

Modulhandbuch

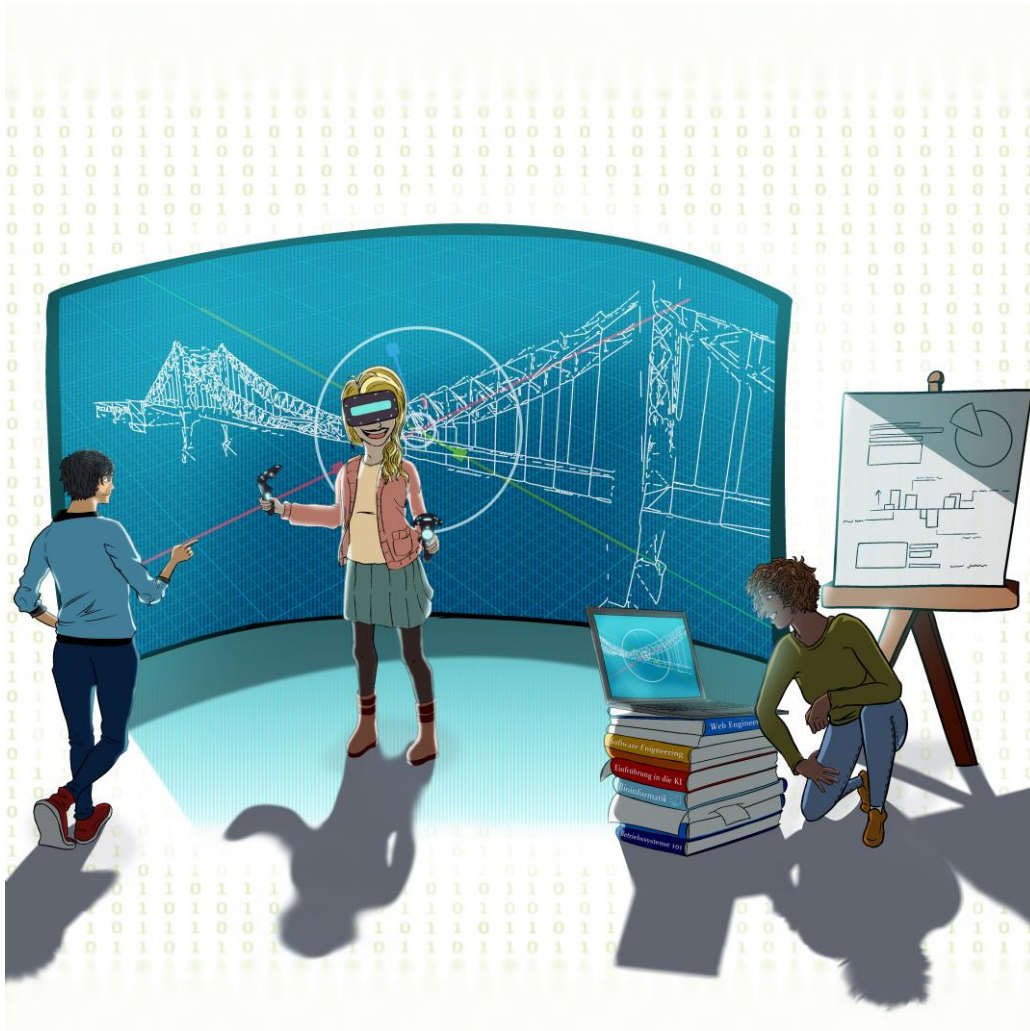
Master of Science (M.Sc.)

Angewandte Informatik [PO 20]

<https://informatik.rub.de/studium/studiengaenge/ai/msc/>

Sommersemester 2023

Stand: 06.06.2023



Wo finde ich Hilfe während des Studium?

Wichtige Informationsquellen rund um das Studium der Angewandten Informatik:

Fakultätswebseite:
<https://informatik.rub.de/>

Studiengangswabseite:
<https://informatik.rub.de/studium/studiengaenge/ai/>

Studienfachberatung der Fakultät für Informatik:
<https://informatik.rub.de/studium/studienberatung/>

Prüfungsamt der Fakultät für Informatik:
<https://informatik.rub.de/studium/pruefungsamt/>

Fachschaftsrat Angewandte Informatik:
<https://blog.ai-rub.de/>

Bei fachlichen Fragen besteht die Möglichkeit die Dozenten während Ihrer Sprechstunden (siehe individuelle Webseiten) zu kontaktieren.

Weitere wichtige Kontaktadressen auf dem Campus sind:

Zentrale Studienberatung:
<https://studium.ruhr-uni-bochum.de/de/zentrale-studienberatung>
Bietet Hilfe und Coaching bei individuellen Problemen (auch psychologische Betreuung).

Studienfinanzierungsberatung:
<https://studium.ruhr-uni-bochum.de/de/studienfinanzierung>

Stipendienberatung der RUB:
<https://studium.ruhr-uni-bochum.de/de/stipendienberatung>
Beratung zu Stipendien für Studieninteressierte und Studierende.

Beratungszentrum zur Inklusion Behinderter und chronisch Kranker:
<https://www.akafoe.de/inklusion/>

International Office:
<https://international.ruhr-uni-bochum.de/de/auslandsaufenthalte-im-studium>
Beratung zu Studienaufenthalten im Ausland

Wohnheimplätze:
<https://www.akafoe.de/wohnen/>

Studiengangsziele:

Im Masterstudiengang werden gemäß DQR Niveaustufe 7 Kompetenzen im Bereich der Angewandten Informatik vermittelt, die zur Bearbeitung von neuen komplexen Aufgaben- und Problemstellungen sowie zur eigenverantwortlichen Steuerung von Prozessen benötigt werden. Absolventinnen und Absolventen verfügen über umfassendes, detailliertes und spezialisiertes Wissen auf dem neuesten Erkenntnisstand in verschiedenen Anwendungsbereichen der Informatik. Sie nutzen spezialisierte fachliche und konzeptionelle Fertigkeiten zur Lösung von Problemen und können darüber hinaus auch neue Verfahren unter Berücksichtigung gegebener Rahmenbedingungen entwickeln. Fehlendes Wissen erschließen sie sich selbstständig. In Gruppenprojekten tragen sie verantwortlich zur Lösung komplexer Aufgabenstellungen bei und vertreten die Ergebnisse vor anderen. Darüber hinaus sind Absolventinnen und Absolventen des Masterstudienganges in der Lage fachspezifische Diskussionen auch in englischer Sprache zu führen. Der Umgang mit englischer Fachliteratur ist eine Selbstverständlichkeit. Je nach Wahl der freien Wahlmodule können noch weitere fachübergreifende Kompetenzen erlangt werden.

Modularisierungskonzept:

Das Studium ist modular aufgebaut. Die Module stellen zeitlich und inhaltlich abgeschlossene Teilqualifikationen dar. Sie haben, abgesehen von wenigen Ausnahmen, einen Workload von mindestens 5 Creditpoints (CP), wobei ein Creditpoint in etwa einer Arbeitsbelastung von 30 Stunden entspricht. In die Berechnung des Workloads fließt neben der Präsenzzeit auch die Zeit für das Selbststudium mit ein (Bearbeitung von Übungsaufgaben, Vor- und Nachbereitung der Vorlesungen, Lesen geeigneter Literatur, ...). Im Masterstudium gibt es keine Pflichtmodule. Die im Wahlpflichtbereich angesiedelten Module erweitern das Grundlagenwissen aus dem Bachelorstudium, um den spezialisierteren Anwendungsmodulen folgen zu können. Insgesamt bietet das Studium zahlreiche Wahlmöglichkeiten für die Studierenden, um ein individuelles Studienprofil zu entwickeln.

Prüfungsformen:

Prüfungsleistungen können in Form einer Klausur (auch in elektronischer Form), einer mündlichen Prüfung, eines Seminarbeitrags, eines Referates oder Präsentation, einer Hausarbeit, einer Projektarbeit, einer praktischen Prüfung oder eines Kolloquiumsvortrags erbracht werden. Auch Kombinationen von verschiedenen Prüfungsformen sind möglich.

Studienplan Master Angewandte Informatik Ruhr-Universität Bochum

| Nr | Modul | Umfang (LP) | Empfohlenes Semester | Bewertung |
|---------------------------|--------------------------------------|-------------|----------------------|-----------|
| Wahlpflichtbereich | | | | |
| 1 | Wahlpflichtmodule** | 20* | 1-3 | benotet |
| Anwendungsbereich | | | | |
| 2 | Anwendungsmodule** | 35* | 1-3 | benotet |
| 3 | Fachwissenschaftliche Vertiefung**** | 6 | 1-3 | benotet |
| Wahlbereich | | | | |
| 4 | Freie Wahlmodule*** | 15 | 1-3 | unbenotet |
| Praktische Fächer | | | | |
| 5 | Studienprojekt | 10 | 2-3 | benotet |
| Abschlussarbeit | | | | |
| 6 | Masterarbeit und Kolloquium | 30 | 4 | benotet |
| Summe: | | 120 | | |

* Der Gesamtumfang der Wahlpflichtmodule und Anwendungsmodulen darf 59 CP in Summe nicht unterschreiten.

** Informationen zu den wählbaren Wahlpflicht- und Anwendungsmodulen befinden sich im jeweils aktuellen Modulhandbuch. Informationen zu den angebotenen Seminaren finden Sie im Vorlesungsverzeichnis der RUB.

*** Hier können (nahezu) alle Veranstaltungen des Vorlesungsverzeichnisses der RUB, sowie Veranstaltungen im Rahmen der Universitätsallianz Ruhr gewählt werden.

Angebote Wahlpflicht- und Anwendungsmodule

| Lehrveranstaltung | Einheit | Umfang Modul (LP) | Semester | Bewertung |
|--|------------|-------------------|--|-----------|
| Wahlpflichtmodule | | | | |
| Effiziente Algorithmen | Mathe | 9 | SS | benotet |
| Geometrische Algorithmen | Informatik | 5 | SS | benotet |
| Fundamentals of GPU Programming | ETIT | 5 | WS | benotet |
| Machine Learning: Evolutionary Algorithms | Informatik | 6 | WS | benotet |
| Machine Learning: Supervised Methods | Informatik | 6 | SS | benotet |
| Machine Learning: Unsupervised Methods | Informatik | 9 | WS | benotet |
| Kryptographie | Informatik | 8 | WS | benotet |
| Komplexitätstheorie | Informatik | 9 | SS (entfällt im SS 23) | benotet |
| Deterministic Network Calculus | Informatik | 5 | SS | benotet |
| Theorie des maschinellen Lernens | Informatik | 9 | SS | benotet |
| Anwendungsmodule | | | | |
| Ingenieurinformatik | | | | |
| Design Optimization | Bauing | 6 | WS | benotet |
| Advanced System Engineering | MB | 5 | WS | benotet |
| Verkehrstechnik | Bauing | 6 | SS | benotet |
| Grundlagen der FEM | Bauing | 5 | WS (ab WS 23/24) | benotet |
| Bildverarbeitung in der Medizin | ETIT | 5 | SS | benotet |
| Programmier- und Simulationstechnik | | | | |
| Simulationstechnik | Bauing | 5 | WS | benotet |
| 3D-Simulation in der Automatisierungstechnik | MB | 5 | WS | benotet |
| High-Performance Computing on Clusters | Bauing | 6 | WS | benotet |
| High-Performance Computing on Multi- and Manycore Processors | Bauing | 6 | SS | benotet |
| Knowledge Graphs | Informatik | 5 | SS | benotet |
| Neuroinformatik | | | | |
| Autonomous Robotics: Action, Perception and Cognition | Informatik | 6 | SS | benotet |
| Computational Neuroscience: Neural Dynamics | Informatik | 6 | WS | benotet |
| Computational Neuroscience: Vision and Memory | Informatik | 5 | SS (entfällt im SS 23) | benotet |
| Computational Neuroscience: Single Neuron Models | Informatik | 5 | SS | benotet |
| Master-Praktikum: Autonomous Robotics | Informatik | 3 | WS/SS | benotet |
| Deep Learning | Informatik | 5 | WS | benotet |
| Autonomous Vehicles and Artificial Intelligence | Informatik | 5 | SS/WS | benotet |
| Master-Praktikum: Deep Learning and Natural Language Processing | Informatik | 5 | Unregelmäßig (Angebot im SS 23) | benotet |
| Kryptologie und theoretische Informatik | | | | |
| Symmetrische Kryptanalyse | Informatik | 5 | WS | benotet |
| Kryptographische Protokolle | Informatik | 5 | SS | benotet |
| Systemsicherheit | Informatik | 5 | SS | benotet |
| Model Checking | Informatik | 5 | SS | benotet |
| Quantum Information and Computation | Informatik | 5 | WS | benotet |
| Information Theory | Informatik | 5 | SS | benotet |
| IT in Wirtschaft , Management und Gesellschaft | | | | |
| Product Lifecycle Management | MB | 5 | SS | benotet |
| KIbox: Lösung von Nachhaltigkeits-Challenges durch den Einsatz künstlicher Intelligenz | AW + MB | 5 | WS/SS (entfällt im SS 23) | benotet |
| Complexity Economics and Agent-based Modeling | WiWi | 10 | Unregelmäßig (voraussichtlich WS 23/24) | benotet |
| Inklusives IT-Design | AW | 6 | SS | benotet |
| Computerlinguistik | | | | |

| | | | | |
|--|------------|---------------|---------------------|---------|
| Anwendungen der Computerlinguistik (Im SS 23: Natural Language Processing for Meaning Analyses (050044)) | Philologie | 10 bzw. 12 | SS/WS | benotet |
| Introduction to CL in Python | Philologie | 5 | WS | benotet |
| Introduction to Linguistic Data Science with R | Philologie | 5 | WS | benotet |
| Computational Linguistics and AI | Philologie | 5 | SS | benotet |
| Linguistic Data Science | Philologie | 5 | SS | benotet |
| Schwerpunktseminar Computational Linguistics (Im SS 23: 050831 Data Wrangling with Pandas) | Philologie | 5 | WS/SS (ab SS 23) | benotet |
| Schwerpunktseminar Linguistic Data Science (Im SS 23: 050833 Visualizing Linguistic Data in R) | Philologie | 5 | WS/SS (ab SS 23) | benotet |
| Bioinformatik | | | | |
| Bioinformatics for Proteomics | MPC | 5 | WS | benotet |
| Bioimage Informatics | Biologie | 5 | WS | benotet |
| Master-Praktikum: Big Data in der Bioinformatik | Biologie | 5 | WS/SS | benotet |
| Master-Praktikum: Computational Proteomics | MPC | 5 | SS | benotet |

Angebote Vertiefungsseminare im Modul „Fachwissenschaftliche Vertiefung“

| Lehrveranstaltung | Einheit | Umfang Modul (LP) | Semester | Bewertung |
|--|-------------------|-------------------------|------------------|-----------|
| Vertiefungsseminare | | | | |
| Seminar Ingenieurinformatik | Baulng | 3 | SS und WS | benotet |
| Seminar Bioinformatik | Biologie / MPC | 3 | SS und WS | benotet |
| Seminar Symmetrische Kryptanalyse | Informatik | 3 | SS und WS | benotet |
| Seminar Ressourceneffiziente Systemsoftware | Informatik | 3 | SS und WS | benotet |
| Seminar on Knowledge Graphs | Informatik | 3 | WS | benotet |
| Seminar Topics in Deep Learning for Sequence Processing | Informatik | 3 | WS | benotet |
| Seminar Approximationsalgorithmen | Informatik | 3 | SS | benotet |
| Seminar Satisfiability | Informatik | 3 | SS | benotet |
| Seminar Computational Neuroscience | Informatik | 3 | SS | benotet |
| Seminar From Biological to Artificial Neural Networks | Informatik | 3 | SS | benotet |
| Seminar Implementation Security | Informatik | 3 | SS | benotet |
| Seminar Quantum Algorithms | Informatik | 3 | SS | benotet |
| Seminar Advanced topics in Deep Learning | Informatik | 3 | SS | benotet |
| Seminar Fortgeschrittene Themen des Model Checking | Informatik | 3 | SS (nicht SS 23) | benotet |
| Seminar über Grenzen in der theoretischen Informatik | Informatik | 3 | unregelmäßig | benotet |

Abkürzungen:AW: Institut für Arbeitswissenschaften
Baulng: Fakultät für Bau- und Umweltingenieurwissenschaften
ETIT: Fakultät für Elektrotechnik und Informationstechnik
MB: Fakultät für Maschinenbau
MPC: Medizinisches Proteom Center
WIWI: Fakultät für Wirtschaftswissenschaft

SS: Sommersemester
WS: Wintersemester

CP: Creditpoints

MODULHANDBUCH

Übersicht der Module

Angewandte Informatik - Master (1-Fach, PO 2020)

Wahlpflichtbereich

3D-Simulation in der Automatisierungstechnik
Advanced System Engineering
Anwendungen der Computerlinguistik [Master]
Autonomous Robotics: Action, Perception, Cognition
Autonomous Vehicles and Artificial Intelligence
Bildverarbeitung in der Medizin
Bioimage Informatics
Bioinformatics for Proteomics
Complexity Economics and Agent-based Modeling
Computational Linguistics and AI
Computational Neuroscience: Neural Dynamics
Computational Neuroscience: Single-Neuron Models
Computational Neuroscience: Vision and Memory
Deep Learning
Design Optimization
High-Performance Computing on Clusters
High-Performance Computing on Multi- and Manycore Processors
Information Theory
Inklusives IT-Design
Introduction to Computational Linguistics in Python
Introduction to Linguistic Data Science with R
Klbox: Lösung von Nachhaltigkeits-Challenges durch den Einsatz künstlicher Intelligenz
Knowledge Graphs
Kryptographische Protokolle
Linguistic Data Science
Master-Praktikum Deep Learning and Natural Language Processing
Master-Praktikum: Autonomous Robotics
Master-Praktikum: Big Data in der Bioinformatik
Master-Praktikum: Computational Proteomics
Model Checking
Product Lifecycle Management
Quantum Information and Computation
Schwerpunktseminar Computational Linguistics
Schwerpunktseminar Linguistic Data Science

Simulationstechnik
Symmetrische Kryptanalyse
Systemsicherheit
Verkehrstechnik
Deterministic Network Calculus
Effiziente Algorithmen
Fundamentals of GPU Programming
Geometrische Algorithmen
Komplexitätstheorie
Kryptographie
Machine Learning: Evolutionary Algorithms
Machine Learning: Supervised Methods
Machine Learning: Unsupervised Methods
Parallel Computing
Theorie des maschinellen Lernens
Fachwissenschaftliche Vertiefung (Angewandte Informatik)
Studienprojekt (Master)

Wahlbereich

Freie Wahlmodule

Masterarbeit

Masterarbeit und Kolloquium (AI)

| Titel des Moduls: 3D-Simulation in der Automatisierungstechnik | | | | | |
|---|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen 3D-Simulation in der Automatisierungstechnik (139050) | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße Studierende |
| Unterrichtssprache Deutsch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr.-Ing. Bernd Kuhlenkötter Lehrende: Dr.-Ing. Alfred Hypki | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik MSc. Maschinenbau MSc. Sales Engineering and Product Management | | | | | |
| Lernziele (learning outcomes) Nach einer allgemeinen Einführung in die Anforderungen und Möglichkeiten der 3D Simulation in der Automatisierungstechnik erlernen die Studierenden die verschiedenen Schritte, die sich von der Idee über die Modellierung der Arbeitszelle und Programmierung der Automatisierungskomponenten bis zur Virtuellen Inbetriebnahme erstrecken. Ein besonderes Augenmerk wird in dieser Vorlesung auf die Industrielle Robotik gelegt, die in zahlreichen Beispielen und Anwendungen thematisiert wird. Nach dem erfolgreichen Abschluss des Moduls <ul style="list-style-type: none"> • beherrschen die Studierenden die Grundlagen der 3D-Simulationstechnik. • haben die Studierenden Fähigkeit erworben, automatisierungstechnische Aufgabenstellungen vorab über eine 3D-Simulation abzubilden und abzusichern. • kennen die Studierenden die wichtigsten Methoden und Softwaresysteme zur Lösung simulationstechnischer Probleme. | | | | | |
| Inhalt Die Vorlesung deckt die folgenden Themenbereiche ab <ul style="list-style-type: none"> • Simulation in der Automatisierungstechnik: Anforderungen und Möglichkeiten • Grafische 3D-Simulation • CAD-basierte Arbeitszellenmodellierung und 3D-Datenaustausch • Roboterprogrammierung • Offline-Programmierung und Virtuelle Inbetriebnahme • Grundlagen und Leistungsmerkmale von grafischen 3D-Simulationssystemen im industriellen Einsatz <p>Im Bereich der Kompetenzen im Kontext Digitalisierung ist besonders die Übung zu nennen, diese besteht aus der praktischen Umsetzung der von Modellierungs-, Programmierungs- und Simulationsaufgaben mit einem kommerziell verfügbaren und industriell eingesetzten 3D- Robotersimulations- und Offline-Programmiersystem.</p> | | | | | |
| Lehrformen Gemeinsame Vorlesung und Übung in kleineren Gruppen | | | | | |
| Prüfungsformen Klausur '3D-Simulation in der Automatisierungstechnik' (90 Min., Anteil der Modulnote 100 %) Studienbegleitende Aufgaben: Umsetzung einer Simulationsaufgabe (Sofern die Aufgabe vor der Modulabschlussprüfung absolviert wird, sind optional Bonuspunkte für die Klausur möglich) (Umfang und Abgabefristen wird am Anfang des Semesters bekanntgegeben) | | | | | |

Voraussetzungen für die Vergabe von Credits

Bestandene Modulabschlussprüfung: Klausur

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

5/105: M.Sc. Angewandte Informatik

Titel des Moduls: Advanced System Engineering
Advanced System Engineering

| | | | | | |
|---|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen Advanced System Engineering | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße Studierende |
| Unterrichtssprache Deutsch | | | Teilnahmevoraussetzungen | | |

Modulbeauftragte/r und hauptamtlich Lehrende

Modulbeauftragte/r: Prof. Dr.-Ing. Detlef Gerhard
 Lehrende: Prof. Dr.-Ing. Detlef Gerhard

Verwendung des Moduls

M.Sc. Angewandte Informatik

M.Sc. Sales Engineering and Product Management
 M.Sc. Maschinenbau

Lernziele (learning outcomes)

Nach dem erfolgreichen Abschluss des Moduls

- verfügen Studierende über ein breites, integriertes Wissen über die über Methoden und Werkzeuge des Advanced Systems Engineering
- haben Studierende ein umfassendes Verständnis vom Zusammenwirken der relevanten Softwaresysteme und von Integrationskonzepten aus Sicht des Advanced Systems Engineerings erworben und können dieses situativ angepasst anwenden, um z.B. differenziert die CAx und PDM/ PLM Prozessketten zu beurteilen.
- sind die Studierenden in der Lage Problemstellungen im Rahmen der Entwicklung komplexer technischer Systeme eigenständig zu analysieren, zu bewerten und eine Systemarchitektur auf Basis von Anforderungsmanagement zu erarbeiten.
- verfügen Studierende über exemplarische, interdisziplinäre Methodenkompetenz der domänenübergreifenden Modellierung und können komplexe technische Systeme selbstständig in einer objektorientierten Modellierungssprache (z.B. SysML) modellieren, verstehen und die erarbeiteten Modelle bewerten
- können kooperativ Aufgabenstellungen der Modellierung und Simulation im Umfeld von Advanced Systems Engineering in heterogenen Gruppen bearbeiten, Abläufe und Ergebnisse begründen und über Sachverhalte in Verbindung mit dem Aufbau von IoT-Lösungen und Digitalen Zwillingen kommunizieren. Indem sie praktische Fallbeispiele und Aufgaben bearbeiten, können sie die erlernten Fertigkeiten im Umgang mit entsprechenden Softwaresystemen auf konkrete Problemstellungen der des Modellbasierten Systems Engineering übertragen.

Inhalt

Die Digitalisierung eröffnet zahlreiche Nutzenpotenziale für neuartige Marktleistungen der Industrie auf Basis von technischen Systemen, die durch hohe Komplexität und Interdisziplinarität gekennzeichnet sind. Ein Ansatz, die Grenzen klassischer Entwicklungsmethoden zu überwinden, ist das Advanced Systems Engineering, das auf das modellbasierte und integrative Zusammenwirken der Disziplinen und Aspekte der Produktentstehung in einem Entwicklungsansatz fokussiert und somit eine ganzheitliche Produktentstehungsmethodik unter Einbeziehung systemtechnischer Methoden darstellt. Die Vorlesung behandelt aus einer übergeordneten Perspektive des Gesamtprozesses die Methoden und Werkzeuge, die bei der Entwicklung von komplexen technischen Systemen Anwendung finden. Dabei wird insbesondere das Modell Based Systems Engineering (MBSE) Konzept sowie die Systemmodellierungssprache SysML angewendet und von den Studierenden selbst Projektaufgaben im Umfeld von IoT Anwendungen und dem Aufbau von Digitalen Zwillingen erarbeitet.

Folgende Themen sind Gegenstand des Moduls

- Einführung Grundlagen und Begriffe des Advanced Systems Engineering
 - Model Based Definition und Model Based Systems Engineering
 - Systemdefinition und formale Modellierung, Modellierungssprachen, z.B. Sys-ML
 - Anforderungsmanagement
 - Modellierung und Simulation zusätzlich zur mechanischen Struktur
 - Domänenübergreifende Aspekte von Steuerungstechnik/Elektrotechnik
 - Spezialisierte Modellierungsmethoden für Logik- und Verhalten
 - Lückenlose Werkzeugketten und Durchgängigkeit der Modelle
 - Integration aller Domänen in PLM (Mechanik + E/E + Software/Steuerungstechnik)
 - Simulation, Verifikation und Validierung
- Methoden für die Planung, Überwachung und Durchführung von Projekten im Hinblick auf Technologie, Zeit und Kosten

Lehrformen

Übung (2 SWS) / Vorlesung (2 SWS)

Prüfungsformen

Klausur 'Advanced Systems Engineering' (90 Min., Anteil der Modulnote 100 %)

Studienbegleitende Aufgaben: Hausarbeiten (Sofern die Hausarbeiten vor der Modulabschlussprüfung absolviert werden, sind optional Bonuspunkte für die Klausur möglich) (Umfang und Abgabefristen wird am Anfang des Semesters bekanntgegeben)

(Beide Prüfungsteile sind über Flexnow anzumelden)

Voraussetzungen für die Vergabe von Credits

- Bestandene Modulabschlussprüfung: Klausur
- Bestandene studienbegleitende Aufgaben: Hausarbeiten (Umfang und Abgabefristen wird am Anfang des Semesters bekanntgegeben)

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

5/150: M.Sc. Angewandte Informatik

Titel des Moduls: Anwendungen der Computerlinguistik [Master]

| | | | | | |
|---|-------------------------|-----------------------------------|---------------------------------|--|------------------------------------|
| Modul-Nr./Code | Credits 10 CP | Workload 300 bzw. 360 h | Semester | Turnus Jedes Semester | Dauer 1 Semester |
| Lehrveranstaltungen Wechselndes Seminarangebot (siehe VVZ) | | | Kontaktzeit 30 h | Selbststudium 270 bzw. 330 h | Gruppengröße Studierende |
| Unterrichtssprache Deutsch oder Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Stefanie Dipper Lehrende: Prof. Dr. Stefanie Dipper, Prof. Dr. Ralf Klabunde, u.a. | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) 10 CP-Variante <ul style="list-style-type: none">Nach dem erfolgreichen Abschluss des Moduls verfügen Sie über vertieftes Wissen und Fertigkeiten in weiteren von Ihnen ausgewählten Teilgebieten der Computerlinguistik.Sie haben einen Überblick über mehrere Bereiche computerlinguistischer Forschung. Sie kennen jeweils verschiedene Ansätze und Methoden, die in diesen Bereichen Anwendung finden, und können diese miteinander vergleichen und nach ihren Stärken und Schwächen bewerten.Sie können die passende Methode für eine Aufgabe auswählen und sie implementieren und evaluieren. Zudem können Sie zu Ihrer Implementation eine entsprechende wissenschaftliche Publikation verfassen. 12 CP-Variante: <ul style="list-style-type: none">Nach dem erfolgreichen Abschluss des Moduls verfügen Sie über fundiertes Wissen zum aktuellen Forschungsstand in weiteren von Ihnen ausgewählten Teilgebieten der Computerlinguistik.Sie haben einen Überblick über mehrere Bereiche computerlinguistischer Forschung und kennen die jeweiligen Ansätze und Methoden. Basierend auf diesen Kenntnissen können Sie einen eigenen Ansatz entwickeln und diesen in einer Implementation umsetzen sowie evaluieren.Sie können zu ihrem eigenen Ansatz eine Publikation in Form eines wissenschaftlichen Aufsatzes verfassen. | | | | | |
| Inhalt 10 CP-Variante: In den im Modul angebotenen Seminaren werden beispielhaft Themen aus dem gesamten Spektrum der Computerlinguistik abgedeckt, wie z. B. automatische Textzusammenfassung, Koreferenzauflösung, Dialogmodellierung, Metaphernanalyse. In jedem Themenbereich werden Sie eigenständig mit computerlinguistischer Fachliteratur arbeiten, diese einordnen und bewerten und ausgewählte Ansätze nachimplementieren. 12 CP-Variante: In den im Modul angebotenen Seminaren werden beispielhaft Themen aus dem gesamten Spektrum der Computerlinguistik abgedeckt, wie z. B. automatische Textzusammenfassung, Koreferenzauflösung, Dialogmodellierung, Metaphernanalyse. In jedem Teilgebiet werden Sie eigenständig mit computerlinguistischer Fachliteratur arbeiten und eigene Ansätze entwickeln und nachimplementieren. | | | | | |
| Lehrformen Seminaristischer Unterricht mit Gruppenarbeiten, mündlichen Präsentationen von Forschungsarbeiten. Die Betreuung bei Selbststudiumanteilen von 150h+ pro Lehrveranstaltung erfolgt | | | | | |

nach individueller Absprache. Im Regelfall erfolgen individuelle Sitzungen (Zoom oder Präsenz) im wöchentlichen oder 14-tägigen Wechsel.

Prüfungsformen

Implementation und Research Thesis (semesterbegleitend)

Voraussetzungen für die Vergabe von Credits

Erfolgreicher Abschluss der Modulprüfung (7,0 bzw. 9 CP) sowie bestandene Studienleistung (3,0 CP) im Seminar.

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

10/105 bzw. 12/105: M.Sc. Angewandte Informatik

| | | | | | |
|--|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Titel des Moduls: Autonomous Robotics: Action, Perception, Cognition | | | | | |
| Modul-Nr./Code | Credits 6 CP | Workload 180 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Autonomous Robotics: Action, Perception, Cognition (211048) | | | Kontaktzeit 45 h | Selbststudium 135 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Gregor Schöner Lehrende: Prof. Dr. Gregor Schöner | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik M.Sc. Informatik | | | | | |
| Lernziele (learning outcomes) After the successful completion of this course the students: <ul style="list-style-type: none"> • are familiar with the concepts in dynamical systems theory and can use them practically, • know the mathematical models of movement generation • have practice in reading and writing academic research papers. | | | | | |
| Inhalt Autonomous robotics is an interdisciplinary research field in which embodied systems equipped with their own sensors and with actuators generate behavior that is not completely preprogrammed. Autonomous robotics thus entails perception, movement generation, as well as core elements of cognition such as making decisions, planning, and integrating multiple constraints. This course touches on various approaches to this interdisciplinary problem. In the first half of the course, the main emphasis will be on dynamical systems methods for generating movement in vehicles. The main focus of the course is, however, on solutions to autonomous movement generation that are inspired by analogies with how nervous systems generate movement. In fact, the second half of the course will review core problems in human movement science, including the degree of freedom problem, coordination, motor control, and the reflex control of muscles. | | | | | |
| Lehrformen Lecture with Exercise | | | | | |
| Prüfungsformen Oral Examination; Exercises for bonus-points | | | | | |
| Voraussetzungen für die Vergabe von Credits Passed oral examination | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 6/105: M.Sc. Angewandte Informatik | | | | | |

| Titel des Moduls: Autonomous Vehicles and Artificial Intelligence | | | | | |
|---|------------------------|--------------------------|---|---------------------------------|---------------------------------------|
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester siehe Prüfungsordnung | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Autonomous Vehicles and Artificial Intelligence | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße 25 Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen keine | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Thorsten Berger Lehrende: Prof. Dr. Thorsten Berger, Dr. Sven Peldszus | | | | | |
| Verwendung des Moduls B.Sc. Informatik B.Sc. IT-Sicherheit B.Sc. Angewandte Informatik M.Sc. Informatik M.Sc. Angewandte Informatik M.Sc. IT-Sicherheit/Informationstechnik | | | | | |
| Lernziele (learning outcomes) <ul style="list-style-type: none"> • Understanding requirements on autonomous vehicles • Understanding the architecture of autonomous vehicles • Ability to build a self-driving car with ROS2 • Understanding and applying quality assurance for autonomous vehicles | | | | | |
| Inhalt Autonomous driving is the future of individual mobility and all major manufacturers are working on fully autonomous vehicles. While there are robust and good solutions for the individual problems in autonomous driving, the main challenge lies in their integration. Altogether, an autonomous vehicle's software is the biggest problem. Therefore, the key in self-driving vehicles is about getting the software right. In this course, we will investigate the different aspects of self-driving vehicles as well as the importance and application of artificial intelligence in this domain. The course will primarily focus on the following topics: <ul style="list-style-type: none"> • Requirements on autonomous vehicles • Architecture of autonomous vehicles • Operation systems and frameworks for robotic systems • Specification and Implementation of autonomous vehicles based on ROS2 • Artificial intelligence for autonomous vehicles • Simulation of autonomous vehicles &#8729; Localization and perception • Mission planning • Quality assurance for autonomous vehicles In the course's lecture, we provide the required theoretical background and practically apply the course's content in exercises by building a self-driving robot. | | | | | |
| Lehrformen Vorlesung mit Übungen | | | | | |
| Prüfungsformen Mündliche Prüfung | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestandene Modulabschlussprüfung und erfolgreiche Teilnahme an den Übungen | | | | | |

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

5/158: B.Sc. Informatik [PO 22]

5/165: B.Sc. Informatik [PO 22]

5/150: B.Sc. IT-Sicherheit/Informationstechnik [PO 22]

5/149: B.Sc. IT-Sicherheit/Informationstechnik [PO 20]

5/168: B.Sc. Angewandte Informatik [PO 22]

5/170: B.Sc. Angewandte Informatik [PO 20]

5/ 97: M.Sc. Informatik

5/105: M.Sc. Angewandte Informatik

5/91: M.Sc. IT-Sicherheit/Informationstechnik [PO 22]

5/84: M.Sc. IT-Sicherheit/Informationstechnik [PO 20]

| Titel des Moduls: Bildverarbeitung in der Medizin | | | | | |
|--|------------------------|--------------------------|--|---------------------------------|---------------------------------------|
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Bildverarbeitung in der Medizin (141220) | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße 20 Studierende |
| Unterrichtssprache Deutsch | | | Teilnahmevoraussetzungen Eine Teilnahme ist ausgeschlossen, wenn im Bachelor bereits das Modul Computersehen: Einführung (Digitale Bildverarbeitung) absolviert wurde. | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr.-Ing. Georg Schmitz Lehrende: Dr.-Ing. Stefanie Dencks | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) Nach erfolgreichem Abschluss des Moduls verfügen die Studierenden über Kenntnisse der mehrdimensionalen digitalen Signalverarbeitung. Sie kennen und verstehen die Aufnahme mehrdimensionaler Bilddaten der wichtigsten diagnostischen Abbildungsverfahren, können diese modellieren und hieraus Konsequenzen für ihre Verarbeitung ableiten. Die Studierenden können die verschiedenen Schritte der Bildverarbeitung in abstrakte Aufgabenkategorien einordnen (z.B. Filterung, Segmentierung, Klassifikation) und kennen ausgewählte Verfahren im Detail und können diese erläutern und anwenden. Die Studierenden sind in der Lage, eine gegebene Bildverarbeitungsaufgabe zu analysieren, sowie einen geeigneten Lösungsweg zu entwickeln und algorithmisch umzusetzen. Die Verfahren werden am Beispiel medizinischer Bilddaten vermittelt, die Studierenden können die Verfahren aber auch auf andere Anwendungsgebiete übertragen. Durch die Übungen in Kleingruppen, teilweise an Rechnern, sind die Studierenden befähigt, das Erlernte im Team praktisch umzusetzen, Lösungsansätze zu erläutern und argumentativ zu vertreten. | | | | | |
| Inhalt Es werden die Grundlagen und speziellen Verfahren der Bildverarbeitung vorgestellt, die insbesondere bei medizinischen Bilddaten Anwendung finden. Viele Verfahren werden jedoch auch in anderen Anwendungsfeldern wie z.B. der industriellen Bildverarbeitung eingesetzt. Im ersten Abschnitt werden sowohl die Rezeption durch das menschliche visuelle System behandelt, als auch Definitionen und Grundlagen für die Bildverarbeitung eingeführt (z.B. Diskretisierung, Abtasttheorem, globale Kenngrößen von Bildern). Der zweite Abschnitt vermittelt die wichtigsten Operationen im Ortsbereich (Histogrammmodulation, Filterung morphologische Operationen, geometrische Bildoperationen, distance transform, ...). Der dritte Abschnitt umfasst Methoden der Informationsextraktion (Segmentierung, Texturanalyse, Formbeschreibung). Im vierten Abschnitt liegt der Schwerpunkt auf der Klassifikation und verschiedenen Verfahren des Machine Learning (z.B. support vector machines, deep learning). Der fünfte Abschnitt beinhaltet die Bildrestauration. Zusätzlich wird ein Überblick über die Bildregistrierung und 3D-Visualisierung gegeben. | | | | | |
| Lehrformen Vorlesung mit begleitender Übung. Diese Lehrveranstaltung wird über Moodle organisiert. Die notwendigen Informationen werden in der ersten Vorlesung mitgeteilt. | | | | | |
| Prüfungsformen Mündliche Modulabschlussprüfung (30 Minuten) | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestandene mündliche Modulabschlussprüfung | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 5/105: M.Sc. Angewandte Informatik | | | | | |

| | | | | | |
|--|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Titel des Moduls: Bioimage Informatics | | | | | |
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße Studierende |
| Unterrichtssprache Deutsch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Axel Mosig Lehrende: Prof. Dr. Axel Mosig | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) Die Studierenden sollen elementare und aktuelle Techniken der Bildverarbeitung kennenlernen, die zur Analyse mikroskopischer Bilddaten verwendet werden. Sie sollen die dahinterliegenden algorithmischen, mathematischen und statistischen Ideen verstehen, und lernen in der Praxis, wie diese Methoden auf reale Daten und Fragestellungen angewendet werden. | | | | | |
| Inhalt Die Analyse von mikroskopischen Bilddaten mit Methoden der Bildverarbeitung ist in den vergangenen Jahren ein wichtiges Thema in vielen Anwendungen in den Lebenswissenschaften geworden. In der Vorlesung werden grundlegende Konzepte der Verarbeitung mikroskopischer Bilddaten und deren Anwendungen behandelt. Die Vorlesung gliedert sich wie folgt: <ul style="list-style-type: none"> • Grundlagen der Mikroskopie: Durchlicht- und Konfokalmikroskopie; Fluoreszenzmikroskopie; Mikroskopie in der Histopathologie; Mikroskopie in der Zellbiologie; Mikroskopie in der Neurobiologie • Morphologische Bildanalyse • Texturelle Bildanalyse • Kollokalisations-Verfahren • Algorithmen zur Bildregistrierung • Algorithmen zum Verfolgen von Bewegungsmustern von Zellen ("Cell Tracking") • Verfahren zur Rekonstruktion von Neuronen ("Neuron Tracing") • Deep Learning Methoden zur mikroskopischen Bilddatenanalyse • Methoden der Analyse von markerfreien Mikroskopie-Daten • Methoden zur überwachten und unüberwachten Segmentierung von mikroskopischen Bildern | | | | | |
| Lehrformen Vorlesung mit Übung | | | | | |
| Prüfungsformen Mündliche Prüfung | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestandene mündliche Modulabschlussprüfung | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 5/105: M.Sc. Angewandte Informatik | | | | | |

| Titel des Moduls: Bioinformatics for Proteomics | | | | | |
|--|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen Bioinformatics for Proteomics (201911) | | | Kontaktzeit 45 h | Selbststudium 105 h | Gruppengröße Studierende |
| Unterrichtssprache English | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Martin Eisenacher Lehrende: Prof. Dr. Martin Eisenacher | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) After the successful completion of this module <ul style="list-style-type: none"> • the students have become familiar with knowledge of protein biochemistry, • they are able to explain the principles of mass spectrometry as the key technology of proteomics, • they are able to explain the current methods of bioinformatics for proteomics that are used for the analysis of raw data (i.e., mass spectra), • they are able to identify and quantify peptides or proteins, respectively, and to interpret them biologically, • they understand the underlying algorithmic and statistical concepts of these methods, • they are able to use proteomics-specific software, tools, and / or algorithms, • they are able to design and program own solutions, • and they are able to apply the discussed software tools and methods to real data and problems. | | | | | |
| Inhalt <ul style="list-style-type: none"> • Basics of protein biochemistry • Properties of amino acids • Basics of mass spectrometry • Protein databases • Tryptic and in silico digest of proteins • Principles of spectra identification search engines and scores • Quality control • Estimation of the false discovery rate (FDR) using the target-decoy-approach • Protein inference • Single / multiple / parallel reaction monitoring (SRM / MRM / PRM) • Data independent acquisition (DIA) • Protein quantification (label-based and label-free techniques) • Preprocessing of quantitative data • Basic statistics / statistics for the comparison of experimental groups • Artificial Intelligence / Machine learning / Classification (e. g. for biomarker discovery) • Protein overrepresentation / enrichment analysis • Software tools used in bioinformatics for proteomics (Tutorial) • Practical (programming) tasks (Tutorial) | | | | | |
| Lehrformen Lecture: slide-based lecture. Tutorial: Solution of practical exercises using real example data as homework, programming tasks, group work, live-presentation of code and software | | | | | |
| Prüfungsformen Oral examination or written exam | | | | | |

Voraussetzungen für die Vergabe von Credits

Passed oral examination or written exam

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

5/105: M.Sc. Angewandte Informatik

| | | | | | |
|---|-------------------------|--------------------------|---------------------------------|--|------------------------------------|
| Titel des Moduls: Complexity Economics and Agent-based Modeling | | | | | |
| Modul-Nr./Code | Credits 10 CP | Workload 300 h | Semester | Turnus unregelmäßig (in der Regel im WS) | Dauer 1 Semester |
| Lehrveranstaltungen | | | Kontaktzeit 60 h | Selbststudium 240 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Michael Roos Lehrende: Prof. Dr. Michael Roos | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) Das Modul verfolgt das Ziel, Studierende der Angewandten Informatik sowie der Wirtschaftswissenschaft in die Methode der agentenbasierten Modellierung und den dazugehörigen Simulationstechniken einzuführen. Hierbei lernen Studierende wie volkswirtschaftliche und sozialwissenschaftliche Fragestellungen entworfen, implementiert und analysiert werden. Im Rahmen der Seminararbeit sollen die Studierenden ein eigenes Modell implementieren und analysieren. Im Rahmen des Moduls erwerben Studierende folgende Kenntnisse: <ul style="list-style-type: none"> • Grundlegende Kenntnisse der Komplexitätsökonomik und agentenbasierter Modellierung • Überführung von wissenschaftlichen Fragestellungen aus dem Bereich Volkswirtschaftslehre in agentenbasierte Computersimulationen • Praktische Arbeit mit agentenbasierten Modellen, Interpretation von Simulationsergebnissen und mögliche Grenzen des Ansatzes • Grundlegende Kenntnisse, um eigene Modelle zu implementieren und Simulationen selbständig durchzuführen • Wissenschaftliches Schreiben • Die Programmiersprache NetLogo | | | | | |
| Inhalt Komplexitätsökonomik ist eine neue Denkschule im Bereich der Volkswirtschaftslehre, der zunehmend an Relevanz gewinnt. Hierbei wird eine Volkswirtschaft als ein komplexes, adaptives System betrachtet, das sich aus verschiedenen Agenten, z.B. Firmen, Haushalten und Banken, zusammensetzt. Im Vergleich zu bisherigen volkswirtschaftlichen Ansätzen sind die Agenten des Systems nicht (vollständig) rational und das System entwickelt sich aus den Interaktionen zwischen den (heterogenen) Agenten. Komplexitätsökonomik eignet sich sehr zur Untersuchung von sozialen Transformations- und Innovationsprozessen, wie der Digitalisierung. In der Vorlesung wird Komplexitätsökonomik und die Methode der agentenbasierten Computersimulation vorgestellt und gezeigt, wie sie zur Analyse komplexer ökonomischer Systeme angewendet werden können. Im Rahmen der Übung werden diese Kenntnisse vertieft. Zudem wird geübt, wie agentenbasierte Modelle mithilfe NetLogo implementiert und simuliert werden. | | | | | |
| Lehrformen Vorlesung und Übung | | | | | |
| Prüfungsformen Seminararbeit (70%) und Klausur (30%) (benotet) Studienleistung (unbenotet) | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestehen der folgenden Leistungen: 1) Studienleistung (unbenotet) 2) Seminararbeit (70%) und Klausur (30%) (benotet) | | | | | |

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

10/105: M.Sc. Angewandte Informatik

| | | | | | |
|--|------------------------|--------------------------|---|---------------------------------|---------------------------------------|
| Titel des Moduls: Computational Linguistics and AI | | | | | |
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Computational Linguistics and AI (050825) | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße 20 Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen Erfolgreich abgeschlossenes Modul "Introduction to Computational Linguistics in Python". | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Tibor Kiss Lehrende: Mirjam Koch M.Sc. | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) <ul style="list-style-type: none"> • Die Studierenden können auf den Grundlagen der Programmierung und Nutzung von Modulen aufbauen, um Verfahren der „künstlichen Intelligenz“ bzw. des maschinellen Lernens auf linguistische Fragestellungen anzuwenden. • Sie kennen einige Grundlagen der kollaborativen Softwareentwicklung in der Computerlinguistik. • Sie kennen grundlegende Evaluationsmethoden für Programme und können sie anwenden. • Sie können sich kritisch mit ihrer Rolle in der Verarbeitung von Daten auseinandersetzen und wissen, dass sie durch die Wahl von Datentypen/-codierung, Algorithmen und ihre Auswertung der Ergebnisse Bias einfügen. | | | | | |
| Inhalt In diesem Kurs geht es um die Zusammenhänge zwischen Computerlinguistik und Verfahren des maschinellen Lernens bzw. der „KI“. Aufbauend auf den Grundkenntnissen aus „Introduction to Computational Linguistics (with Python“) im Sommersemester werden diese nachvollzogen und im Bezug auf linguistische Fragestellungen diskutiert. Die „Vorlesungen“ bestehen aus einer Vortrags- und Diskussionskomponente. Dabei enthalten die auch hochgeladenen Folien die Inhalte, die für die Klausur relevant sind, aber keine ausführlichen Erklärungen. Die Einordnung neuer Verfahren, Methoden und Ideen findet gemeinsam im Plenum statt. Die begleitenden Übungen zur Vorlesung sind nicht verpflichtend, aber sind zur Selbstkontrolle offen für alle Studierenden. In den Übungen und Übungsaufgaben liegt der Fokus eher auf der praktischen Anwendung und Implementierung. | | | | | |
| Lehrformen Die Veranstaltung setzt sich aus einer Präsenzkomponekte (Fragen und Antworten zu den Folien, Diskussionen, praktische Übungen und Überlegungen zum aktuellen Thema) und Übungen zum Kurs zusammen. Die Studierenden müssen sich eigenständig auf die Sitzungen vorbereiten und entscheiden, ob die Teilnahme ihr Verständnis fördert. | | | | | |
| Prüfungsformen Schriftliche Modulabschluss | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestandene schriftliche Modulabschlussprüfung | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 5/105: M.Sc. Angewandte Informatik | | | | | |

| | | | | | |
|--|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Titel des Moduls: Computational Neuroscience: Neural Dynamics | | | | | |
| Modul-Nr./Code | Credits 6 CP | Workload 180 h | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen Computational Neuroscience: Neural Dynamics (212005) | | | Kontaktzeit 45 h | Selbststudium 135 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Gregor Schöner Lehrende: Prof. Dr. Gregor Schöner | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik M.Sc. Informatik | | | | | |
| Lernziele (learning outcomes) <ul style="list-style-type: none"> • Gain experience in interdisciplinarity bridging computer science and cognitive science. • Learn the concepts and methods of nonlinear dynamical systems in a concrete applied context. • Improve familiarity with methods of quantitative natural science, including measurement, graphing observables as a function of experimental control parameters and using models to interpret data. • Read scientific literature. | | | | | |
| Inhalt This course provides an introduction into the theoretical cognitive and functional neurosciences from a particular theoretical vantage point, the dynamical systems approach. This approach emphasizes the evolution in time of behavioral and neural patterns as the basis of their analysis and synthesis. Dynamic stability, a concept shared with the classical biological cybernetics framework, is one cornerstone of the approach. Instabilities (or bifurcations) extend this framework and provide a basis for understanding flexibility, task specific adjustment, adaptation, and learning. The course includes tutorial modules that provide mathematical foundations. Theoretical concepts are expounded in reference to a number of experimental model systems which include the coordination of movement, postural stability, the perception of motion, and elementary forms of embodied cognition. In the spirit of Braitenberg's "synthetic psychology", autonomous robots are used to illustrate some of the ideas. Exercises are integrated into the lectures. They consist of elementary mathematical exercises, the design of (thought) experiments and their analysis, and the design of simple artificial systems, all on the basis of the theoretical framework exposed in the main lectures. One exercise takes the form of an essay for which participants read a scientific paper and answer questions in a longer illustrated text. | | | | | |
| Lehrformen Lecture with Exercise | | | | | |
| Prüfungsformen Oral or written exam, Bonus points for the final examination can be achieved by submitting homework and an essay (10 pages) | | | | | |
| Voraussetzungen für die Vergabe von Credits Passed oral or written Exam | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 6/105: M.Sc. Angewandte Informatik | | | | | |

Titel des Moduls: Computational Neuroscience: Single-Neuron Models

| | | | | | |
|---|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 6 CP | Workload 180 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Computational Neuroscience: Single-Neuron Models | | | Kontaktzeit 60 h | Selbststudium 120 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Robert Schmidt Lehrende: Prof. Dr. Robert Schmidt | | | | | |
| Verwendung des Moduls M.Sc. Informatik M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) Nach Abschluss des Moduls können Studierende <ul style="list-style-type: none">• können Studierende Techniken der Computational Neuroscience zur Simulation neuronaler Aktivität anwenden• haben sich Studierende mit verschiedenen Arten von Modellen einzelner Neuronen, ihrer mathematischen Beschreibung und ihren verschiedenen biologischen Abstraktionsebenen vertraut gemacht• haben Studierende die Fähigkeiten zur Modellierung von Neuronen, Synapsen und Schaltkreisen erworben und können diese Modelle mit Biologie und Berechnung verknüpfen• besitzen Studierende ein Verständnis der biologischen Grundlage für Berechnungen in Neuronen | | | | | |
| Inhalt Dieses Modul beginnt mit einer Einführung in die Neurowissenschaften und die Rolle der Computational Neuroscience. Der nächste Teil des Moduls befasst sich mit biologisch fundierten Modellen einzelner Neuronen, einschließlich Leaky-Integrate-and-Fire- und Leitwert-basierter Neuronen, aber auch mit abstrakteren Modellen neuronaler Aktivität und Spike Trains. Sie werden lernen, wie diese verschiedenen Berechnungsmodelle die zugrunde liegenden biologischen Prozesse in unterschiedlichem Maße beschreiben und vereinfachen. Wir werden im Detail untersuchen, wie diese verschiedenen Neuronenmodelle in numerischen Simulationen verwendet werden können, um Forschungsfragen zur Berechnung in einzelnen Neuronen und Schaltkreisen zu beantworten. In den vorlesungsbegleitenden Übungen werden Sie praktische Erfahrungen mit der Implementierung der verschiedenen Neuronenmodelle in Python, der Durchführung numerischer Simulationen und der Durchführung von Berechnungen im Zusammenhang mit analytischen Lösungen der Modellgleichungen und der Biophysik sammeln. Der Schwerpunkt liegt auf Einzelneuronenmodellen, aber wir werden auch verfügbare Software (z.B. NEST Desktop) nutzen, um zu untersuchen, wie Einzelneuronenmodelle in Simulationen von neuronalen Netzwerken integriert werden können. Während der Schwerpunkt des Moduls auf methodischen Fragen liegt und darauf, wie Modelle auf jeder Ebene aufgebaut, getestet und validiert werden können, werden wir auch Verbindungen zu bestimmten Gehirnregionen herstellen, um die Modelle zu motivieren und zu veranschaulichen. | | | | | |
| Lehrformen Vorlesung mit Übungen | | | | | |
| Prüfungsformen Schriftliche Prüfung (120 Minuten) | | | | | |
| Voraussetzungen für die Vergabe von Credits Erfolgreiches Bestehen der Modulklausur. | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 6/97: M.Sc. Informatik | | | | | |

| Titel des Moduls: Computational Neuroscience: Vision and Memory | | | | | |
|--|------------------------|--------------------------|---------------------------------|---------------------------------|---------------------------------------|
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Computational Neuroscience: Vision and Memory (211049) | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße 20 Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Laurenz Wiskott Lehrende: Prof. Dr. Laurenz Wiskott | | | | | |
| Verwendung des Moduls M.Sc. Informatik M.Sc. Angewandte Informatik M.Sc. Cognitive Science | | | | | |
| Lernziele (learning outcomes) After the successful completion of this course the students: <ul style="list-style-type: none"> • know basic neurobiological facts about the visual system and the hippocampus, • know a number of related models and methods in computational neuroscience, • understand the mathematics of these methods, • can communicate about all this in English. | | | | | |
| Inhalt This lecture covers basic neurobiology and models of selforganization in neural systems, in particular addressing | | | | | |
| Learning and self-organization <ul style="list-style-type: none"> • Hebbian Learning • Neural learning dynamics and constrained optimization • Dynamic field theory | | | | | |
| Vision <ul style="list-style-type: none"> • Receptive fields • Neural maps • Hippocampus • Navigation • Episodic memory • Hopfield Network | | | | | |
| Lehrformen This course is given with the flipped/inverted classroom concept. First, the students work through online material by themselves. In the lecture time slot we then discuss the material, find connections to other topics, ask questions and try to answer them. In the tutorial time slot the newly acquired knowledge is applied to analytical exercises and thereby deepened. I encourage all students to work in teams during self-study time as well as in the tutorial. | | | | | |
| Prüfungsformen The course is concluded with a digital written exam. | | | | | |
| Voraussetzungen für die Vergabe von Credits Passed written Exam | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) | | | | | |

5/97: M.Sc. Informatik

5/105: M.Sc. Angewandte Informatik

| | | | | | |
|---|------------------------|--------------------------------|---------------------------------|---------------------------------|------------------------------------|
| Titel des Moduls: Deep Learning | | | | | |
| Modul-Nr./Code | Credits 5 CP | Workload 150 Stunden | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen Deep Learning | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße Studierende |
| Unterrichtssprache Deutsch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Asja Fischer Lehrende: Prof. Dr. Asja Fischer | | | | | |
| Verwendung des Moduls M.Sc. IT-Sicherheit / Informationstechnik M.Sc. IT-Sicherheit / Netze und Systeme M.Sc. Angewandte Informatik M.Sc. Informatik | | | | | |
| Lernziele (learning outcomes) Die Vorlesung hat das Ziel, einen Einblick in dieses Gebiet zu vermitteln. Zu Beginn werden die grundlegenden Begriffe und Konzepte des maschinellen Lernens eingeführt. Im weiteren Verlauf wird auf verschiedene neuronale Netze, Gradienten-basierte Optimierungsverfahren und generative Modelle eingegangen. | | | | | |
| Inhalt Deep Learning ist ein Untergebiet des maschinellen Lernens, welches in den letzten Jahren zu Durchbrüchen in zahlreichen Anwendungsgebieten (wie z.B. in der Objekt- und Spracherkennung und der maschinellen Übersetzung) geführt hat. Deep Learning Methoden finden unter anderem Anwendung im Bereich IT Security | | | | | |
| Lehrformen Vorlesung und Übung | | | | | |
| Prüfungsformen Schriftliche Modulabschlussprüfung (120 Minuten) | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestandene schriftliche Modulabschlussprüfung. | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 5/91: M.Sc. IT-Sicherheit / Informationstechnik [PO 22] 5/84: M.Sc. IT-Sicherheit / Informationstechnik [PO 20] 5/99: M.Sc. IT-Sicherheit / Netze und Systeme [PO 22] 5/96: M.Sc. IT-Sicherheit / Netze und Systeme [PO 20] 5/105: M.Sc. Angewandte Informatik 5/ 97: M.Sc. Informatik | | | | | |

| | | | | | |
|--|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Titel des Moduls: Design Optimization Design Optimization | | | | | |
| Modul-Nr./Code | Credits 6 CP | Workload 180 h | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen Design Optimization (129007) | | | Kontaktzeit 60 h | Selbststudium 120 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Markus König Lehrende: Prof. Dr. Markus König | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) An important goal of this course is to present the theoretical foundations of mathematical optimization to students in a manner which allows them to use and employ design optimization for engineering applications in a sensible manner. This is achieved with a combination of theoretical lectures and practical exercises carried out using various computers software systems. In the second part of the course, students carry out team projects to solve engineering design tasks using their fundamental knowledge acquired in the first half of the course. Further, students must clearly present their projects results in a classroom setting to an audience with various technical background (the course is attended by students from applied computer science as well as by students from computational engineering). When the students have successfully complete this course, students <ul style="list-style-type: none"> • will be familiar with the types of numerical algorithms available today to solve, in particular, advanced engineering tasks; • will be able to program software components to carry out design optimization tasks or employ engineering software systems to include design optimization aspects; • will have a good understanding of the basics of design optimization to be able to select proper optimization techniques in a given engineering situation and be able to implement efficient numerical solutions. | | | | | |
| Inhalt <ul style="list-style-type: none"> • Structural optimization as a tool for the optimal design of engineering tasks with respect to given quality objective functions, side constraints as well as inequality constraints. • Development of optimization models for use in engineering applications • Types of optimization categories (continuous, linear/non-linear, deterministic, simulation-based, multi-level, etc.) • Strategies of optimization methods (classic indirect methods, direct numerical methods, global evolution strategies, partical swarm methods, distributed parallel methods, etc.) • Software systems to implent design optimization tasks • Application of design optimization paradigms to solve engineering tasks as a team project | | | | | |
| Lehrformen lectures, exercises using computers, team projects | | | | | |
| Prüfungsformen Completion of a team project (2-4 students); presentation of project results (end of semester); oral examination (end of semester) | | | | | |
| Voraussetzungen für die Vergabe von Credits successful completion of a team project; presentation of project results in the classroom; passed oral examination | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 6/105: M.Sc. Angewandte Informatik | | | | | |

| | | | | | |
|---|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Titel des Moduls: Deterministic Network Caluculus | | | | | |
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Deterministic Network Caluculus (211054) | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Steffen Bondorf Lehrende: Prof. Dr. Steffen Bondorf | | | | | |
| Verwendung des Moduls M.Sc. Informatik M.Sc. IT-Sicherheit/Informationstechnik M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) Nach erfolgreichem Abschluss des Moduls werden die Studierenden in der Lage sein, <ul style="list-style-type: none"> komplexe, vernetzte Systeme als deterministische Warteschlangensysteme zu modellieren, worst-case Leistungsanalysen von bestehenden Systemen bzw. Modellen durchzuführen, die Herausforderungen bei der Leistungsdimensionierung von geplanten Systemen zu verstehen, &#8729; dabei die Wirkungsweise zentraler Mechanismen in Computernetzen anhand des Network Calculus zu erklären, die vorgestellten Verfahren gegeneinander abzugrenzen und auf wissenschaftliche Fragestellungen anzuwenden. | | | | | |
| Inhalt Verteilte Systeme sind heutzutage allgegenwärtig, und ihre Vernetzung ist von grundlegender Bedeutung für die kontinuierliche Verbreitung und damit Verfügbarkeit von Daten. Die Bereitstellung von Daten in Echtzeit ist einer der wichtigsten nichtfunktionalen Aspekte, den sicherheitskritische Netze gewährleisten müssen. Die formale Verifizierung der Datenkommunikation im Hinblick auf die worst-case Deadlines ist grundlegend für die Zertifizierung von neu entwickelten x-by-Wire-Systemen. Diese Verifizierung erlaubt den Start von Flugzeugen, das Lenken von Autos ohne mechanische Verbindung und den Betrieb sicherheitskritischer Industrieanlagen. Daher wurden verschiedene Methoden für die worst-case Modellierung und Analyse von Echtzeitsystemen entwickelt. Eine davon ist der Deterministische Network Calculus (DNC), eine vielseitige Technik, die in verschiedenen Bereichen wie Paketvermittlung, Task Scheduling, System on Chip, softwaredefinierte Netzwerke, Netzwerke in Rechenzentren und Netzwerkvirtualisierung eingesetzt werden kann. DNC ist eine Methode zur Ableitung deterministischer Schranken für zwei der vorrangigsten Leistungsmetriken in Kommunikationssystemen: <ul style="list-style-type: none"> die Ende-zu-Ende-Verzögerung von Datenflüssen und der Speicherplatz, den ein Server benötigt, um alle eingehenden Daten in einer Warteschlange zu puffern. | | | | | |
| Lehrformen Vorlesung mit Übung | | | | | |
| Prüfungsformen Mündliche Modulabschlussprüfung über 30 Minuten | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestandene mündliche Modulabschlussprüfung. | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 5/97: M.Sc. Informatik 5/91: M.Sc. IT-Sicherheit/Informationstechnik [PO 22] | | | | | |

5/84: M.Sc. IT-Sicherheit/Informationstechnik [PO 20]

5/105: M.Sc. Angewandte Informatik

| | | | | | |
|--|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Titel des Moduls: Effiziente Algorithmen | | | | | |
| Modul-Nr./Code | Credits 9 CP | Workload 270 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Effiziente Algorithmen (150320 + 150321) | | | Kontaktzeit 90 h | Selbststudium 180 h | Gruppengröße Studierende |
| Unterrichtssprache Deutsch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: PD Dr. Daniela Kacso Lehrende: PD Dr. Daniela Kacso | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik M.Sc. IT-Sicherheit/Informationstechnik | | | | | |
| Lernziele (learning outcomes) Nach dem erfolgreichen Abschluss des Moduls: Die Studierenden <ul style="list-style-type: none"> • kennen, wählen aus und nutzen grundlegende Datenstrukturen und Graphenalgorithmen • sind in der Lage Analysetechniken (Korrektheitsbeweise und Laufzeitanalyse) zu erläutern und zu beurteilen • können auch bei praktischen Problemen entscheiden, welche der vermittelten Methoden/Algorithmen/Datenstrukturen anwendbar sind und diese nach Effizienz (insb. Laufzeit der Algorithmen) bewerten • können konkrete Anwendungsprobleme modellieren und bei Bedarf diese Algorithmen weiter entwickeln | | | | | |
| Inhalt Die Lehrveranstaltung kann sowohl in das Gebiet der praktischen als auch in das Gebiet der theoretischen Informatik eingeordnet werden. Die zentralen Themen sind die Folgenden: <ul style="list-style-type: none"> • Berechnung kürzester Pfade in Digraphen • Berechnung eines maximalen Flusses in einem Transportnetzwerk • Berechnung einer optimalen Lösung bei einem Zuordnungsproblem (auch Matching-Problem genannt) Darüberhinaus beschäftigen wir uns mit Anwendungen dieser grundlegenden Probleme. | | | | | |
| Lehrformen Vortrag der Lehrenden in der Vorlesung, Gruppenarbeit in den Übungen | | | | | |
| Prüfungsformen Schriftliche Modulabschlussprüfung (120 Minuten) | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestandene schriftliche Modulabschlussprüfung | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 9/105: M.Sc. Angewandte Informatik 9/91: M.Sc. IT-Sicherheit/Informationstechnik [PO 22] 9/84: M.Sc. IT-Sicherheit/Informationstechnik [PO 22] | | | | | |

| Titel des Moduls: Fachwissenschaftliche Vertiefung (Angewandte Informatik) | | | | | |
|--|------------------------|--------------------------|---------------------------------|---------------------------------|---------------------------------------|
| Modul-Nr./Code | Credits 6 CP | Workload 180 h | Semester | Turnus Jedes Semester | Dauer 2 Semester |
| Lehrveranstaltungen Seminar zu Approximationsalgorithmen Seminar Computational Neuroscience Seminar From Biological to Artificial Neural Networks Seminar Satisfiability Quantum Algorithms Seminar über Grenzen in der theoretischen Informatik Fortgeschrittene Themen des Model Checking Seminar Knowledge Graphs Seminar Topics in Deep Learning for Sequence Processing Seminar Implementation Security Seminar Ingenieurinformatik Seminar Bioinformatik Seminar Ressourceneffiziente Systemsoftware Advanced Topics in Deep Learning Seminar zur symmetrischen Kryptographie | | | Kontaktzeit 60 h | Selbststudium 120 h | Gruppengröße 20 Studierende |
| Unterrichtssprache Deutsch oder Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Studiendekan Angewandte Informatik Lehrende: siehe jeweiliges Seminar | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) Nach dem erfolgreichen Abschluss des Moduls <ul style="list-style-type: none"> • verfügen Studierende über aktuelle fachwissenschaftliche Kenntnisse aus 2 verschiedenen Vertiefungsrichtungen • können die Teilnehmer Lehrinhalte und Forschungsergebnisse eigenständig in einem didaktisch wohl aufbereiteten Vortrag vermitteln • können die Teilnehmer konstruktives Feedback formulieren und entgegennehmen • sind die Studierenden in der Lage, die Ergebnisse der eigenen Arbeit schriftlich zu dokumentieren | | | | | |
| Inhalt Es werden Vertiefungsseminare zu mehreren relevanten Themen angeboten, wie beispielsweise zu Deep Learning, zu Bioinformatik oder zu Ingenieurinformatik. Von den angebotenen Themen wählen die Studierenden abhängig von den eigenen Interessen und den individuellen Vertiefungswünschen in der Regel 2 verschiedene Seminarethemen aus. Diese sollen die Studierenden selbstständig bearbeiten. Dazu gehören die Literaturrecherche, die Einarbeitung in das Thema und schließlich die Präsentation sowie eine schriftliche Ausarbeitung. Nähere Informationen zu den jeweiligen Seminaren sind dem Vorlesungsverzeichnis zu entnehmen. Anmerkung: Die meisten Seminare werden mit 3 CP kreditiert. Hier müssen 2 verschiedenen Seminare absolviert werden (z.B. Ingenieurinformatik und Knowledge Graphs). Sollte ein Seminar mit 6 CP angeboten werden, so ist dieses ausreichend, um das Modul abzuschließen. | | | | | |
| Lehrformen Seminar | | | | | |
| Prüfungsformen | | | | | |

Semesterbegleitend; 1-2 Seminarvorträge zu unterschiedlichen Themenbereichen und evtl. jeweils eine schriftliche Ausarbeitung.

Voraussetzungen für die Vergabe von Credits

Die meisten Seminare werden mit 3 CP kreditiert. In diesem Fall errechnet sich die Modulnote als arithmetisches Mittel aus beiden Seminarnoten. Bei Seminaren mit 6 CP ist die jeweilige Seminarnote auch die Modulnote. Die Gesamtnote muss mindestens "ausreichend" sein.

Um die Lernziele zu erreichen, besteht im Seminaren Anwesenheitspflicht an mindestens 9 von 10 Terminen. Mehrfaches Fehlen muss durch ein ärztliches Attest entschuldigt werden. Die Anwesenheit beim ersten Termin ist obligatorisch, da zu diesem Termin die Themen verteilt werden. Ein Seminar gilt als nicht bestanden, wenn an mehr als einem Termin unentschuldigt gefehlt wurde.

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

6/105: M.Sc. Angewandte Informatik

| | | | | | |
|--|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Titel des Moduls: Fundamentals of GPU Programming Fundamentals of GPU Programming | | | | | |
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen Fundamentals of GPU Programming (141374) | | | Kontaktzeit 45 h | Selbststudium 105 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Dr. Denis Eremin Lehrende: Dr. Denis Eremin | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik M.Sc. IT-Sicherheit/Informationstechnik | | | | | |
| Lernziele (learning outcomes) Die Studierenden erlernen das Programmieren auf Grafikprozessoren (GPUs) | | | | | |
| Inhalt Zu einem bestimmten Zeitpunkt um 2003 stieg die Rechenleistung nicht auf Kosten der Taktfrequenz des Prozessors, sondern durch Erhöhung der Anzahl der auf dem Prozessorchip zugewiesenen Rechenkerne. Grafikprozessoren (GPUs) sind die Meister dieser Computer-Hardware-Entwicklung und bieten bis zu Zehntausende einzelner Kerneinheiten. Gleichzeitig wird das GPU-Speichersystem nicht so sehr durch die Kompatibilitätsanforderungen mit älteren Generationen eingeschränkt wie CPU-Speichersysteme. Deswegen zeigen GPUs im Vergleich zu ihren älteren "Bruder" -Zentraleinheiten (CPUs) eine deutlich bessere Rohleistung der Recheneinheiten und des Speichersystems. Ursprünglich für Videobearbeitungsaufgaben entwickelt, wird die enorme Rechenleistung moderner GPUs üblicherweise zur Unterstützung von CPUs oder zur Lösung einer Vielzahl von Rechenproblemen mit (massiv) parallelisierbaren Teilen verwendet, wodurch Teraflops-hohe Rechenleistung kann schon auf Laptop- / Desktop-Computers erzielt werden. Der vorliegende Kurs zeigt, wie CUDA C (Erweiterung der C-Sprache für die GPU-Programmierung) und das entsprechende (sehr flexible!) CUDA-Laufzeit-API-Framework verwendet werden kann, um die Ausführung einiger typischer Programmiermuster um einen Faktor von 10 oder mehr zu beschleunigen das der CPU. Ausgehend vom CUDA-Programmiermodell geht man zum CUDA-Ausführungsmodell über und betrachtet grundlegende konzeptionelle, Software- und Hardwareprobleme, die zum Verständnis der Funktionsweise von GPUs beitragen. Fallstudien zu mehreren Problemen mit massiv parallelen Algorithmen, die in GPUs implementiert sind, werden ebenfalls weiter ausgeführt. Das theoretische Wissen, das in den Vorlesungen vermittelt wird, wird durch eine Vielzahl von praktischen Beispielen untermauert, an denen die SchülerInnen zu Hause arbeiten können. | | | | | |
| Lehrformen Vorlesung mit Übung | | | | | |
| Prüfungsformen Semesterbegleitend: Projektarbeit und schriftliche Prüfung | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestehen der Projektarbeit und der schriftlichen Prüfung | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 5/105: M.Sc. Angewandte Informatik 5/91: M.Sc. IT-Sicherheit/Informationstechnik [PO 22] 5/84: M.Sc. IT-Sicherheit/Informationstechnik [PO 20] | | | | | |

| | | | | | |
|--|------------------------|--------------------------|---------------------------------|---------------------------------|---------------------------------------|
| Titel des Moduls: Geometrische Algorithmen | | | | | |
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen Geometrische Algorithmen | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße 20 Studierende |
| Unterrichtssprache Deutsch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Maïke Buchin Lehrende: Prof. Dr. Maïke Buchin | | | | | |
| Verwendung des Moduls M.Sc. Informatik M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) Lernziele (learning outcomes): Nach dem erfolgreichen Abschluss des Moduls <ul style="list-style-type: none"> • kennen Studierende grundlegende geometrische Algorithmen und Datenstrukturen • können Studierende Algorithmen nach dem Sweep-Paradigma analysieren und entwerfen • können Studierende inkrementelle Algorithmen entwerfen und analysieren, insbesondere randomisiert inkrementelle Algorithmen • können Studierende geometrische Algorithmen nach dem Teile-und-Herrsche Prinzip analysieren und entwerfen • können Studierende für Bereichsanfragen geeignete Datenstrukturen aussuchen | | | | | |
| Inhalt Die Algorithmische Geometrie beschäftigt sich mit dem Entwurf und der Analyse von Algorithmen und Datenstrukturen für geometrische Probleme. Dazu werden zunächst einige grundlegende Probleme betrachtet, wie das Berechnen der konvexen Hülle einer Punktmenge, der Schnittpunkte einer Menge von Strecken oder einer Triangulierung eines einfachen Polygons. Anschließend sehen wir Algorithmen zum Berechnen bekannter geometrische Strukturen, wie das Voronoi-Diagramm, die Delaunay-Triangulierung und Arrangements. Ebenfalls betrachten wir Datenstrukturen für effiziente Anfragen auf geometrischen Daten, wie Rangetrees, kd-Bäume und Quadrees. Dabei kommen vor allem drei Arten von Algorithmen zum Einsatz: inkrementell, teile-und-herrsche, und sweep. Manche von diesen treten als randomisierte Algorithmen auf. | | | | | |
| Lehrformen Vorlesung als kombinierter Folien- und Tafelvortrag und zugehörige Übungen. | | | | | |
| Prüfungsformen Mündliche Modulabschlussprüfung | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestandene mündliche Prüfung | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 5/105: M.Sc. Angewandte Informatik 5/97: M.Sc. Informatik | | | | | |

Titel des Moduls: High-Performance Computing on Clusters

| | | | | | |
|--|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 6 CP | Workload 180 h | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen High-Performance Computing on Clusters | | | Kontaktzeit 60 h | Selbststudium 120 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Andreas Vogel Lehrende: Prof. Dr. Andreas Vogel | | | | | |
| Verwendung des Moduls M.Sc. Informatik M.Sc. Angewandte Informatik M.Sc. Bauingenieurwesen M.Sc. Computational Engineering | | | | | |
| Lernziele (learning outcomes) In this module, the students acquire professional skills to program and employ parallel computing clusters. Theoretical properties of distributed-memory systems and programming patterns are conveyed as well as the practical implementation. | | | | | |
| Inhalt The lecture deals with the parallelization on cluster computers. Distributed-memory programming concepts (MPI) are introduced and best-practice implementation is presented based on applications from scientific computing including the finite element method and machine learning. Special attention is paid to scalable solvers for systems of equations on distributed memory systems, focusing on iterative schemes such as simple splitting methods (Richardson, Jacobi, Gauß-Seidel, SOR), Krylov-methods (Gradient descent, CG, BiCGStab) and, in particular, the multigrid method. The mathematical foundations for iterative solvers are reviewed, suitable object-oriented interface structures are developed and an implementation of these solvers for modern parallel computer architectures is developed. Numerical experiments and self-developed software implementations are used to discuss and illustrate the theoretical results. | | | | | |
| Lehrformen Beamer, computer lab, numerical experiments | | | | | |
| Prüfungsformen Written Exam (120 minutes) | | | | | |
| Voraussetzungen für die Vergabe von Credits Passed written Exam | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 6/97: M.Sc. Informatik 6/105: M.Sc. Angewandte Informatik | | | | | |

Titel des Moduls: High-Performance Computing on Multi- and Manycore Processors

| | | | | | |
|--|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 6 CP | Workload 180 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen High-Performance Computing on Multi- and Manycore Processors | | | Kontaktzeit 60 h | Selbststudium 120 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Andreas Vogel Lehrende: Prof. Dr. Andreas Vogel | | | | | |
| Verwendung des Moduls M.Sc. Informatik M.Sc. Angewandte Informatik M.Sc. Bauingenieurwesen M.Sc. Computational Engineering | | | | | |
| Lernziele (learning outcomes) In this module, the students acquire professional skills to program multi- and manycore processors employing multi-threaded execution and handling shared-memory access patterns. Theoretical properties are conveyed as well as practical implementation. Via presentations of selected topics, students attain the ability to survey and acquire knowledge on advanced scientific topics independently and are qualified to illustrate such topics in the form of a presentation and numerical examples. | | | | | |
| Inhalt The lecture addresses parallelization for multi- and manycore processors. Thread-based programming concepts (pthreads, C++11 threads, OpenMP, OpenCL) are introduced and bestpractice implementation aspects are highlighted based on applications from scientific computing. In the first part, the lecture provides an overview on relevant data structures, solver techniques and programming patterns from scientific computing. An introduction to multithreading programming on multicore systems is then provided with special attention to shared-memory aspects. Parallelization patterns are discussed and highlighted. Numerical experiments and self-developed software implementations are used to discuss and illustrate the presented content. In the second part, students are assigned advanced topics for sharedmemory computation from the engineering science including finite element methods and artificial intelligence. Based on a scientific paper, students present their topic to the lecture audience in form of a beamer presentation and numerical illustrations. | | | | | |
| Lehrformen Beamer, computer lab, numerical experiments | | | | | |
| Prüfungsformen Semesterbegleitend; Hausarbeit | | | | | |
| Voraussetzungen für die Vergabe von Credits mit mindestens ausreichend bewertete Hausarbeit | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 6/97: M.Sc. Informatik 6/105: M.Sc. Angewandte Informatik | | | | | |

| | | | | | |
|--|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Titel des Moduls: Information Theory | | | | | |
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Information Theory | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Michael Walter Lehrende: Prof. Dr. Michael Walter | | | | | |
| Verwendung des Moduls B.Sc. Informatik B.Sc. IT-Sicherheit M.Sc. Informatik M.Sc. IT-Sicherheit / Informationstechnik M.Sc. IT-Sicherheit / Netze und Systeme M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) You will learn fundamental concepts, algorithms, and results in information theory. After successful completion of this course, you will know the mathematical model of information theory, how to design and analyze algorithms for a variety of information processing tasks, and how to implement them in Python. You will have independently read about a topic in information theory and presented it to your peers. You will be prepared for an advanced course or a research or thesis project in this area. Please see the course homepage for a precise list of learning objectives. | | | | | |
| Inhalt This course will give an introduction to information theory – the mathematical theory of information. Ever since its inception, information theory has had a profound impact on society. It underpins important technological developments, from reliable memories to mobile phone standards, and its versatile mathematical toolbox has found use in computer science, machine learning, physics, electrical engineering, mathematics, and many other disciplines. Starting from probability theory, we will discuss how to mathematically model information sources and communication channels, how to optimally compress information, and how to design error-correcting codes that allow us to reliably communicate over noisy communication channels. We will also see how techniques used in information theory can be applied more generally to make predictions from noisy data. | | | | | |
| Tentative syllabus: - Welcome, Introduction to Information Theory - Probability Theory Refresher - Numerical Random Variables, Convexity and Concavity, Entropy - Symbol Codes: Lossless Compression, Huffman Algorithm - Block Codes: Shannon's Source Coding Theorem, its Proof, and Variations - Stream Codes: Lempel-Ziv Algorithm - Stream Codes: Arithmetic Coding | | | | | |

- Joint Entropies & Communication over Noisy Channels
- Shannon's Noisy Coding Theorem
- Proof of the Noisy Coding Theorem
- Proof of the Converse, Shannon's Theory vs Practice
- Reed-Solomon Codes
- Message Passing for Decoding and Inference, Outlook
- Student Presentations

Please see the course homepage https://qi.rub.de/it_ss23 for more information.

Lehrformen

Lecture with Exercise

Prüfungsformen

Wird noch nachgeliefert

Voraussetzungen für die Vergabe von Credits

Passed Exam.

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

5/158: B.Sc. Informatik [PO 22]

5/170: B.Sc. Informatik [PO 20]

5/150: B.Sc. IT-Sicherheit [PO 22]

5/149: B.Sc. IT-Sicherheit [PO 20]

5/97: M.Sc. Informatik

5/91: M.Sc. IT-Sicherheit / Informationstechnik [PO 22]

5/84: M.Sc. IT-Sicherheit / Informationstechnik [PO 20]

5/99: M.Sc. IT-Sicherheit / Netze und Systeme [PO 22]

5/105: M.Sc. Angewandte Informatik

Titel des Moduls: Inklusives IT-Design
Inclusive IT design

| | | | | | |
|--------------------------------------|------------------------|--------------------------|---|---------------------------------|---------------------------------------|
| Modul-Nr./Code | Credits 6 CP | Workload 180 h | Semester siehe Prüfungsordnung | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen | | | Kontaktzeit 60 h | Selbststudium 120 h | Gruppengröße 20 Studierende |
| Unterrichtssprache Deutsch | | | Teilnahmevoraussetzungen Keine | | |

Modulbeauftragte/r und hauptamtlich Lehrende

Modulbeauftragte/r: Prof. Dr. Christian Meske
 Lehrende: Prof. Dr. Christian Meske
 Markus Jelonek, M.Sc.

Verwendung des Moduls

B.Sc. Angewandte Informatik
 M.Sc. Angewandte Informatik

Lernziele (learning outcomes)

Die Veranstaltung richtet sich an Studierende mit Interesse an Themen der Inklusion und dem Nutzerzentrierten Design.

Die Studierenden

- ... kennen die Perspektiven von Endnutzer:innen mit geistiger und/oder körperlicher Behinderung hinsichtlich Interaktionsdesign, Usability und User Experience und können diese bei der Entwicklung eines Prototyps anwenden, begründen und beurteilen.
- ... leisten einen Dienst an der Gesellschaft, indem sie reale Probleme bzgl. der Inklusion von Menschen mit Behinderungen im Rahmen der digitalen Transformation aufgreifen, analysieren, beurteilen und gemeinsam prototypische Lösungen konstruieren (Service Learning).
- ... können partizipative und Design Science Methoden mit Aspekten der Inklusion und des Wohlbefindens in zukünftigen IT-Design-Prozessen zusammenführen und beurteilen.
- ... können theoretisch-konzeptionelle Design-Science-Vorgehensmodelle mit einer praxisorientierten Anwendung im Themenfeld des Positive Computing anwenden.
- ... kennen Anforderungen an inklusives IT-Design und können diese praktisch anwenden.
- ... können Fachwissen, welches in anderen Fächern erworben wurde (UX, Software-Ergonomie und Usability Engineering, Positive Computing, etc.), problemorientiert anwenden und beurteilen sowie neues Wissen selbstständig aneignen.
- ... können zielgerichtet und interdisziplinär zusammenarbeiten sowie Lösungen in einem festen Zeitraum eigenständig konstruieren, analysieren und präsentieren.
- ... können die Vorteile und Herausforderungen bei der Entwicklung von Anwendungen mit und für die spezifische Zielgruppe beschreiben.
- ... können Aspekte des agilen IT-Projektmanagements für Ihren Praxisteil anwenden.
- ... können Aspekte des Positive Computing in ihrer Prototypenwicklung anwenden.

Inhalt

- Inclusive und accessible IT-Design
- Universal Design vs. Inclusive and Accessible Design
- Barrierefreiheit
- UX und Usability in inclusive und accessible IT-Design
- Design Science Research
- Partizipative Methoden und Co-Creation

Lehrformen

- Vorlesung (Flipped Classroom, hybrid) mit praktischen Phasen in Kooperation mit Studierenden der Hochschule Ruhr-West (digital und analog)
- Praxisorientierte Blockveranstaltungen und Projektphasen mit Praxispartner vor Ort und digital

Prüfungsformen

Projektarbeit

Voraussetzungen für die Vergabe von Credits

- Regelmäßige Teilnahme an Pflichtterminen
- Bestandene Zwischen- und Abschlusspräsentation
- Bestandene Abgabe der Dokumentation und Präsentation

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

6/168: B.Sc. Angewandte Informatik PO22

6/170: B.Sc. Angewandte Informatik PO20

6/105: M.Sc. Angewandte Informatik

| Titel des Moduls: Introduction to Computational Linguistics in Python | | | | | |
|---|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen Introduction to Computational Linguistics in Python (050825) | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Tibor Kiss Lehrende: Mirjam Koch M.Sc. | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) <ul style="list-style-type: none"> • Die Studierenden kennen Grundlagen der Programmierung mit Python mit einem besonderen Fokus auf kommentiertem und strukturiertem Code. • Sie können Fragen zu Programmierproblemen angemessen formulieren und beantworten. • Sie können Anforderungen an Programme formulieren und umsetzen. • Sie haben gelernt, bestehende Komponenten einzubinden und angemessen auf Quellen zu verweisen. • Sie verstehen den Unterschied zwischen Computational Linguistics, Informatik und NLP-Verfahren. • Sie kennen grundlegende Fragestellungen aus der Computerlinguistik und kennen die linguistische Perspektive auf NLP-Verfahren. | | | | | |
| Inhalt Das Modul bietet eine Einführung in die Programmiersprache R aus der Perspektive der Computerlinguistik (und Linguistic Data Science). Es beleuchtet Grundlagenverfahren und spezielle Fallstricke kennen, auf die sie bei der Verarbeitung von Sprachdaten für linguistische Fragestellungen achten müssen. Dabei werden einerseits externe Module genutzt und eingesetzt, andererseits aber vor allem Funktionen auch selbst entwickelt, um ihre Funktion nachzuvollziehen und zu verstehen, welche Überlegungen sie auch bei der reinen Anwendung von „fertigen“ Programmen berücksichtigen müssen. In den „Vorlesungen“ werden die vorher hochgeladenen Folien besprochen, nicht vorgetragen. Für die Klausur werden nur Inhalte der Folien vorausgesetzt. Die begleitenden Übungen zur Vorlesung sind nicht verpflichtend, aber sind zur Selbstkontrolle offen für alle Studierenden. | | | | | |
| Lehrformen Die Veranstaltung setzt sich aus einer Präsenzkomponente (Fragen und Antworten zu den Folien, Diskussionen, praktische Übungen und Überlegungen zum aktuellen Thema) und Übungen zum Kurs zusammen. Die Studierenden müssen sich eigenständig auf die Sitzungen vorbereiten und entscheiden, ob die Teilnahme ihr Verständnis fördert. | | | | | |
| Prüfungsformen Schriftliche Modulabschlussprüfung | | | | | |
| Voraussetzungen für die Vergabe von Credits | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 5/105: M.Sc. Angewandte Informatik | | | | | |

Titel des Moduls: Introduction to Linguistic Data Science with R

| | | | | | |
|---|------------------------|--------------------------|---------------------------------|---------------------------------|---------------------------------------|
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen Introduction to Linguistic Data Science with R (050826) | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße 20 Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |

Modulbeauftragte/r und hauptamtlich Lehrende

Modulbeauftragte/r: Prof. Dr. Tibor Kiss
Lehrende: Prof. Dr. Tibor Kiss

Verwendung des Moduls

M.Sc. Angewandte Informatik

Lernziele (learning outcomes)

- Die Studierenden erwerben grundlegende Modelle und Methoden der Linguistic Data Science und können sie kritisch auf Fragestellungen und Probleme anwenden.
- Sie werden mit der Fachsprache vertraut und können sich gut schriftlich und mündlich zu den behandelten Themen ausdrücken.
- Sie verstehen die Position von Linguistic Data Science in Abgrenzung zu verwandten Fächern wie reiner Linguistik oder allgemeiner Data Science und haben einen ersten Überblick über mögliche Vertiefungen in Abgrenzung zu Computerlinguistik und theoretischer Linguistik.
- Sie verstehen einige grundlegende Evaluationsmethoden und können sie auf kleine Beispielfälle anwenden.
- Sie vollziehen den Einfluss von Entscheidungen bei der Wahl von Datensätzen, Datentypen oder der Struktur von Experimenten auf Daten in einfachen Beispielen nach.
- Sie können sich über kleine fachspezifische Problemstellungen mit andere austauschen und diese gemeinsam lösen.
- Sie erwerben den Umgang mit der Programmiersprache R und können Programme schreiben, die einzelne Funktionen erfüllen, sowie Programme, die mehrere Module/Funktionen kombinieren.

Inhalt

Das Modul bietet eine Einführung in die Verfahren, Methoden und Modelle sowie Theorien aus dem Bereich Linguistic Data Science. Es werden zunächst diejenigen Konzepte der Programmiersprache R vorgestellt, die eine Repräsentation und Manipulation linguistischer Daten (Rohdaten, Annotationen, experimentell erhobene Daten, Datenübersichten) ermöglichen. Anschließend erfolgt die Einführung in grundlegende lineare Modelle, sowie deren Grundannahmen und Verfahren zur Evaluation.

Darüber hinaus werden Entwicklungsverfahren und Projektkonzeption vorgestellt.

In den Übungen werden kleine Aufgaben von den Studierenden in Gruppen oder allein bearbeitet.

Lehrformen

Die Veranstaltung setzt sich aus einer Präsenzkomponente (Vortrag der Lehrenden im Plenum mit anschließender Diskussion, gemeinsames Lösen von Aufgaben) und Übungen zum Kurs zusammen. Dabei können insbesondere die Übungen als eLearning-Einheiten angeboten werden.

Prüfungsformen

Schriftliche Modulabschlussprüfung

Voraussetzungen für die Vergabe von Credits

Bestandene schriftliche Modulabschlussprüfung

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

5/105: M.Sc. Angewandte Informatik

Titel des Moduls: KIbox: Lösung von Nachhaltigkeits-Challenges durch den Einsatz künstlicher Intelligenz

| | | | | | |
|--|------------------------|--------------------------|---------------------------------|---|------------------------------------|
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Jedes Semester (nicht SS 23!) | Dauer 1 Semester |
| Lehrveranstaltungen KIbox: Lösung von Nachhaltigkeits-Challenges durch den Einsatz künstlicher Intelligenz | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße Studierende |
| Unterrichtssprache Deutsch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Christian Meske Lehrende: Prof. Dr. Jens Pöppelbuß Prof. Dr. Christian Meske | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) Die Studierenden erwerben durch das Absolvieren der Lehrveranstaltung folgende Kompetenzen: <ul style="list-style-type: none"> • Die „Kickbox“-Innovationsmethode erläutern und anwenden, • Beispielhafte Nachhaltigkeits-Challenges identifizieren und analysieren, • Potenziale von KI zur Lösung von Nachhaltigkeits-Challenges identifizieren, • Kreativitätstechniken, Ideation- und Prototyping-Methoden (z. B. aus dem Design Thinking sowie Lean-Start-Up) für den eigenen Innovationsprozess auswählen und anwenden, • Eigene kreative Ideen präsentieren und testen und mittels Feedbacks überarbeiten, • Ziele definieren, Projekte managen und Meilensteine eigenständig erreichen, Teamarbeit unter Einsatz von begrenzten Ressourcen koordinieren, • KI-Lösungen prototypisch implementieren, • Unternehmerisches Handeln, • Nachhaltigkeit anhand etablierter Dimensionen (z. B. Triple-Bottom-Line und Corporate Social Responsibility) zur Bewertung eigener Ideen nutzen. | | | | | |
| Inhalt <ul style="list-style-type: none"> • Basierend auf dem Kickbox-Innovationsprozess sollen Studierende für gesellschafts- und praxisrelevante Herausforderungen eigenständig Problemlösungen durch den Einsatz künstlicher Intelligenz (KI) entwickeln. • Als Ausgangspunkt des Lehrformats werden Nachhaltigkeits-Challenges formuliert, bei denen KI zu einem umweltfreundlicheren und wettbewerbsfähigeren Europa beitragen kann. • Die für das Lehrformat spezifisch zu gestaltende Kickbox (KIbox) unterstützt die Studierenden im Innovationsprozess. | | | | | |
| Lehrformen <ul style="list-style-type: none"> • Vereinzelt Vorlesungen/ Inputs von Dozierenden • Projektarbeit und Projektveranstaltungen als Schwerpunkt • Feedback und Möglichkeiten zum Austausch mit Dozierenden | | | | | |
| Prüfungsformen Es sind zwei Prüfungsleistungen zu erbringen. Abschlusspräsentation (Gruppenleistung, Dauer: 15 Minuten plus anschließende Diskussion von 10 Minuten; 100 % der Endnote) Abschlussbericht (Gruppenleistung; Bestehen notwendig zum Abschluss des Moduls und zum Erwerb der Leistungspunkte) Prüfungstermine: Die Prüfungstermine werden zu Beginn des Semesters bekanntgegeben. | | | | | |
| Voraussetzungen für die Vergabe von Credits <ul style="list-style-type: none"> • Prüfungsleistung bestanden und • Um das Modul erfolgreich abzuschließen ist es außerdem notwendig, an mindestens 60% der Treffen im Plenum teilzunehmen | | | | | |

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)
5/105: M.Sc. Angewandte Informatik

| | | | | | |
|--|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Titel des Moduls: Knowledge Graphs | | | | | |
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Knowledge Graphs | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Jun.-Prof. Maribel Acosta Deibe Lehrende: Jun.-Prof. Maribel Acosta Deibe | | | | | |
| Verwendung des Moduls M.Sc. Informatik M.Sc. Angewandte Informatik M.Sc. IT-Sicherheit Informationstechnik | | | | | |
| Lernziele (learning outcomes) In this lecture, students will learn about the foundations of modelling, querying, publishing, and reasoning over KGs. The topics will be complemented with exercises and Jupyter Notebooks (https://jupyter.org/) to show how KG technologies work in practice. | | | | | |
| Inhalt Knowledge Graphs (KG) allow for representing inter-connected facts or statements annotated with semantics. In KGs, concepts and entities are typically modeled as nodes while their connections are modeled as directed and labeled edges, creating a graph. In recent years, KGs have become core components of modern data ecosystems. KGs, as building blocks of many Artificial Intelligence approaches, allow for harnessing and uncovering patterns from the data. Currently, KGs are used in the data-driven business processes of multinational companies like Google, Microsoft, IBM, eBay, and Facebook. Furthermore, thousands of KGs are openly available on the web following the Linked Data (https://lod-cloud.net/) principles. The specific topics covered in the lecture are as follows: <ol style="list-style-type: none"> 1. Introduction to Knowledge Graphs 2. The Resource Description Framework (RDF) 3. RDF Schema (RDFS) 4. The SPARQL Query Language 5. Semantics of SPARQL 6. Linked Data: Knowledge Graphs and Ontologies on the Web 7. The Web Ontology Language (OWL) 8. Entailment Regimes 9. Reasoning over Knowledge Graphs 10. Property Graphs 11. Knowledge Graph Applications | | | | | |
| Lehrformen Lecture with Exercise | | | | | |

Prüfungsformen

Written Exam (120 minutes)

Voraussetzungen für die Vergabe von Credits

Passed written Exam.

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

5/97: M.Sc. Informatik

5/105: M.Sc. Angewandte Informatik

5 /91 M.Sc. IT-Sicherheit | Informationstechnik

| | | | | | |
|--|------------------------|--------------------------|---------------------------------|--|------------------------------------|
| Titel des Moduls: Komplexitätstheorie Complexity Theory | | | | | |
| Modul-Nr./Code | Credits 9 CP | Workload 270 h | Semester | Turnus Unregelmäßig (i.d.R. Sommersemester) | Dauer 1 Semester |
| Lehrveranstaltungen Komplexitätstheorie | | | Kontaktzeit 90 h | Selbststudium 180 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Thomas Zeume Lehrende: Prof. Thomas Zeume | | | | | |
| Verwendung des Moduls M.Sc. Informatik M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) Die Studierenden lernen, algorithmische Probleme bezüglich ihrer Komplexität einzuordnen und so geeignete algorithmische Techniken zu ihrer Lösung zu identifizieren. Sie können insbesondere algorithmische Methoden für NP-vollständige Probleme anwenden. Sie können mit unterschiedlichen Berechnungsmodellen umgehen und sind in der Lage, einfache Aussagen über sie zu beweisen. Sie lernen im Diskurs eigene und fremde Lösungsansätze zu bewerten. | | | | | |
| Inhalt Die Komplexitätstheorie untersucht und klassifiziert Berechnungsprobleme bezüglich ihrer algorithmischen Schwierigkeit. Ziel ist es, den inhärenten Ressourcenverbrauch bezüglich verschiedener Ressourcen wie Rechenzeit oder Speicherplatz zu bestimmen, und Probleme mit ähnlichem Ressourcenverbrauch in Komplexitätsklassen zusammenzufassen. Die bekanntesten Komplexitätsklassen sind sicherlich P und NP, die die in polynomieller Zeit lösbaren bzw. verifizierbaren Probleme umfassen. Die Frage, ob P und NP verschieden sind, wird als eine der bedeutendsten offenen Fragen der theoretischen Informatik, ja sogar der Mathematik, angesehen. P und NP sind jedoch nur zwei Beispiele von Komplexitätsklassen. Andere Klassen ergeben sich unter anderem bei der Untersuchung der des benötigten Speicherplatzes, der effizienten Parallelisierbarkeit von Problemen, der Lösbarkeit durch zufallsgesteuerte Algorithmen, und der approximativen Lösbarkeit von Problemen. Die Vorlesung hat das Ziel, einen breiten Überblick über die grundlegenden Konzepte und Resultate der Komplexitätstheorie zu geben: <ul style="list-style-type: none"> • Klassische Resultate für Platz- und Zeitkomplexitätsklassen: z.B. die Korrespondenz zwischen Spielen und Speicherplatz-Beschränkungen, der Nachweis, dass sich mit mehr Platz oder Zeit auch mehr Probleme lösen lassen, weitere grundlegende Beziehungen zwischen Zeit- und Platzbasierten Klassen, und die Komplexitätswelt zwischen NP und PSPACE • Grundzüge der Komplexitätstheorie paralleler, zufallsbasierter und approximativer Algorithmen • Einführung in ausgewählte neuere Themen: Komplexitätstheorie des interaktiven Rechnens, des probabilistischen Beweisens und Fine-grained Complexity. | | | | | |
| Lehrformen Vorlesung mit Übungen | | | | | |
| Prüfungsformen Abschlussprüfung; mündlich, 20-30min | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestandene mündliche Modulabschlussprüfung | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 9/105: M.Sc. Angewandte Informatik [PO22] | | | | | |

9/97: M.Sc. Informatik [PO 23]

9/91: M.Sc. IT-Sicherheit | Informationstechnik [PO 22]

9/99: M.Sc. IT-Sicherheit | Netze und Systeme [PO22]

| | | | | | |
|---|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Titel des Moduls: Kryptographie Cryptography | | | | | |
| Modul-Nr./Code | Credits 8 CP | Workload 240 h | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen Kryptographie (212017) | | | Kontaktzeit 90 h | Selbststudium 150 h | Gruppengröße Studierende |
| Unterrichtssprache Deutsch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Alexander May Lehrende: Prof. Alexander May | | | | | |
| Verwendung des Moduls B.Sc. IT-Sicherheit/Informationstechnik M.Sc. IT-Sicherheit/Netze und Systeme M.Sc. Informatik M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) Die Studierenden haben ein Verständnis der wesentlichen mathematischen Methoden und Verfahren, auf denen moderne kryptographische Verfahren beruhen. Die Tiefe der Behandlung der Verfahren geht deutlich über das in den vorhergehenden Veranstaltungen vermittelte Maß hinaus. Die Teilnehmer sind zur Analyse und dem Design aktueller und zukünftiger kryptographischer Methoden befähigt. Zudem weisen sie ein Bewusstsein für Methodik und Mächtigkeit verschiedenster Angriffsszenarien auf. | | | | | |
| Inhalt Es wird eine Einführung in moderne Methoden der symmetrischen und asymmetrischen Kryptographie geboten. Dazu wird ein Angreifermodell definiert und die Sicherheit der vorgestellten Verschlüsselungs-, Hash- und Signaturverfahren unter wohldefinierten Komplexitätsmaßnahmen in diesem Angreifermodell nachgewiesen. Themenübersicht: <ul style="list-style-type: none">• Sichere Verschlüsselung gegenüber KPA-, CPA- und CCA-Angreifern• Pseudozufallsfunktionen und -permutationen• Message Authentication Codes• Kollisionsresistente Hashfunktionen• Blockchiffren• Konstruktion von Zufallszahlengeneratoren• Diffie-Hellman Schlüsselaustausch• Trapdoor Einwegpermutationen• Public Key Verschlüsselung: RSA, ElGamal, Goldwasser-Micali, Rabin, Paillier• Einwegsignaturen• Signaturen aus kollisionsresistenten Hashfunktionen• Random-Oracle Modell | | | | | |
| Lehrformen Vorlesung und Übungen | | | | | |
| Prüfungsformen Schriftliche Modulabschlussprüfung (120 Minuten) | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestandene schriftliche Modulabschlussprüfung. | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) | | | | | |

8/150: B.Sc. IT-Sicherheit/Informationstechnik [PO 22]

8/149: B.Sc. IT-Sicherheit/Informationstechnik [PO 20]

8/99: M.Sc. IT-Sicherheit/Netze und Systeme [PO 22]

8/96: M.Sc. IT-Sicherheit/Netze und Systeme [PO 20]

8/97: M.Sc. Informatik

8/105: M.Sc. Angewandte Informatik

| | | | | | |
|---|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Titel des Moduls: Kryptographische Protokolle | | | | | |
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Kryptographische Protokolle | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Eike Kiltz Lehrende: Prof. Dr. Eike Kiltz | | | | | |
| Verwendung des Moduls M.Sc. IT-Sicherheit / Informationstechnik M.Sc. IT-Sicherheit / Netze und Systeme M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) - Vertiefung des Verständnisses für beweisbare Sicherheit - Schreiben von fehlerfreien Sicherheitsreduktionen - Neue Techniken für Sicherheitsbeweise - Erlernen fortgeschrittener kryptographischer Konstruktionen | | | | | |
| Inhalt Die Vorlesung beschäftigt sich mit erweiterten kryptographischen Protokollen und deren Anwendungen. Themenübersicht: <ul style="list-style-type: none"> • Game-based security definitions and proofs • Bilinear maps • Digital Signatures • Identification Protocols • Zero-Knowledge Proofs • Identity-based Encryption • CCA-secure encryption | | | | | |
| Lehrformen Vorlesung mit Übung | | | | | |
| Prüfungsformen Mündliche (30 Minuten) oder schriftliche Modulabschlussprüfung (120 Minuten) | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestandene Modulabschlussprüfung. | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 5/91: M.Sc. IT-Sicherheit / Informationstechnik [PO 22] 5/84: M.Sc. IT-Sicherheit / Informationstechnik [PO 20] 5/99: M.Sc. IT-Sicherheit / Netze und Systeme [PO 22] 5/96: M.Sc. IT-Sicherheit / Netze und Systeme [PO 20] 5/105: M.Sc. Angewandte Informatik | | | | | |

| | | | | | |
|---|------------------------|--------------------------|--|---------------------------------|------------------------------------|
| Titel des Moduls: Linguistic Data Science | | | | | |
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen Linguistic Data Science (050826) | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen Erfolgreicher Abschluss des Moduls "Introduction to Linguistic Data Science with R" | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Tibor Kiss Lehrende: Prof. Dr. Tibor Kiss | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) <ul style="list-style-type: none"> • Die Studierenden erwerben grundlegende Modelle und Methoden der Linguistic Data Science und können sie kritisch auf Fragestellungen und Probleme anwenden. • Sie werden mit der Fachsprache vertraut und können sich gut schriftlich und mündlich zu den behandelten Themen ausdrücken. • Sie verstehen die Position von Linguistic Data Science in Abgrenzung zu verwandten Fächern wie reiner Linguistik oder allgemeiner Data Science und haben einen ersten Überblick über mögliche Vertiefungen in Abgrenzung zu Computerlinguistik und theoretischer Linguistik. • Sie verstehen einige grundlegende Evaluationsmethoden und können sie auf kleine Beispielfälle anwenden. • Sie vollziehen den Einfluss von Entscheidungen bei der Wahl von Datensätzen, Datentypen oder der Struktur von Experimenten auf Daten in einfachen Beispielen nach. • Sie können sich über kleine fachspezifische Problemstellungen mit andere austauschen und diese gemeinsam lösen. • Sie erwerben den Umgang mit der Programmiersprache R und können Programme schreiben, die einzelne Funktionen erfüllen, sowie Programme, die mehrere Module/Funktionen kombinieren. | | | | | |
| Inhalt Das Modul bietet eine Vertiefung in die Verfahren, Methoden und Modelle sowie Theorien aus dem Bereich Linguistic Data Science. Auf der Basis der erfolgten Einführung in grundlegende lineare Modelle, sowie deren Grundannahmen und Verfahren zur Evaluation werden schrittweise Generalized Linear Models und Generalized Linear Mixed Models vorgestellt und durch Entwicklungskonzepte und Fragestellungen der Projektkonzeption unterfüttert. In den Übungen werden kleine Aufgaben von den Studierenden in Gruppen oder allein bearbeitet. | | | | | |
| Lehrformen Die Veranstaltung setzt sich aus einer Präsenzkomponekte (Vortrag der Lehrenden im Plenum mit anschließender Diskussion, gemeinsames Lösen von Aufgaben) und Übungen zum Kurs zusammen. Dabei können insbesondere die Übungen als eLearning-Einheiten angeboten werden. | | | | | |
| Prüfungsformen Schriftliche Modulabschlussprüfung | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestandene schriftliche Modulabschlussprüfung | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 5/105: M.Sc. Angewandte Informatik | | | | | |

| | | | | | |
|---|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Titel des Moduls: Machine Learning: Evolutionary Algorithms | | | | | |
| Modul-Nr./Code | Credits 6 CP | Workload 180 h | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen Machine Learning: Evolutionary Algorithms | | | Kontaktzeit 60 h | Selbststudium 120 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Tobias Glasmachers Lehrende: | | | | | |
| Verwendung des Moduls Master Angewandte Informatik Master Informatik | | | | | |
| Lernziele (learning outcomes) Internationalisierung: Die Veranstaltung wird auf Englisch durchgeführt. Digitalisierung: Inhalte werden durch Videos und Lesematerial vermittelt. Übungsaufgaben mit Programmieranteilen werden in Form von Jupyter-Notebooks bereitgestellt. Nach dem erfolgreichen Abschluss des Moduls <ul style="list-style-type: none"> • kennen die Teilnehmer die wichtigsten Klassen direkter Optimierungsverfahren und ihre algorithmischen Komponenten, • haben die Teilnehmer ein tiefes Verständnis evolutionärer Algorithmen, insbesondere für kontinuierliche Probleme, • kennen die Teilnehmer eine Reihe spezifischer Problemschwierigkeiten und die zugehörigen algorithmischen Komponenten, welche diese Adressieren, • können die Teilnehmer elementare Laufzeitanalysen durchführen und verstehen die wichtigsten Konvergenzklassen; können die Teilnehmer Optimierungsverfahren selbst implementieren und zur Lösung neuer Probleme anwenden. | | | | | |
| Inhalt Breiter Überblick über Optimierungsverfahren. Evolutionäre Optimierungsverfahren für black-box Optimierungsverfahren Algorithmische Komponenten für schlechte Konditionierung, Multimodalität, Rauschen, Nebenbedingungen und Mehrzieloptimierung. Konvergenz und Laufzeitanalyse. | | | | | |
| Lehrformen flipped classroom | | | | | |
| Prüfungsformen Abschlussprüfung: Klausur 90 Minuten | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestandene schriftliche Modulabschlussprüfung | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 6/105: M.Sc. Angewandte Informatik 6/97: M.Sc. Informatik | | | | | |

| | | | | | |
|---|------------------------|--------------------------------|---------------------------------|---------------------------------|---------------------------------------|
| Titel des Moduls: Machine Learning: Supervised Methods | | | | | |
| Modul-Nr./Code | Credits 6 CP | Workload 180 Stunden | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Machine Learning: Supervised Methods | | | Kontaktzeit 60 h | Selbststudium 120 h | Gruppengröße 80 Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Tobias Glasmachers Lehrende: Prof. Dr. Tobias Glasmachers | | | | | |
| Verwendung des Moduls Master Informatik Master Angewandte Informatik Master IT-Sicherheit/ Informationstechnik | | | | | |
| Lernziele (learning outcomes) Internationalisierung: Die Veranstaltung wird auf Englisch durchgeführt. Digitalisierung: Inhalte werden durch Videos und Lesematerial vermittelt. Übungsaufgaben mit Programmieranteilen werden in Form von Jupyter-Notebooks bereitgestellt. Nach dem erfolgreichen Abschluss des Moduls <ul style="list-style-type: none"> • verstehen die Teilnehmer die Grundlagen der statistischen Lerntheorie • kennen die Teilnehmer die wichtigsten Algorithmen des überwachten statistischen Lernens und können diese auf Lernprobleme anwenden, • kennen die Teilnehmer Stärken und Beschränkungen verschiedenen Lernverfahren, • können die Teilnehmer Standardsoftware zum maschinellen Lernen zur Lösung neuer Probleme einsetzen. | | | | | |
| Inhalt Grundlagen der statistischen Lerntheorie, Querschnitt der wichtigsten Algorithmen des maschinellen Lernens, konkrete Problemlösung mit Standardsoftware | | | | | |
| Lehrformen flipped classroom | | | | | |
| Prüfungsformen Abschlussprüfung; Klausur 90 Minuten, | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestandene schriftliche Modulabschlussprüfung | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 6/105: M.Sc. Angewandte Informatik 6/97: M.Sc. Informatik 6/91: M.Sc IT-Sicherheit Informationstechnik | | | | | |

| Titel des Moduls: Machine Learning: Unsupervised Methods | | | | | |
|--|------------------------|--------------------------------|---------------------------------|---------------------------------|---------------------------------------|
| Modul-Nr./Code | Credits 9 CP | Workload 270 Stunden | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen Machine Learning: Unsupervised Methods | | | Kontaktzeit 60 h | Selbststudium 210 h | Gruppengröße 40 Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Laurenz Wiskott Lehrende: Prof. Dr. Laurenz Wiskott | | | | | |
| Verwendung des Moduls Master Informatik Master Angewandte Informatik Master Elektro-und Informationstechnik | | | | | |
| Lernziele (learning outcomes) After the successful completion of this course the students: <ul style="list-style-type: none"> • know a number of important unsupervised learning methods, • can discuss and decide which of the methods are appropriate for a given data set, • understand the mathematics of these methods, • know how to implement and apply these methods in python, • have gained experience in organizing and working in a team, • know problem solving strategies like brain storming, • can communicate about all this in English. | | | | | |
| Inhalt This course covers a variety of shallow unsupervised methods from machine learning such as principal component analysis, independent component analysis, vector quantization, clustering, Bayesian theory and graphical models. | | | | | |
| Lehrformen This course is given in a hybrid of inverted classroom and problem based learning. The course starts with a two-week introduction into unsupervised methods of machine learning, providing an overview. The students then work in groups of about 4 on realistic problems that can be solved with these methods. In the first week of a problem, they develop hypotheses and strategies for a solution and identify which methods they want to learn. Then the course agrees on a method to focus on theoretically, which will then be done in an inverted classroom format. The students then try to solve the problem and present their results in a short talk with slides. Thus the students will not only learn about machine learning but also soft skills. | | | | | |
| Prüfungsformen Semesterbegleitend; The exam is a combination of graded presentations for the problems and graded quizzes for the theory. 40% of the grade come from the average group performance on solving the problems. 10% come from the presentations, taking into account slides and presentation style, this is an individual grade of the presenter. 50% come from a digital quiz about the theory of the methods covered. Thus 60% of the grade are individual, 40% come from the group. In addition you can gain up to 8 bonus points for (i) being voted for as a 'most valuable player (MVP)' on a project and (ii) creating useful quiz questions. | | | | | |
| Voraussetzungen für die Vergabe von Credits Continuous participation and passed exam. | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 9/105: M.Sc. Angewandte Informatik | | | | | |

Titel des Moduls: Master-Praktikum Deep Learning and Natural Language Processing

| | | | | | |
|---|------------------------|--------------------------|---------------------------------|---------------------------------|--------------------------------------|
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen | | | Kontaktzeit 30 h | Selbststudium 120 h | Gruppengröße 3 Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Asja Fischer Lehrende: Prof. Dr. Asja Fischer | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) After successful completion of this module, <ul style="list-style-type: none">• The students should be able to develop applications using deep learning techniques for machine learning tasks.• The students should be able to measure the efficacy of the developed machine learning applications using scientific methods.• The students should be familiar with PyTorch, a library for implementing neural network based machine learning applications and experiments | | | | | |
| Inhalt This praktikum teaches the basics of the development of applications for machine learning tasks. The praktikum is aimed at but not necessarily limited to natural language processing (NLP) tasks, such as machine translation or sentiment analysis. NLP tasks are aimed at the ability to process or generate natural language (e.g. English) texts. The course is started with a short introduction to deep learning, NLP, Python programming and a development environment setup (installing dependencies and setting up an IDE (e.g. PyCharm)). Subsequently, teams of up to 3 students will choose (and may propose) topics that consist of a machine learning task. These tasks have to be solved by the students over the course of one semester, with regular meetings to discuss progress. | | | | | |
| Lehrformen Introductory lectures in the first weeks, followed by independent work on the chosen practical task, supported by regular meetings. | | | | | |
| Prüfungsformen Written report describing the motivation, methodology and result of the chosen task in Latex (max. 10 pages in ACL format), code for the task and its documentation, final presentation using slides (15 min. + 5 min. discussion) | | | | | |
| Voraussetzungen für die Vergabe von Credits Written report, documented code, participation in meetings, final presentation | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 5/105: M.Sc. Angewandte Informatik | | | | | |

Titel des Moduls: Master-Praktikum: Autonomous Robotics

| | | | | | |
|--|------------------------|-------------------------|---|---------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 3 CP | Workload 90 h | Semester | Turnus Jedes Semester | Dauer 1 Semester |
| Lehrveranstaltungen Autonomous Robotics | | | Kontaktzeit 40h Blockpraktikum | Selbststudium 50 h | Gruppengröße Studierende |
| Unterrichtssprache Deutsch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Greogor Schöner Lehrende: Prof. Dr. Greogor Schöner | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) <ul style="list-style-type: none">• Einüben des praktischen Arbeitens im Umfeld der autonomen Robotik, insbesondere der Entwurf von mathematischen Verfahren und ihre algorithmische Implementation zur Erzeugung von zeitlich strukturierten Abläufen• Einüben der Dokumentation von technischen Programmieraufgaben.• Verständnis der Grundlagen der autonomen Robotik anhand des einfachsten Robotersystems, eines autonomen Kleinstvehikels.• Einüben des Lesens und Schreibens von wissenschaftlich-technischen Artikeln | | | | | |
| Inhalt <p>Ziel des Praktikums ist es, die Grundlagen der autonomen Robotik anhand von Experimenten zu vermitteln, die eine enge Kopplung an aktuelle wissenschaftliche Probleme dieses Feldes aufweisen. Am Beispiel der Kleinstroboter vom Typ E-Puck wird hierbei die Erzeugung und Organisation von künstlichem Verhalten behandelt. Unter Verwendung der robotereigenen Sensorik werden experimentelle Aufgaben durch Implementierungen in der Programmierumgebung MATLAB gelöst. Der theoretische Hintergrund der Lösungen ist der dynamische Systeme Ansatz der kognitiven Robotik.</p> <p>Eine optionale Erweiterung des Praktikums durch das der dritte CP erzielt wird, ist das Lesen einer wissenschaftlichen Veröffentlichung im Umfeld der im Praktikum behandelten Methoden und das Schreiben eines Essays, in dem Fragen zu dem Artikel durch einen längeren, strukturierten und illustrierten Text beantwortet werden.</p> | | | | | |
| Lehrformen Praktikum | | | | | |
| Prüfungsformen Praktische Prüfung | | | | | |
| Voraussetzungen für die Vergabe von Credits Erfolgreiche Bearbeitung der Programmieraufgaben; Abgabe von schriftlichen Dokumentationen | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 3/105: M.Sc. Angewandte Informatik | | | | | |

| Titel des Moduls: Master-Praktikum: Big Data in der Bioinformatik | | | | | |
|--|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Jedes Semester | Dauer 1 Semester |
| Lehrveranstaltungen Big Data in der Bioinformatik (202621 bzw. 190801) | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße Studierende |
| Unterrichtssprache Deutsch oder Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Axel Mosig Lehrende: Prof. Dr. Axel Mosig | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) Nach dem erfolgreichen Abschluss des Moduls <ul style="list-style-type: none"> • können Studierende praktischen Herausforderungen bei der Entwicklung von Bioinformatik-Anwendungen begegnen, • haben die Studierenden Programmier-Bibliotheken aus dem Bereich Bioinformatik kennen gelernt • haben Studierenden die Verwendung von Workflow-Systemen eingeübt • haben Studierende die Mechanismen von Code- und Programm-Dokumentation eingeübt • haben Studierende Verfahren zur Bereitstellung eigener Libraries (z. B. R-Pakete, .jar-Files) angewendet | | | | | |
| Inhalt Die Bioinformatik wendet naturgemäß Informatik-Methoden auf die Daten eines lebenswissenschaftlichen Anwendungsfaches an, stellt also per se angewandte Digitalisierung dar. Das Praktikum vermittelt Grundlagen der Programmierung mit Bezug zu lebenswissenschaftlichen Anwendungen mit großen Datenmengen. Dies geschieht anhand aktueller Beispiele aus den Themengebieten Bildverarbeitung und Sequenzanalyse. Nach einer kurzen Einführung (Präsenztreffen) in Programmierung und Entwicklungsumgebungen (z. B. Java, R, C++, eclipse, RStudio, Python, Matlab) werden praktische Programmieraufgaben ausgegeben und im Laufe des Semesters bearbeitet, ggfls. mit weiteren Präsenztreffen zur Diskussion des Fortschritts. | | | | | |
| Lehrformen Einführung als seminaristischer Unterricht, Bearbeitung der praktischen Aufgabe selbständig oder als Gruppenarbeit. | | | | | |
| Prüfungsformen Semesterbegleitend; Praktische Prüfung: Protokoll max. 10 Seiten (wenn Abgabe bis zwei Wochen nach Praktikumsende, dann Korrekturrunde vor Benotung), Dokumentation der praktischen Aufgabe | | | | | |
| Voraussetzungen für die Vergabe von Credits Protokoll (siehe Prüfungsformen), Bearbeitung / Dokumentation der praktischen Aufgabe, Teilnahme an den Präsenztreffen, Abschlusspräsentation (15 min. + 5 min. Diskussion) | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 5/105: M.Sc. Angewandte Informatik | | | | | |

| Titel des Moduls: Master-Praktikum: Computational Proteomics | | | | | |
|---|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Jedes Semester | Dauer 1 Semester |
| Lehrveranstaltungen Master-Praktikum: Computational Proteomics (201914) | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße Studierende |
| Unterrichtssprache Deutsch oder Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Martin Eisenacher Lehrende: Prof. Dr. Martin Eisenacher | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) Nach dem erfolgreichen Abschluss des Moduls <ul style="list-style-type: none"> • können Studierende praktischen Herausforderungen bei der Entwicklung von Bioinformatik-Anwendungen begegnen, • haben die Studierenden Programmier-Bibliotheken aus dem Bereich Bioinformatik kennen gelernt • haben Studierenden die Verwendung von Workflow-Systemen eingeübt • haben Studierende die Mechanismen von Code- und Programm-Dokumentation eingeübt • haben Studierende Verfahren zur Bereitstellung eigener Libraries (z. B. R-Pakete, .jar-Files) angewendet | | | | | |
| Inhalt Die Bioinformatik wendet naturgemäß Informatik-Methoden auf die Daten eines lebenswissenschaftlichen Anwendungsfaches an, stellt also per se angewandte Digitalisierung dar. Das Praktikum vermittelt Grundlagen der Programmierung mit Bezug zu lebenswissenschaftlichen Anwendungen mit großen Datenmengen. Dies geschieht anhand aktueller Beispiele aus dem Themengebiet Hochdurchsatz Massenspektrometrie/ Proteomics. Nach einer kurzen Einführung (Präsenztreffen) in Programmierung und Entwicklungsumgebungen (z. B. Java, R, C++, eclipse, RStudio, Python, Matlab, KNIME) werden praktische Programmieraufgaben ausgegeben und im Laufe des Semesters bearbeitet, ggfls. mit weiteren Präsenztreffen zur Diskussion des Fortschritts. | | | | | |
| Lehrformen Einführung als seminaristischer Unterricht, Bearbeitung der praktischen Aufgabe selbständig oder als Gruppenarbeit. | | | | | |
| Prüfungsformen Semesterbegleitend; Praktische Prüfung: Protokoll max. 10 Seiten (wenn Abgabe bis zwei Wochen nach Praktikumsende, dann Korrekturrunde vor Benotung), Dokumentation der praktischen Aufgabe | | | | | |
| Voraussetzungen für die Vergabe von Credits Protokoll (siehe Prüfungsformen), Bearbeitung / Dokumentation der praktischen Aufgabe, Teilnahme an den Präsenztreffen, Abschlusspräsentation (15 min. + 5 min. Diskussion) | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 5/105: M.Sc. Angewandte Informatik | | | | | |

| | | | | | |
|---|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Titel des Moduls: Model Checking | | | | | |
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Model Checking | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße Studierende |
| Unterrichtssprache Deutsch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Thomas Zeume Lehrende: Dr. Nils Vortmeier Marko Schmellenkamp | | | | | |
| Verwendung des Moduls B.Sc. Informatik M.Sc. Informatik M.Sc. Angewandte Informatik B.Sc. IT-Sicherheit/Informationstechnik M.Sc. Mathematik | | | | | |
| Lernziele (learning outcomes) Die Studierenden lernen wie sich verteilte Systeme durch Transitionssysteme modellieren und Eigenschaften in logischen Spezifikationssprachen wie LTL und CTL spezifizieren lassen. Sie sollen elementare Algorithmen zur Überprüfung von Eigenschaften in Transitionssystemen kennenlernen. Sie sollen ein Verständnis für die Möglichkeiten und Grenzen des Model Checking entwickeln, und in die Lage versetzt werden, sich eigenständig mit fortgeschrittenen Methoden des Model Checkings auseinanderzusetzen. | | | | | |
| Inhalt Wie kann die Korrektheit von Software und Hardware formal überprüft werden? Im Model Checking werden Software- und Hardware-Module durch Transitionssysteme formalisiert; gewünschte Eigenschaften mit Hilfe logischer Formalismen formal beschrieben; und mit Hilfe von Algorithmen automatisiert überprüft, ob ein Transitionssystem eine formal spezifizierte Eigenschaft besitzt. In dieser Veranstaltung werden die theoretischen Grundlagen des Model Checkings vermittelt, mit einem Fokus auf logik-basierten Spezifikationssprachen. Die Spezifikationssprachen LTL und CTL werden eingeführt, ihre Ausdrucksstärke untersucht, und die wichtigsten algorithmischen Ansätze für das Model Checking vorgestellt. | | | | | |
| Lehrformen Vorlesung mit Übungen | | | | | |
| Prüfungsformen Abschlussprüfung; mündliche Prüfung (20-30min) oder schriftliche Klausur (120min) in Abhängigkeit der Teilnehmerzahl | | | | | |
| Voraussetzungen für die Vergabe von Credits Erfolgreiches Bestehen der Modulprüfung. | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 5/170: B.Sc. Informatik [PO 22] 5/158: B.Sc. Informatik [PO 20] 5/97: M.Sc. Informatik 5/105: M.Sc. Angewandte Informatik | | | | | |

5/150: B.Sc. IT-Sicherheit/Informationstechnik [PO 22]

5/149: B.Sc. IT-Sicherheit/Informationstechnik [PO 22]

| | | | | | |
|--|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Titel des Moduls: Parallel Computing | | | | | |
| Modul-Nr./Code | Credits 6 CP | Workload 180 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Parallel Computing (126519) | | | Kontaktzeit 60 h | Selbststudium 120 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Markus König Lehrende: Prof. Dr. Markus König | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) Goals of the course "Parallel Computing" include a basic understanding of the theoretical foundation for efficient parallel algorithms, the architecture of parallel hardware systems and current parallel software paradigms used in research and industry today. Also, students learn how to develop parallel algorithms and implement them using state-of-the-art software systems and programming tools. In particular, numerically intensive engineering applications are a focus within the lecture itself as well as a target for the team projects carried out by the students of this course. Further, students must clearly present their projects results in a classroom setting to an audience with various technical background (the course is attended by students from applied computer science as well as by students from computational engineering). When the students have successfully complete this course, <ul style="list-style-type: none"> students will be familiar with the development of parallel algorithms and parallel software systems and have a good understanding of how to realize efficient parallel programs; students will have a good understanding of the basics of parallel programming to be able to, for example, select a proper parallel software system or tool that is appropriate for a given numerical engineering task, or be able to discuss possible solutions concerning necessary hardware or software requirements at a competent level. | | | | | |
| Inhalt <ul style="list-style-type: none"> Introduction to parallel computing using examples from the field of engineering Theoretical foundations of parallel computing (concurrency, parallel processes, dead-locks, Amdahl's Law, Flynn's taxonomy, efficiency metrics, memory models, etc.) Parallel programming based on "shared memory" using the OpenMP application programming interface Parallel programming based on "distributed memory" based the Message Passing Interface (MPI) Hardware based parallel programming based on the General Purpose Graphical Processing Unit (GPGPU) Application of parallel programming paradigms to solve engineering tasks as a teamproject | | | | | |
| Lehrformen <ul style="list-style-type: none"> lectures exercises using computers team projects | | | | | |
| Prüfungsformen completion of a team project (2-4 students) presentation of project results (end of semester) oral examination (end of semester) | | | | | |
| Voraussetzungen für die Vergabe von Credits <ul style="list-style-type: none"> successful completion of a team project | | | | | |

- presentation of project results in the classroom
- oral examination

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

6/105: M.Sc. Angewandte Informatik

| | | | | | |
|---|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Titel des Moduls: Product Lifecycle Management | | | | | |
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Product Lifecycle Management (138577) | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße Studierende |
| Unterrichtssprache Deutsch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr.-Ing. Detlef Gerhard Lehrende: Prof. Dr.-Ing. Detlef Gerhard | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik MSc. Maschinenbau MSc. Sales Engineering and Product Management | | | | | |
| Lernziele (learning outcomes) Nach dem erfolgreichen Abschluss des Moduls <ul style="list-style-type: none"> • verfügen Studierende über ein breites, integriertes Wissen über die Herausforderungen des ganzheitlichen Informationsmanagements im Produktlebenszyklus und die resultierenden Anforderungen an Softwaresysteme zur Unterstützung von PLM. • kennen und verstehen Studierende die Teilprozesse des Produktlebenszyklus, die Methoden des Product Lifecycle Management (PLM) sowie die wissenschaftlichen Grundlagen der zugrundeliegenden IT-Systeme. Indem sie praktische Beispiele und Aufgaben mit entsprechender Anwendungssoftware bearbeiten, können sie die erlernten Fertigkeiten im Umgang mit PLM Software auf konkrete und praxisorientierte PLM Problemstellungen übertragen. • haben Studierende ein umfassendes Verständnis vom Zusammenwirken der PLM Softwaresysteme und können kritisch die Eignung von Methoden zum Objektmanagement, zum Produktstruktur- und Konfigurationsmanagement sowie zum Projekt- und Prozessmanagement differenzieren und beurteilen. • können Studierende prozessorientiert an PLM Aufgabenstellungen herangehen, diese reflektieren und bewerten sowie selbstgesteuert verfolgen. • können Studierende kooperativ PLM-Aufgabenstellungen in heterogenen Gruppen bearbeiten, Abläufe und Ergebnisse begründen sowie über Sachverhalte umfassend kommunizieren. | | | | | |
| Inhalt Das Modul vermittelt Methoden und Werkzeuge zum Product Lifecycle Management (PLM), insbesondere das dazu erforderliche Grundlagenwissen und die relevanten methodischen Aspekte von Produktinnovationsprozessen. Schwerpunkte bilden dabei die verschiedenen PLM-Funktionen entsprechender Softwaresysteme (z.B. Teile-, Dokumenten- und Produktstrukturmanagement, Klassifizierung, Konfigurationsmanagement, Projekt- und Prozessmanagement). Weiterhin werden allgemeine Methoden zur Organisation und Handhabung von Produktdaten und Benutzerinformationen sowie Methoden des Collaborative Engineerings und die Vorgehensweise bei der PLM-Einführung vermittelt. | | | | | |
| Lehrformen Vorlesung mit Übung | | | | | |
| Prüfungsformen Klausur 'Modulabschlussprüfung: Product Lifecycle Management' (90 Min., Anteil der Modulnote 100 %) Studienbegleitende Aufgaben: Hausarbeiten (Sofern die Hausarbeiten vor der Modulabschlussprüfung absolviert werden, sind optional Bonuspunkte für die Klausur möglich) (Umfang und Abgabefristen wird am Anfang des Semesters bekanntgegeben) | | | | | |

Voraussetzungen für die Vergabe von Credits

- Bestandene Modulabschlussprüfung: Klausur
- Bestandene studienbegleitende Aufgaben: Hausarbeiten (Umfang und Abgabefristen wird am Anfang des Semesters bekanntgegeben)

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

6/105: M.Sc. Angewandte Informatik

| | | | | | |
|---|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Titel des Moduls: Quantum Information and Computation | | | | | |
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen Quantum Information and Computation | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße Studierende |
| Unterrichtssprache German or Englisch (depends on audience) | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Michael Walter Lehrende: Prof. Dr. Michael Walter | | | | | |
| Verwendung des Moduls B.Sc. Informatik B.Sc. IT-Sicherheit/Informationstechnik M.Sc. Angewandte Informatik M.Sc. IT-Sicherheit Informationstechnik M.Sc. IT-Sicherheit Netze und Systeme | | | | | |
| Lernziele (learning outcomes) You will learn fundamental concepts, algorithms, and results in quantum information and computation. After successful completion of this course, you will know the theoretical model of quantum information and computation, how to generalize computer science concepts to the quantum setting, how to design and analyze quantum algorithms and protocols for a variety of computational problems, and how to prove complexity theoretic lower bounds. You will be prepared for an advanced course or a research or thesis project in this area. | | | | | |
| Inhalt This course will give an introduction to quantum information and quantum computation from the perspective of theoretical computer science. Topics to be covered will likely include: <ul style="list-style-type: none"> • Fundamentals of quantum computing: quantum bits, states and operations • The power of quantum entanglement: nonlocal games • Entanglement as a resource: superdense coding and teleportation • Quantum circuit model of computation • Quantum computing with oracles: Deutsch-Jozsa, Bernstein-Vazirani, Simon • Quantum Fourier transform and phase estimation • Shor's factoring algorithm • Grover's search algorithm and beyond: how to solve SAT on a quantum computer? • From no cloning to quantum money: a peek at quantum cryptography The course should be of interest to students of computer science, mathematics, physics, and related disciplines. Students interested in a BSc or MSc project in quantum information, computing, cryptography, etc. are particularly encouraged to participate. | | | | | |
| Lehrformen Lecture with Exercise | | | | | |
| Prüfungsformen Written Exam (120 Minutes) | | | | | |
| Voraussetzungen für die Vergabe von Credits Passed written Exam. | | | | | |

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

5/158: B.Sc. Informatik [PO 22]

5/165: B.Sc. Informatik [PO 20]

5/150: B.Sc. IT-Sicherheit/Informationstechnik [PO 22]

5/149: B.Sc. IT-Sicherheit/Informationstechnik [PO 20]

5/105: M.Sc. Angewandte Informatik

5 /91: M.Sc. IT-Sicherheit | Informationstechnik [PO 22]

5/ 99: M.Sc. IT-Sicherheit | Netze und Systeme [PO 22]

| | | | | | |
|--|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Titel des Moduls: Schwerpunktseminar Computational Linguistics | | | | | |
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Jedes Semester | Dauer 1 Semester |
| Lehrveranstaltungen siehe Vorlesungsverzeichnis | | | Kontaktzeit 30 h | Selbststudium | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Tibor Kiss Lehrende: wechselt ja nach Seminar | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) <ul style="list-style-type: none"> • Die Studierenden sind in der Lage, kleinere Problemstellungen der Linguistic Data Science im jeweiligen Schwerpunktbereich angemessen zu formulieren. • Sie kennen verschiedene Ansätze zur Problemlösung im jeweiligen Schwerpunktbereich und können einige davon kritisch auf Fragestellungen und Probleme anwenden. • Sie verstehen, dass es in vielen Bereichen keine einzig richtige Antwort gibt, und können sich mit anderen konstruktiv über Lösungsansätze austauschen und Vor- und Nachteile von unterschiedlichen Perspektiven abwägen. • Die Studierenden kennen weiterführende computerlinguistische Methoden, Modelle und Theorien und können sie kritisch auf Fragestellungen und Probleme anwenden. • Sie sind im Umgang mit den Programmiersprachen vertraut und können mit weiterführenden Anleitungen oder Dokumentationen umgehen. • Sie planen einfache Programme strukturiert und können sie gemäß ihrer Planung implementieren. • Sie lesen und verstehen Programmcode von anderen und schreiben und dokumentieren ihren eigenen Code für andere nachvollziehbar. • Sie diskutieren Theorien und Umsetzungen mit anderen kritisch. • Typische Bias-Fallen durch Datencodierung oder Implementierung von Algorithmen sind ihnen vertraut und sie können bewusste Entscheidungen über kritische Fälle treffen. | | | | | |
| Inhalt In Seminaren des Schwerpunkts Computational Linguistics lernen die Studierenden Fragestellungen, Theorien und Methoden aus den Bereichen Computerlinguistik und Sprachmodellierung kennen. In einzelnen Seminaren beschäftigen sie sich auch mit spezifischen Tools, Frameworks oder Anwendungsbereichen. Ein besonderes Augenmerk soll dabei auf den Konsequenzen von perspektivischem oder technischem Bias liegen und der Verantwortung von Computerlinguisten im Umgang mit teils persönlichen Sprachdaten. | | | | | |
| Lehrformen Das Modul besteht aus einem Seminar, in denen unterschiedliche Lernformen angewendet werden, wie beispielsweise <ul style="list-style-type: none"> • Vortrag der Lehrenden im Plenum • Gastvorträge • Vortrag der Studierenden im Plenum • Diskussionen im Plenum • Arbeitsgruppen • Arbeitsaufgaben • Selbststudium | | | | | |
| Prüfungsformen Seminararbeit (semesterbegleitend) | | | | | |
| Voraussetzungen für die Vergabe von Credits | | | | | |

Die zu erbringende Studienleistung für ein Seminar, die mögliche Prüfungsleistung in diesem Seminar und die Schwerpunktzuordnung der Veranstaltung werden im Vorlesungsverzeichnis angegeben.
Dozierende können für die Seminare die aktive Teilnahme durch Teilnahme an Diskussionen oder Reflektionsrunden in den Sitzungen verpflichtend machen. Ergibt sich dadurch eine Anwesenheitspflicht zu Veranstaltungsterminen, muss dies im Vorlesungsverzeichnis zur Veranstaltung angemerkt werden.

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

| | | | | | |
|--|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Titel des Moduls: Schwerpunktseminar Linguistic Data Science | | | | | |
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Jedes Semester | Dauer 1 Semester |
| Lehrveranstaltungen siehe Vorlesungsverzeichnis | | | Kontaktzeit 30 h | Selbststudium | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Tibor Kiss Lehrende: wechselt ja nach Seminar | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) <ul style="list-style-type: none"> • Die Studierenden sind in der Lage, kleinere Problemstellungen der Linguistic Data Science im jeweiligen Schwerpunktbereich angemessen zu formulieren. • Sie kennen verschiedene Ansätze zur Problemlösung im jeweiligen Schwerpunktbereich und können einige davon kritisch auf Fragestellungen und Probleme anwenden. • Sie verstehen, dass es in vielen Bereichen keine einzig richtige Antwort gibt, und können sich mit anderen konstruktiv über Lösungsansätze austauschen und Vor- und Nachteile von unterschiedlichen Perspektiven abwägen. • Die Studierenden kennen weiterführende computerlinguistische Methoden, Modelle und Theorien und können sie kritisch auf Fragestellungen und Probleme anwenden. • Sie sind im Umgang mit den Programmiersprachen vertraut und können mit weiterführenden Anleitungen oder Dokumentationen umgehen. • Sie planen einfache Programme strukturiert und können sie gemäß ihrer Planung implementieren. • Sie lesen und verstehen Programmcode von anderen und schreiben und dokumentieren ihren eigenen Code für andere nachvollziehbar. • Sie diskutieren Theorien und Umsetzungen mit anderen kritisch. • Typische Bias-Fallen durch Datencodierung oder Implementierung von Algorithmen sind ihnen vertraut und sie können bewusste Entscheidungen über kritische Fälle treffen. | | | | | |
| Inhalt In Seminaren des Schwerpunkts Computational Linguistics lernen die Studierenden Fragestellungen, Theorien und Methoden aus den Bereichen Computerlinguistik und Sprachmodellierung kennen. In einzelnen Seminaren beschäftigen sie sich auch mit spezifischen Tools, Frameworks oder Anwendungsbereichen. Ein besonderes Augenmerk soll dabei auf den Konsequenzen von perspektivischem oder technischem Bias liegen und der Verantwortung von Computerlinguisten im Umgang mit teils persönlichen Sprachdaten. | | | | | |
| Lehrformen Das Modul besteht aus einem Seminar, in denen unterschiedliche Lernformen angewendet werden, wie beispielsweise <ul style="list-style-type: none"> • Vortrag der Lehrenden im Plenum • Gastvorträge • Vortrag der Studierenden im Plenum • Diskussionen im Plenum • Arbeitsgruppen • Arbeitsaufgaben • Selbststudium | | | | | |
| Prüfungsformen Seminararbeit (semesterbegleitend) | | | | | |
| Voraussetzungen für die Vergabe von Credits | | | | | |

Die zu erbringende Studienleistung für ein Seminar, die mögliche Prüfungsleistung in diesem Seminar und die Schwerpunktzuordnung der Veranstaltung werden im Vorlesungsverzeichnis angegeben.
Dozierende können für die Seminare die aktive Teilnahme durch Teilnahme an Diskussionen oder Reflektionsrunden in den Sitzungen verpflichtend machen. Ergibt sich dadurch eine Anwesenheitspflicht zu Veranstaltungsterminen, muss dies im Vorlesungsverzeichnis zur Veranstaltung angemerkt werden.

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

| | | | | | |
|---|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Titel des Moduls: Simulationstechnik | | | | | |
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen Simulationstechnik (127022) | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße Studierende |
| Unterrichtssprache Deutsch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Markus K Lehrende: Prof. Dr. Markus K | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) Dieses Modul vermittelt Kompetenzen zum Einsatz von rechnergestützten Simulationskonzepten zur Lösung praxisbezogener Aufgabenstellungen im Bau- und Umweltingenieurwesen. Im Rahmen der Übungsveranstaltungen erfolgt eine Einführung in aktuelle Simulations- und Optimierungssoftware. Die Projektarbeit wird als Gruppenarbeit durchgeführt. | | | | | |
| Inhalt Es werden Vorgehensweisen zur Simulation komplexer Systeme vermittelt. Folgende Themen werden behandelt: <ul style="list-style-type: none"> • Systemanalyse und Modellbildung • System Dynamics • Ereignisdiskrete Simulation • Multiagentensimulation • Aufbereitung von Eingangsdaten • Stochastische Simulation • Simulationsgestützte Optimierung • Einführung in die Software AnyLogic <p>Innerhalb der Projektarbeit werden aktuelle Fragestellungen aus den Bau- und Umweltingenieurwissenschaften (Montage- und Logistikprozesse, Fußgängersimulation, Schadstoffausbreitung, etc.) aufgearbeitet und mit Hilfe einer objektorientierten Simulationssoftware analysiert. Den Studierenden werden Softwarelizenzen durch den Lehrstuhl zur Verfügung gestellt. Die Programmierung erfolgt unter Verwendung der Programmiersprache Java.</p> | | | | | |
| Lehrformen Vorlesung mit Übung | | | | | |
| Prüfungsformen Projektarbeit mit Abschlusspräsentation | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestandene Projektarbeit mit Abschlusspräsentation | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 5/105: M.Sc. Angewandte Informatik | | | | | |

| | | | | | |
|---|-------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Titel des Moduls: Studienprojekt (Master) | | | | | |
| Modul-Nr./Code | Credits 10 CP | Workload 300 h | Semester 3 | Turnus Jedes Semester | Dauer 1 Semester |
| Lehrveranstaltungen Individuelle Team- bzw. Gruppenprojekte | | | Kontaktzeit 30 h | Selbststudium 270 h | Gruppengröße Studierende |
| Unterrichtssprache Deutsch oder Englisch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Studiendekan Angewandte Informatik Lehrende: Dozierende im Studiengang Angewandte Informatik | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) Ziele des Studienprojekts sind: <ul style="list-style-type: none"> • Die Anwendung des erlernten Fachwissens. • Der Erwerb zusätzlicher Fachkompetenz gemäß der jeweiligen projektspezifischen Aufgabenstellung. • Die Schulung hinsichtlich der Erarbeitung eigener Lösungsstrategien. • Die Schulung hinsichtlich Arbeitsteilung und Zusammenarbeit im Team (Teamfähigkeit und Projektorganisation). • Erwerb weiterer Kompetenz im Hinblick auf die Dokumentation und die Präsentation von Ergebnissen. | | | | | |
| Inhalt Im Rahmen des Studienprojekts soll eine Aufgabe aus Bereichen der Angewandten Informatik in Teamarbeit unter Anleitung eines Betreuers gelöst werden. Die angebotenen Projekte decken dabei thematisch die gesamte Bandbreite der Vertiefungsmodule ab. Darüber hinaus werden auch interdisziplinäre Studienprojekte angeboten. In diesen arbeiten z.B. Angewandte Informatiker mit Studierenden der Sozialwissenschaften oder der Sportwissenschaft zusammen. | | | | | |
| Lehrformen Betreutes Projekt | | | | | |
| Prüfungsformen Projektarbeit mit Abschlussbericht | | | | | |
| Voraussetzungen für die Vergabe von Credits Aktive Teilnahme am Projekt, fristgerechte Abgabe des Abschlussberichts und Bewertung des Abschlussberichts mit min. ausreichend. Hinweis: Jeder Abschlussbericht muss im Umfang von einer DinA4 Seite eine Reflexion über das methodische Vorgehen enthalten. | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 10/105: M.Sc. Angewandte Informatik | | | | | |

| | | | | | |
|---|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Titel des Moduls: Symmetrische Kryptanalyse | | | | | |
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Wintersemester | Dauer 1 Semester |
| Lehrveranstaltungen Symmetrische Kryptanalyse | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße Studierende |
| Unterrichtssprache Deutsch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Nils-Gregor Leander Lehrende: Prof. Dr. Nils-Gregor Leander | | | | | |
| Verwendung des Moduls M.Sc. IT-Sicherheit/Informationstechnik M.Sc. IT-Sicherheit/Netze und Systeme M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) Die Studierenden haben ein vertieftes Verständnis für die Sicherheit symmetrischer Chiffren. | | | | | |
| Inhalt Wir behandeln die wichtigsten Themen in der symmetrischen Kryptanalyse. Nach einer ausführlichen Vorstellung von linearer und differentieller Kryptanalyse werden weitere Angriffe auf symmetrische Primitive, insbesondere Block-Chiffren behandelt. Hierzu zählen insbesondere Integral (auch Square) Attacks, Impossible Differentials, Boomerang-Angriffe und Slide-Attacks. Neben den Angriffen selbst werden auch immer die daraus resultierenden Design-Kriterien beschrieben, um neue Algorithmen sicher gegen die Angriffe zu machen. | | | | | |
| Lehrformen | | | | | |
| Prüfungsformen Mündliche Modulabschlussprüfung (30 Minuten) | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestandene mündliche Modulabschlussprüfung. | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 5/91: M.Sc. IT-Sicherheit/Informationstechnik [PO 22] 5/84: M.Sc. IT-Sicherheit/Informationstechnik [PO 20] 5/99: M.Sc. IT-Sicherheit/Netze und Systeme [PO 22] 5/96: M.Sc. IT-Sicherheit/Netze und Systeme [PO 20] 5/105: M.Sc. Angewandte Informatik | | | | | |

| | | | | | |
|--|------------------------|--------------------------|--|---------------------------------|------------------------------------|
| Titel des Moduls: Systemsicherheit System Security | | | | | |
| Modul-Nr./Code | Credits 5 CP | Workload 150 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Systemsicherheit | | | Kontaktzeit 60 h | Selbststudium 90 h | Gruppengröße Studierende |
| Unterrichtssprache Englisch | | | Teilnahmevoraussetzungen keine | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Ghassan Karame Lehrende: Prof. Dr. Ghassan Karame | | | | | |
| Verwendung des Moduls B.Sc. IT-Sicherheit / Informationstechnik B.Sc. Informatik M.Sc. Angewandte Informatik M.Sc. IT-Sicherheit / Netze und Systeme | | | | | |
| Lernziele (learning outcomes) At the end of this course, students will be able to <ul style="list-style-type: none"> • classify and describe vulnerabilities and protection mechanisms of popular systems and protocols, and • analyze / reason about basic protection mechanisms for modern OSs, software, and hardware systems. Students will also develop the ability to reason about the security of a given protocol and independently develop appropriate security defenses and security models. | | | | | |
| Inhalt While clearly beneficial, the large-scale deployment of online services has resulted in the increase of security threats against existing services. As the size of the global network grows, the incentives of attackers to abuse the operation of online applications also increase and their advantage in mounting successful attacks becomes considerable. These cyber-attacks often target the resources, availability, and operation of online services. With an increasing number of services relying on online resources, integrating proper security measures therefore becomes integral to ensure the correct functioning of every online service. In this course, we discuss important theoretical and analytical aspects in system security. The focus of the course is to understand basic attack strategies on modern systems and platforms, with a focus on side-channel attacks, software-based attacks, malware analysis, as well as software-based defenses (e.g., address space randomization and non-executable memory) and hardware-based defenses (e.g., using TPMs and TEEs). Other topics of the course include analyzing the security of modern cryptocurrencies and ML platforms, and similar aspects in system security. An integral part of this course are exercises and homeworks, which aim to deepen the understanding of the material with practical examples. | | | | | |
| Lehrformen Lecture with Exercise | | | | | |
| Prüfungsformen Written Exam (120 Minutes) | | | | | |
| Voraussetzungen für die Vergabe von Credits Passed written Exam | | | | | |

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

5/150: B.Sc. IT-Sicherheit / Informationstechnik [PO 22]

5/149: B.Sc. IT-Sicherheit / Informationstechnik [PO 20]

5/158: B.Sc. Informatik [PO 22]

5/165: B.Sc. Informatik [PO 20]

5/105: M.Sc. Angewandte Informatik

5/99: M.Sc. IT-Sicherheit / Netze und Systeme [PO 22]

5/96: M.Sc. IT-Sicherheit / Netze und Systeme [PO 20]

| | | | | | |
|--|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Titel des Moduls: Theorie des maschinellen Lernens | | | | | |
| Modul-Nr./Code | Credits 9 CP | Workload 270 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen Theorie des maschinellen Lernens (211052) | | | Kontaktzeit 90 h | Selbststudium 180 h | Gruppengröße Studierende |
| Unterrichtssprache Deutsch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr. Asja Fischer Lehrende: Prof. Dr. Asja Fischer | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik M.Sc. Informatik | | | | | |
| Lernziele (learning outcomes) Die Studierenden werden mit mathematischen Modellen für das maschinelle Lernen vertraut gemacht. Sie erwerben die Fähigkeit, Lernalgorithmen zu beurteilen und zu vergleichen anhand des Grades, in welchem diese (exakt beschriebene) Erfolgskriterien erreichen. Sie erwerben Techniken sowohl zum Design effizienter Lernalgorithmen als zum Nachweis der inhärenten Härte eines Problems. Nach dem erfolgreichen Abschluss des Moduls <ul style="list-style-type: none"> • kennen Studierende die wichtigsten Lernmaschinen (wie zum Beispiel Support Vector Machines und verwandte Modelle), • verstehen Studierende den Unterschied zwischen empirischer und realer Fehlerrate und kennen Techniken zum Umgang mit dem Problem des overfitting der Daten (mit einem zu komplexen Modell), • können Studierende zwischen uniformer und nicht uniformer Lernbarkeit einer Hypothesenklasse unterscheiden und kennen die dazu passenden Theorien und Lernregeln. | | | | | |
| Inhalt Gegenstand der Vorlesung ist die statistik-basierte Theorie des maschinellen Lernens. Insbesondere wird die Methode der strukturierten Risikominimierung vermittelt sowie die ihr zugrunde liegenden statistischen Lehrsätze. Es werden sowohl Techniken zum Entwurf effizienter Lernalgorithmen besprochen als auch informations- oder berechnungstheoretische Barrieren, die bestimmte Lernprobleme als nicht effizient lösbar erscheinen lassen. | | | | | |
| Lehrformen Vorlesung mit Übung | | | | | |
| Prüfungsformen mündliche Modulabschlussprüfung (30 Minuten) | | | | | |
| Voraussetzungen für die Vergabe von Credits Bestandene mündliche Modulabschlussprüfung. | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 9/105: M.Sc. Angewandte Informatik 9/97: M.Sc. Informatik | | | | | |

Titel des Moduls: Verkehrstechnik

| | | | | | |
|---|------------------------|--------------------------|---------------------------------|---------------------------------|------------------------------------|
| Modul-Nr./Code | Credits 6 CP | Workload 180 h | Semester | Turnus Sommersemester | Dauer 1 Semester |
| Lehrveranstaltungen a) Verkehrssteuerung (128026) b) Modellierung und Simulation des Verkehrsflusses (128025) | | | Kontaktzeit 60 h | Selbststudium 120 h | Gruppengröße Studierende |
| Unterrichtssprache Deutsch | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Prof. Dr.-Ing. Justin Geistefeldt Lehrende: Prof. Dr.-Ing. Justin Geistefeldt, Prof. Dr.-Ing. Ning Wu | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik M.Sc. Bau- und Umweltingenieurwesen | | | | | |
| Lernziele (learning outcomes) a) Die Hörer verfügen über das aktuelle technische Wissen und besitzen erweiterte Kenntnisse über die Methoden der verkehrstechnischen Analyse und Steuerung von Knotenpunkten. Sie haben die Fähigkeit, die in der Praxis angewandten Planungstechniken für Lichtsignalanlagen zu verstehen und komplexe Anlagen einschließlich einer Koordinierung praxisgerecht zu entwerfen. b) Die Hörer verfügen über differenzierte Kenntnisse der Gesetzmäßigkeiten des Verkehrsflusses auf Straßen. Sie sind in der Lage, wissenschaftliche Beschreibungsmöglichkeiten dieser Gesetzmäßigkeiten zu reflektieren und ihre praktische Anwendbarkeit zu erkennen. Sie haben die Fähigkeit, selbständig Erweiterungen oder Anpassungen von Verkehrsflussmodellen zu entwickeln. | | | | | |
| Inhalt a) Es werden Methoden der verkehrstechnischen Analyse und Bemessung von Straßenknotenpunkten sowie Steuerungssysteme für Knotenpunkte und die zu ihrem Betrieb erforderlichen Einrichtungen behandelt. Die in der Praxis üblichen Verfahren werden in der Übung an einigen Beispielen veranschaulicht. Dabei werden EDV-Verfahren eingesetzt. Im Einzelnen werden behandelt: Wartezeitermittlung an Knotenpunkten, vorfahrtgeregelte Knotenpunkte, Festzeitsteuerung von Signalanlagen, Grüne Welle, Koordinierung im Netz, verkehrsabhängige Steuerung einschließlich Signalprogrammbildung, Signaltechnik, Steuerungskriterien. b) Die theoretischen Grundlagen für die Beschreibung des Verkehrsflusses auf Straßen werden mit Hilfe mathematischer Verfahren erarbeitet. Die zu Grunde liegenden Gesetzmäßigkeiten werden hergeleitet. Im Einzelnen werden behandelt: Kenngrößen des Verkehrsablaufs und deren Zusammenhänge, Fundamentaldiagramm, Kapazität, freier Verkehrsfluss, Kontinuumstheorie, Abstandsmodelle, Fahrzeugfolge-theorie, mikroskopische Verkehrsflusssimulation. | | | | | |
| Lehrformen Powerpoint-Präsentationen, Tafel, Vorführungen und Übungen am PC | | | | | |

Prüfungsformen

Schriftliche Modulabschlussprüfung (120 Minuten); Klausur über das gesamte Modul.

Voraussetzungen für die Vergabe von Credits

Bestandene schriftliche Modulabschlussprüfung.

Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS)

6/105: M.Sc. Angewandte Informatik

| | | | | | |
|--|-------------------------|--------------------------|--|---|------------------------------------|
| Titel des Moduls: Freie Wahlmodule | | | | | |
| Modul-Nr./Code | Credits 15 CP | Workload 450 h | Semester | Turnus Jedes Semester | Dauer 1 Semester |
| Lehrveranstaltungen | | | Kontaktzeit abhängig von Veranstaltungswahl | Selbststudium Je nach Veranstaltungswahl | Gruppengröße Studierende |
| Unterrichtssprache Je nach Veranstaltungswahl | | | Teilnahmevoraussetzungen | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Studienfachberatung Angewandte Informatik Lehrende: | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) Die Teilnehmer erwerben so genannte Schlüsselfähigkeiten in den freien Wahlfächern | | | | | |
| Inhalt Studierende müssen Veranstaltungen im Gesamtumfang von 15 CP absolvieren. Je nach Veranstaltungswahl werden unterschiedliche Inhalte vermittelt. Studierende haben die Möglichkeit unter den Freien Wahlmodulen auch Fächer jenseits der Informatik zu absolvieren und ihre Soft Skills zu erweitern. Z.B. Die freien Wahlfächer erweitern die Soft Skills. Z.B. wird die englische Fachsprache verbessert, in die Grundlagen der Rechtswissenschaften eingeführt oder Grundkenntnisse der Betriebswirtschaft vermittelt. Die Studierenden haben die Möglichkeit eine Auswahl entsprechend der eigenen Interessen zu treffen. | | | | | |
| Lehrformen abhängig von Veranstaltungswahl | | | | | |
| Prüfungsformen abhängig von Veranstaltungswahl | | | | | |
| Voraussetzungen für die Vergabe von Credits abhängig von Veranstaltungswahl | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 0/105 (unbenotet): M.Sc. Angewandte Informatik | | | | | |

| Titel des Moduls: Masterarbeit und Kolloquium (AI) | | | | | |
|---|-------------------------|--------------------------|--|---------------------------------|--------------------------------------|
| Modul-Nr./Code | Credits 30 CP | Workload 900 h | Semester 4 | Turnus Jedes Semester | Dauer 1 Semester |
| Lehrveranstaltungen | | | Kontaktzeit 15 h | Selbststudium 885 h | Gruppengröße 1 Studierende |
| Unterrichtssprache Deutsch oder Englisch | | | Teilnahmevoraussetzungen M.Sc. Angewandte Informatik PO 22: Erfolgreich abgeschlossene Module im Umfang von 70 CP. M.Sc. Angewandte Informatik PO 20: Erfolgreich abgeschlossene Module im Umfang von 60 CP | | |
| Modulbeauftragte/r und hauptamtlich Lehrende Modulbeauftragte/r: Studiendekan Angewandte Informatik Lehrende: Lehrende im Studiengang Angewandte Informatik | | | | | |
| Verwendung des Moduls M.Sc. Angewandte Informatik | | | | | |
| Lernziele (learning outcomes) Nach erfolgreichem Abschluss des Moduls: <ul style="list-style-type: none"> • können Studierende selbstständig und fristgerecht ein wissenschaftliches Thema bearbeiten von der Recherche bis zur Dokumentation der Resultate • können Studierende geeignete wissenschaftliche Verfahren und Methoden, die sie im Studium kennengelernt haben, auswählen, anwenden und weiterentwickeln, um ein konkretes Problem zu lösen • können Studierende ihre Ergebnisse kritisch mit dem Stand der Forschung vergleichen und evaluieren • können Studierende ihre eigenen Ergebnisse angemessen in Wort und Schrift darstellen. | | | | | |
| Inhalt Es soll eine anspruchsvolle Fragestellung der Angewandten Informatik bearbeitet und dokumentiert werden. Im Anschluss an die Bearbeitung der Masterarbeit werden die Ergebnisse in Form eines Kolloquiumsvortrags mit anschließender Diskussion präsentiert. | | | | | |
| Lehrformen Abschlussarbeit | | | | | |
| Prüfungsformen Masterarbeit und Kolloquiumsvortrag | | | | | |
| Voraussetzungen für die Vergabe von Credits Sowohl die Masterarbeit als auch der Kolloquiumsvortrag müssen bestanden sein. Der Anteil der Kolloquiumsnote an der Gesamtnote beträgt 10% | | | | | |
| Stellenwert der Note für die Endnote (bei einem Gesamtstudienumfang von 120 ECTS) 30/105: M.Sc. Angewandte Informatik | | | | | |