms: Self-study based on a video lecture in English; material on the videos worked out (in English); in-person meetings and virtual group work (German or English depending on participants); one-on-one coaching during the project.

Examination: See course assessment notification

Objectives and Content:

Course objectives: Learn a programming language in widespread use in technology and use it in combination with operating systems used in embedded systems. Capability to develop and implement software in the target language on the target systems. Learn to work with different tools for the development and operation of applications. Improve employability in the international market by means of the practical use of written and spoken technical English. Learn independent time and goal planning as part of project work. Independent performance of a project. Improve presentation skills.

Content overview: The course offers an introduction to Objective-C and the tools needed to build iOS applications (apps). The focus is on the details of creating applications with iOS. Suitable examples will be used to learn the details of the object-oriented development language as well as the frameworks used, and practical exercises will involve isolated examples. The focus will be on learning the UI
framework Cocoa Touch, the use of object-oriented patterns in programming, as well as memory management and iOS.

**Implementation:**
The course will be carried out in a "blended learning" style. The actual lectures (in English) will be provided as video files and students will work through them at their own pace as self-study within the first phase of the course. Review sessions on the material will take place in parallel (some as in-person meetings, some virtual). In the second phase of the course, the students will implement a (self-selected) programming project in iOS/Objective-C. During this phase, additional review sessions and individual coaching will take place (both as virtual meetings and in person). The projects will be presented and discussed before the group at the end of the course.

**Contents:**
- Overview of iOS, MVC, Objective-C
- Xcode 5
- Objective-C
- Foundation and Attributed Strings
- View Controller Lifecycle
- Polymorphism with Controllers, UINavigation, UITabBar
- Views and Gestures
- Protocols, Blocks, and Animation
- Animation and Autolayout
- Multithreading, Scroll View
- Table View and iPad
- Documents and Core Data
- Core Data and Table View
- UIApplication, Network Activity Indicator, and Maps
- MapKit and Embed Segue
- Modal Segues, Text Fields, Alerts, and Action Sheets
- Camera, Core Motion, Application Lifecycle
- Localization, Adding UI to Settings

**Literature:**
**SL: Sequence Learning**

**Responsible for module:** Prof. Dr.-Ing. Korbinian Riedhammer

**Course:** Master Computer Science

**Frequency of module:** Once a year - *summer term*

**Compulsory/FWPM:** FWPM (subject specific elective)

**Prerequisites:**
- Advanced English, German recommended
- Math and Algorithm-Know-How.

**Language:** Written materials are in English, *the class is taught in German (Instructor is fluent in English).*

**Form of teaching:** Lecture and assignments/project (alternating)

**Workload:** 150 hours
- 60 hours attendance in person
- 90 hours self-study/project work

**ECTS credits:** 5

**Media forms:** Books, supplemental syllabus, in-person meetings and virtual group work (German or English depending on participants); one-on-one coaching during the project.

**Examination:** See course assessment notification

**Objectives and Content:**

**Course objectives:** The students know and understand the theoretical and algorithmic principles of time series analysis (sequence learning). They can analyze difficult subject-specific problems scientifically and understand complex contexts and implement them in software through targeted methods.

**Content overview:** The students develop the theoretical and algorithmic foundations of time series analysis: dynamic time warping, sequence kernels for classification, hidden Markov models, recurrent neural networks, sequence-to-sequence models, and their practical application. It is recommended (but not required) to attend the Machine Learning (ML) class, which covers the basics in classification, regression and scientific experimentation.

The written materials for the course are in English, the lectures are taught in German.
Contents:

1. Dynamic Time Warping (DTW)
2. Markov Chains
3. Hidden Markov Models
4. Classification: Windowing and Sequence Kernels
5. Recurrent Neural Networks
6. Sequence-to-Sequence learning

Literature


Huang, Acero, Hon: Spoken Language Processing: A Guide to Theory, Algorithm and System Development


Methods of teaching:

Presentation (Laptop, Projector); materials in the community; exercises.